RM0440
リファレンスマニュアル

STM32G4 シリーズ
advanced Arm®-based 32-bit MCU

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32G4 シリーズマイクロコントローラメモリとペリフェラルを使用する方法について、詳しく説明しています。

STM32G4 シリーズ は、さまざまなメモリ·サイズ、パッケージ、およびペリフェラルで構成するマイクロコントローラ·ファミリの製品です。

注文情報、機械的特性、電気的特性については、該当のデータシートを参照してください。

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

STM32G4 シリーズのマイクロコントローラは、ST が特許を保有する最先端テクノロジーを搭載しています。

関連ドキュメント

- Cortex®-M4 テクニカルリファレンスマニュアル（http://infocenter.arm.com から入手可能）。
- STM32G4xx データシート
- STM32G4xx 正誤表
- STM32F3、STM32F4、STM32G4、および STM32L4 シリーズの Cortex®-M4 プログラミング·マニュアル（PM0214）

目次

1 このマニュアルにおける表記の規則 72

1.1 一般情報 72

1.2 レジスタに関する略記 72

1.3 用語 73

1.4 製品カテゴリの定義 73

1.5 使用可能なペリフェラル 74

2 システムおよびメモリの概要 77

2.1 システムアーキテクチャ 77

2.1.1 I- バス 78

2.1.2 D- バス 78

2.1.3 S- バス 78

2.1.4 DMA バス 79

2.1.5 バスマトリックス 79

2.2 メモリ構成 80

2.2.1 概要 80

2.2.2 メモリマップとレジスタ境界アドレス 81

2.3 ビットバンディング 85

2.4 内蔵 SRAM 86

2.4.1 パリティチェック 87

2.4.2 CCM SRAM 書込み保護 87

2.4.3 CCM SRAM 読出し保護 88

2.4.4 CCM SRAM の消去 88

2.5 Flash メモリの概要 89

2.6 ブート設定 89

2.6.1 ブート設定 89

3 内蔵 Flash メモリ（Flash）：カテゴリ 3 デバイスの場合 92

3.1 概要 92

3.2 Flash の主な機能 92

3.3 Flash の機能説明 93

3.3.1 Flash メモリの構成 93

3.3.2 エラーコード訂正（ECC） 95

3.3.3 読出しアクセスの遅延 96
3.3.4 適応型リアルタイム･メモリ･アクセラレータ（ART Accelerator） 98
3.3.5 Flash のプログラムおよび消去操作 100
3.3.6 Flash メインメモリの消去シーケンス 101
3.3.7 Flash メインメモリのプログラミングシーケンス 102
3.3.8 デュアル･バンク･モード（DBANK = 1）でのみ使用可能な書込み中読出し（RWW） 106
3.4 Flash オプションバイト 107
3.4.1 オプションバイトの説明 107
3.4.2 オプション･バイトのプログラミング 115
3.5 Flash メモリの保護 118
3.5.1 読出し保護（RDP） 118
3.5.2 商用コード読出し保護（Proprietary code readout protection: PCROP） 121
3.5.3 書込み保護（WRP） 123
3.5.4 セキュリティ保護可能な領域 124
3.5.5 コア･デバッグ･アクセスの無効化 125
3.5.6 Flash メモリからの強制ブート 125
3.6 Flash 割込み 126
3.7 Flash レジスタ 127
3.7.1 Flash アクセス制御レジスタ（FLASH_ACR） 127
3.7.2 Flash パワーダウンキーレジスタ（Flash_PDKEYR） 128
3.7.3 Flash キー レジスタ（FLASH_KEYR） 129
3.7.4 Flash オプションキーレジスタ（Flash_OPTKEYR） 129
3.7.5 Flash ステータスレジスタ（Flash_SR） 130
3.7.6 Flash 制御レジスタ（Flash_CR） 132
3.7.7 Flash ECC レジスタ（Flash_ECCR） 134
3.7.8 Flash オプション･レジスタ（FLASH_OPTR） 136
3.7.9 Flash PCROP1 開始アドレスレジスタ（Flash_PCROP1SR） 138
3.7.10 Flash PCROP1 終了アドレスレジスタ（Flash_PCROP1ER） 138
3.7.11 Flash バンク 1 WRP 領域 A アドレスレジスタ（Flash_WRP1AR） 139
3.7.12 Flash バンク 1 WRP 領域 B アドレスレジスタ（Flash_WRP1BR） 140
3.7.13 Flash PCROP2 開始アドレスレジスタ（Flash_PCROP2SR） 140
3.7.14 Flash PCROP2 終了アドレスレジスタ（Flash_PCROP2ER） 141
3.7.15 Flash バンク 2 WRP 領域 A アドレスレジスタ（Flash_WRP2AR） 141
3.7.16 Flash バンク 2 WRP 領域 B アドレスレジスタ（Flash_WRP2BR） 142
3.7.17 Flash セキュリティ保護可能領域バンク 1 レジスタ（FLASH_SEC1R） 142
3.7.18 Flash セキュリティ保護可能領域バンク 2 レジスタ（FLASH_SEC2R） 143
3.7.19 Flash レジスタマップ 144

4 内蔵 Flash メモリ（Flash）カテゴリ 4 デバイスの場合 146
4.1 概要 146
4.2 Flash の主な機能 146
4.3 Flash の機能説明 146
4.3.1 Flash メモリの構成 146
4.3.2 エラーコード訂正（ECC） 147
4.3.3 読出しアクセスの遅延 148
4.3.4 適応型リアルタイム･メモリ･アクセラレータ（ART Accelerator） 149
4.3.5 Flash のプログラムおよび消去操作 151
4.3.6 Flash メインメモリの消去シーケンス 152
4.3.7 Flash メインメモリのプログラミングシーケンス 153
4.4 Flash オプションバイト 157
4.4.1 オプションバイトの説明 157
4.4.2 オプション･バイトのプログラミング 162
4.5 Flash メモリの保護 163
4.5.1 読出し保護（RDP） 163
4.5.2 独自仕様コード読出し保護（Proprietary code readout protection: PCROP） 166
4.5.3 書込み保護（WRP） 167
4.5.4 セキュリティ保護可能な領域 168
4.5.5 コア･デバッグ･アクセスの無効化 168
4.5.6 Flash メモリからの強制ブート 169
4.6 Flash 割込み 169
4.7 Flash レジスタ 170
4.7.1 Flash アクセス制御レジスタ（FLASH_ACR） 170
4.7.2 Flash パワーダウンキーレジスタ（Flash_PDKEYR） 171
4.7.3 Flash キー レジスタ（FLASH_KEYR） 172
4.7.4 Flash オプションキーレジスタ（Flash_OPTKEYR） 172
4.7.5 Flash ステータスレジスタ（Flash_SR） 173
4.7.6 Flash 制御レジスタ（Flash_CR） 175
4.7.7 Flash ECC レジスタ（Flash_ECCR） 177
4.7.8 Flash オプションレジスタ（Flash_OPTR） 178
4.7.9 Flash PCROP1 開始アドレスレジスタ（Flash_PCROP1SR） 180
4.7.10 Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER） 180
4.7.11 Flash WRP 領域 A アドレス･レジスタ（FLASH_WRP1AR） 181
4.7.12 Flash WRP 領域 B アドレス･レジスタ（FLASH_WRP1BR） 181

4.7.13 Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）.....182
4.7.14 Flash レジスタマップ 183

5 内蔵 Flash メモリ（Flash）カテゴリ 2 デバイスの場合 185

5.1 概要 185
5.2 Flash の主な機能 185
5.3 Flash の機能説明 185
5.3.1 Flash メモリの構成 185
5.3.2 エラーコード訂正（ECC）.....186
5.3.3 読出しアクセスの遅延 187
5.3.4 適応型リアルタイム･メモリ･アクセラレータ（ART Accelerator）.....188
5.3.5 Flash のプログラムおよび消去操作 190
5.3.6 Flash メインメモリの消去シーケンス **191**
5.3.7 Flash メイン･メモリのプログラミング･シーケンス 192
5.4 Flash オプションバイト 196
5.4.1 オプションバイトの説明 196
5.4.2 オプションバイトのプログラミング 201
5.5 Flash メモリの保護 202
5.5.1 読出し保護（RDP）.....202
5.5.2 独自仕様コード読出し保護（PCROP）.....205
5.5.3 書込み保護（WRP）.....206
5.5.4 セキュリティ保護可能な領域 207
5.5.5 コア･デバッグ･アクセスの無効化 207
5.5.6 Flash メモリからの強制ブート 208
5.6 Flash 割込み 208
5.7 Flash レジスタ 209
5.7.1 Flash アクセス制御レジスタ（FLASH_ACR）.....209
5.7.2 Flash パワーダウンキーレジスタ（Flash_PDKEYR）.....210
5.7.3 Flash キー レジスタ（FLASH_KEYR）.....211
5.7.4 Flash オプションキーレジスタ（Flash_OPTKEYR）.....211
5.7.5 Flash ステータスレジスタ（Flash_SR）.....212
5.7.6 Flash 制御レジスタ（Flash_CR）.....214
5.7.7 Flash ECC レジスタ（Flash_ECCR）.....216
5.7.8 Flash オプションレジスタ（Flash_OPTR）.....217
5.7.9 Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）.....219
5.7.10 Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）.....219

5.7.11 Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）.....220
5.7.12 Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）.....220
5.7.13 Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）.....221
5.7.14 Flash レジスタマップ.....222

6 電源制御（PWR）.....224

6.1 電源.....224
6.1.1 独立したアナログペリフェラルの電源.....225
6.1.2 USB トランシーバ電源.....226
6.1.3 バッテリバックアップドメイン.....226
6.1.4 電圧レギュレータ.....227
6.1.5 ダイナミック電圧スケーリングの管理.....228
6.2 電源供給スーパバイザ.....230
6.2.1 パワーオンリセット（POR）／パワーダウンリセット（PDR）／ブラウンアウトリセット（BOR）.....230
6.2.2 プログラム可能な電圧検出器（PVD）.....231
6.2.3 ペリフェラル電圧モニタ（PVM）.....231
6.3 低消費電力モード.....232
6.3.1 RUN モード.....237
6.3.2 低消費電力 RUN モード（LP RUN）.....237
6.3.3 低消費電力モード.....238
6.3.4 SLEEP モード.....239
6.3.5 低消費電力 SLEEP モード（LP SLEEP）.....240
6.3.6 STOP 0 モード.....241
6.3.7 STOP 1 モード.....243
6.3.8 STANDBY モード.....244
6.3.9 SHUTDOWN モード.....247
6.3.10 低消費電力モードからの自動ウェイクアップ.....248
6.4 PWR レジスタ.....249
6.4.1 電源制御レジスタ 1（PWR_CR1）.....249
6.4.2 電源制御レジスタ 2（PWR_CR2）.....250
6.4.3 電源制御レジスタ 3（PWR_CR3）.....251
6.4.4 電源制御レジスタ 4（PWR_CR4）.....252
6.4.5 電源ステータスレジスタ 1（PWR_SR1）.....253
6.4.6 電源ステータスレジスタ 2（PWR_SR2）.....254
6.4.7 電源ステータスクリアレジスタ（PWR_SCR）.....256
6.4.8 電源ポート A プルアップ制御レジスタ（PWR_PUCRA）.....256

6.4.9 電源ポート A プルダウン制御レジスタ（PWR_PDCRA）. 257
6.4.10 電源ポート B プルアップ制御レジスタ（PWR_PUCRB）. 258
6.4.11 電源ポート B プルダウン制御レジスタ（PWR_PDCRB）. 258
6.4.12 電源ポート C プルアップ制御レジスタ（PWR_PUCRC）. 259
6.4.13 電源ポート C プルダウン制御レジスタ（PWR_PDCRC）. 259
6.4.14 電源ポート D プルアップ制御レジスタ（PWR_PUCRD）. 260
6.4.15 電源ポート D プルダウン制御レジスタ（PWR_PDCRD）. 260
6.4.16 電源ポート E プルアップ制御レジスタ（PWR_PUCRE）. 261
6.4.17 電源ポート E プルダウン制御レジスタ（PWR_PDCRE）. 261
6.4.18 電源ポート F プルアップ制御レジスタ（PWR_PUCRF）. 262
6.4.19 電源ポート F プルダウン制御レジスタ（PWR_PDCRF）. 262
6.4.20 電源ポート G プルアップ制御レジスタ（PWR_PUCRG）. 263
6.4.21 電源ポート G プルダウン制御レジスタ（PWR_PDCRG）. 263
6.4.22 電源制御レジスタ （PWR_CR5）. 264
6.4.23 PWR レジスタマップとリセット値の表 265

7 リセットおよびクロック制御（RCC）. 267

7.1 リセット 267
7.1.1 電源リセット 267
7.1.2 システムリセット 267
7.1.3 RTC ドメインリセット 269
7.2 クロック 269
7.2.1 HSE クロック 273
7.2.2 HSI16 クロック 274
7.2.3 HSI48 クロック 275
7.2.4 PLL 275
7.2.5 LSE クロック 276
7.2.6 LSI クロック 276
7.2.7 システムクロック（SYSCLK）の選択 276
7.2.8 クロックソースの周波数と電圧スケーリング 277
7.2.9 クロックセキュリティシステム（CSS）. 277
7.2.10 LSE のクロックセキュリティシステム 278
7.2.11 ADC クロック 278
7.2.12 RTC クロック 278
7.2.13 タイマクロック 279
7.2.14 ウォッチドッグクロック 279
7.2.15 クロック信号出力 279

7.2.16 TIM5/TIM15/TIM16/TIM17 を使用した内部／外部クロックの測定 280
7.2.17 ペリフェラル･クロック有効化レジスタ（RCC_AHBxENR、RCC_APBxENRy）. 282
7.3 低電力モード 283
7.4 RCC レジスタ 284
7.4.1 クロック制御レジスタ（RCC_CR） 284
7.4.2 内部クロックソース較正レジスタ（RCC_ICSCR） 286
7.4.3 クロック設定レジスタ（RCC_CFGR）. 286
7.4.4 PLL 設定レジスタ（RCC_PLLCFGR） 288
7.4.5 クロック割込み有効レジスタ（RCC_CIER） 291
7.4.6 クロック割込みフラグ･レジスタ（RCC_CIFR）. 292
7.4.7 クロック割込みクリアレジスタ（RCC_CICR） 294
7.4.8 AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）. 295
7.4.9 AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）. 296
7.4.10 AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）. 298
7.4.11 APB1 ペリフェラル･リセット･レジスタ 1（RCC_APB1RSTR1）. 298
7.4.12 APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）. 301
7.4.13 APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR） 301
7.4.14 AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1ENR） 303
7.4.15 AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2ENR） 304
7.4.16 AHB3 ペリフェラル･クロック有効化レジスタ（RCC_AHB3ENR）. 306
7.4.17 APB1 ペリフェラル･クロック有効化レジスタ 1（RCC_APB1ENR1） 306
7.4.18 APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1ENR2） 309
7.4.19 APB2 ペリフェラルクロック有効レジスタ（RCC_APB2ENR）. 310
7.4.20 SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）. 311
7.4.21 SLEEP モードおよび STOP モードにおける AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2SMENR）. 313
7.4.22 SLEEP モードおよび STOP モードにおける AHB3 ペリフェラル･クロック有効化レジスタ（RCC_AHB3SMENR） 315
7.4.23 SLEEP モードおよび STOP モードにおける APB1 ペリフェラル･クロック有効化レジスタ 1（RCC_APB1SMENR1）. 316
7.4.24 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1SMENR2）. 319
7.4.25 SLEEP モードおよび STOP モードにおける APB2 ペリフェラルクロック有効レジスタ（RCC_APB2SMENR） 320
7.4.26 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR） 322
7.4.27 RTC ドメイン制御レジスタ（RCC_BDCR）. 326
7.4.28 制御／ステータスレジスタ（RCC_CSR）. 328

7.4.29 クロックリカバリ RC レジスタ（RCC_CRRCR）.....330
7.4.30 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR2）.....331
7.4.31 RCC レジスタマップ.....332

8 クロック･リカバリ･システム（CRS）.....336
8.1 概要.....336
8.2 CRS の主な機能.....336
8.3 CRS の実装.....336
8.4 CRS の機能詳細.....337
8.4.1 CRS ブロック図.....337
8.4.2 同期入力.....337
8.4.3 周波数誤差測定.....338
8.4.4 周波数誤差評価および自動トリミング.....338
8.4.5 CRS の初期化と設定.....339
8.5 CRS 低電力モード.....340
8.6 CRS 割込み.....340
8.7 CRS レジスタ.....341
8.7.1 CRS 制御レジスタ（CRS_CR）.....341
8.7.2 CRS 設定レジスタ（CRS_CFGR）.....342
8.7.3 CRS 割込みおよびステータスレジスタ（CRS_ISR）.....343
8.7.4 CRS 割込みフラグクリアレジスタ（CRS_ICR）.....345
8.7.5 CRS レジスタマップ.....346

9 汎用 I/O（GPIO）.....347
9.1 概要.....347
9.2 GPIO の主な機能.....347
9.3 GPIO の機能説明.....347
9.3.1 汎用 I/O（GPIO）.....350
9.3.2 I/O ピンオルタネート機能マルチプレクサと配置.....350
9.3.3 I/O ポート制御レジスタ.....351
9.3.4 I/O ポートデータレジスタ.....351
9.3.5 I/O データのビット単位の操作.....351
9.3.6 GPIO ロック機構.....352
9.3.7 I/O オルタネート機能の入力／出力.....352
9.3.8 外部割込み／ウェイクアップ･ライン.....352
9.3.9 入力設定.....353

9.3.10 出力設定 353
9.3.11 オルタネート機能設定 354
9.3.12 アナログ設定 355
9.3.13 HSE または LSE オシレータのピンを GPIO として使用 355
9.3.14 GPIO ピンを RTC 供給ドメインで使用 355
9.3.15 PB8 を GPIO として使用 356
9.3.16 PG10 を GPIO として使用 356
9.4 GPIO レジスタ 357
9.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x = A ~ G） 357
9.4.2 GPIO ポート出力タイプ･レジスタ（GPIOx_OTYPER）（x = A ~ G） 357
9.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ~ G） 358
9.4.4 GPIO ポート･プルアップ／プルダウン･レジスタ（GPIOx_PUPDR）（x = A ~ G） 358
9.4.5 GPIO ポート入力データ･レジスタ（GPIOx_IDR）（x = A ~ G） 359
9.4.6 GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A ~ G） 359
9.4.7 GPIO ポート･ビット･セット／リセット･レジスタ（GPIOx_BSRR）（x = A ~ G） 360
9.4.8 GPIO ポート設定ロック･レジスタ（GPIOx_LCKR）（x = A ~ G） 361
9.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A ~ G） 362
9.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ~ G） 363
9.4.11 GPIO ポート･ビット･リセット･レジスタ（GPIOx_BRR）（x = A ~ G） 363
9.4.12 GPIO レジスタマップ 364

10 システム設定コントローラ（SYSCFG） 366
10.1 SYSCFG の主な機能 366
10.2 SYSCFG レジスタ 366
10.2.1 SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP） 366
10.2.2 SYSCFG 設定レジスタ 1（SYSCFG_CFGR1） 367
10.2.3 SYSCFG 外部割込み設定レジスタ 1（SYSCFG_EXTICR1） 369
10.2.4 SYSCFG 外部割込み設定レジスタ 2（SYSCFG_EXTICR2） 370
10.2.5 SYSCFG 外部割込み設定レジスタ 3（SYSCFG_EXTICR3） 371
10.2.6 SYSCFG 外部割込み設定レジスタ 4（SYSCFG_EXTICR4） 373
10.2.7 SYSCFG CCM SRAM 制御およびステータスレジスタ（SYSCFG_SCSR） . 374
10.2.8 SYSCFG 設定レジスタ 2（SYSCFG_CFGR2） 375
10.2.9 SYSCFG CCM SRAM 書込み保護レジスタ（SYSCFG_SWPR） 376
10.2.10 SYSCFG CCM SRAM キーレジスタ（SYSCFG_SKR） 376
10.2.11 SYSCFG レジスタマップ 377

11 ペリフェラル相互接続マトリックス 378

11.1 概要 378

11.2 接続の一覧 379

11.3 相互接続の詳細 381

11.3.1 タイマ（TIMx、HRTIM）からタイマ（TIMx）へ 381

11.3.2 タイマ（TIMx、HRTIM）および EXTI から ADC（ADCx）へ 383

11.3.3 ADC（ADCx）からタイマ（TIMx、HRTIM）へ 384

11.3.4 タイマ（TIMx、HRTIM）および EXTI から DAC（DACx）へ 385

11.3.5 HSE、LSE、LSI、HSI16、MCO、RTC からタイマ（TIMx）へ 386

11.3.6 RTC、COMPx から低電力タイマ（LPTIM1）へ 386

11.3.7 タイマ（TIMx）からコンパレータ（COMPx）へ 387

11.3.8 内部アナログ･ソースから ADC（ADCx）、コンパレータ（COMPx）、OPAMP（OPAMPx）へ 387

11.3.9 コンパレータ（COMPx）からタイマ（TIMx、HRTIM）へ 391

11.3.10 システムエラーからタイマ（TIMx、HRTIM）および HRTIM へ 396

11.3.11 タイマ（TIM16/TIM17）から IRTIM へ 396

12 ダイレクト･メモリ･アクセス（DMA）コントローラ 397

12.1 概要 397

12.2 DMA の主な機能 397

12.3 DMA の実装 398

12.3.1 DMA1 および DMA2 398

12.3.2 DMA リクエストマッピング 398

12.4 DMA の機能詳細 398

12.4.1 DMA のブロック図 398

12.4.2 DMA ピンおよび内部信号 400

12.4.3 DMA 転送 400

12.4.4 DMA アービトレーション 401

12.4.5 DMA チャネル 401

12.4.6 DMA データの幅、アライメント、およびエンディアン 405

12.4.7 DMA エラー管理 406

12.5 DMA 割込み 407

12.6 DMA レジスタ 407

12.6.1 DMA 割込みステータスレジスタ（DMA_ISR） 407

12.6.2 DMA 割込みフラグクリアレジスタ（DMA_IFCR） 410

12.6.3 DMA チャネル x 設定レジスタ（DMA_CCRx） 411

12.6.4 DMA チャネル x 転送データ数レジスタ（DMA_CNDTRx）.....414
12.6.5 DMA チャネル x ペリフェラルアドレスレジスタ（DMA_CPARx）.....414
12.6.6 DMA チャネル x メモリアドレスレジスタ（DMA_CMARx）.....415
12.6.7 DMA レジスタ･マップ.....416

13 DMA リクエスト･マルチプレクサ（DMAMUX）..... 418
13.1 概要.....418
13.2 DMAMUX の主な機能.....419
13.3 DMAMUX の実装.....419
13.3.1 DMAMUX の構成.....419
13.3.2 DMAMUX のマッピング.....420
13.4 DMAMUX の機能詳細.....422
13.4.1 DMAMUX のブロック図.....422
13.4.2 DMAMUX 信号.....423
13.4.3 DMAMUX チャネル.....423
13.4.4 DMAMUX リクエストラインマルチプレクサ.....423
13.4.5 DMAMUX リクエストジェネレータ.....426
13.5 DMAMUX 割込み.....427
13.6 DMAMUX レジスタ.....428
13.6.1 DMAMUX リクエスト･ライン･マルチプレクサ･チャネル x 設定レジスタ（DMAMUX_CxCR）.....428
13.6.2 DMAMUX リクエストラインマルチプレクサ割込みチャネルステータスレジスタ（DMAMUX_CSR）.....429
13.6.3 DMAMUX リクエストラインマルチプレクサ割込みクリアフラグレジスタ（DMAMUX_CFR）.....429
13.6.4 DMAMUX リクエスト･ジェネレータ･チャネル x 設定レジスタ（DMAMUX_RGxCR）.....430
13.6.5 DMAMUX リクエストジェネレータ割込みステータスレジスタ（DMAMUX_RGSR）.....431
13.6.6 DMAMUX リクエスト･ジェネレータ割込みクリア･フラグ･レジスタ（DMAMUX_RGCFR）.....431
13.6.7 DMAMUX レジスタ･マップ.....432

14 ネスト化されたベクタ割込みコントローラ（NVIC）..... 434
14.1 NVIC の主な機能.....434
14.2 SysTick 較正値レジスタ.....434
14.3 割込みベクタと例外ベクタ.....435

15 拡張割込み／イベント･コントローラ（EXTI） 439
15.1 概要 439
15.2 EXTI の主な機能 439
15.3 EXTI の機能詳細 439
15.3.1 EXTI ブロック図 440
15.3.2 ウェイクアップイベント管理 440
15.3.3 ペリフェラル非同期割込み 441
15.3.4 ハードウェア割込みの選択 441
15.3.5 ハードウェアイベントの選択 441
15.3.6 ソフトウェア割込み／イベントの選択 441
15.4 EXTI 割込み／イベントラインの配置 442
15.5 EXTI レジスタ 444
15.5.1 割込みマスクレジスタ 1（EXTI_IMR1） 444
15.5.2 イベントマスクレジスタ 1（EXTI_EMR1） 444
15.5.3 立ち上がりトリガ選択レジスタ 1（EXTI_RTSR1） 445
15.5.4 立ち下がりトリガ選択レジスタ 1（EXTI_FTSR1） 445
15.5.5 ソフトウェア割込みイベントレジスタ 1（EXTI_SWIER1） 446
15.5.6 ペンディングレジスタ 1（EXTI_PR1） 447
15.5.7 割込みマスクレジスタ 2（EXTI_IMR2） 448
15.5.8 イベントマスクレジスタ 2（EXTI_EMR2） 448
15.5.9 立ち上がりトリガ選択レジスタ 2（EXTI_RTSR2） 449
15.5.10 立ち下がりトリガ選択レジスタ 2（EXTI_FTSR2） 450
15.5.11 ソフトウェア割込みイベントレジスタ 2（EXTI_SWIER2） 450
15.5.12 ペンディングレジスタ 2（EXTI_PR2） 451
15.5.13 EXTI レジスタマップ 452

16 巡回冗長検査計算ユニット（CRC） 453
16.1 概要 453
16.2 CRC の主な機能 453
16.3 CRC の機能説明 454
16.3.1 CRC ブロック図 454
16.3.2 CRC 内部信号 454
16.3.3 CRC 操作 454
16.4 CRC レジスタ 456
16.4.1 CRC データ･レジスタ（CRC_DR） 456
16.4.2 CRC 独立型データレジスタ (CRC_IDR) 456

16.4.3 CRC 制御レジスタ（CRC_CR）.....457
16.4.4 CRC の初期値（CRC_INIT）.....458
16.4.5 CRC 多項式（CRC_POL）.....458
16.4.6 CRC レジスタマップ.....459

17 CORDIC コプロセッサ（CORDIC）..... 460
17.1 CORDIC の概要.....460
17.2 CORDIC の主な機能.....460
17.3 CORDIC の機能詳細.....460
17.3.1 概要.....460
17.3.2 CORDIC 関数.....460
17.3.3 固定小数点表現.....467
17.3.4 スケーリングファクタ.....467
17.3.5 精度.....467
17.3.6 ゼロオーバーヘッドモード.....471
17.3.7 ポーリングモード.....472
17.3.8 割込みモード.....473
17.3.9 DMA モード.....473
17.4 CORDIC レジスタ.....474
17.4.1 CORDIC 制御 / ステータス･レジスタ（CORDIC_CSR）.....474
17.4.2 CORDIC 引数レジスタ（CORDIC_WDATA）.....476
17.4.3 CORDIC 結果レジスタ（CORDIC_RDATA）.....477
17.4.4 CORDIC レジスタマップ.....477

18 フィルタ演算アクセラレータ（FMAC）..... 478
18.1 FMAC の概要.....478
18.2 FMAC の主な機能.....478
18.3 FMAC の機能詳細.....479
18.3.1 概要.....479
18.3.2 ローカルメモリとバッファ.....480
18.3.3 入力バッファ.....481
18.3.4 出力バッファ.....483
18.3.5 初期化機能.....485
18.3.6 フィルタ機能.....486
18.3.7 固定小数点表現.....490
18.3.8 FMAC を用いた FIR フィルタの実装.....490

18.3.9 FMAC を用いた IIR フィルタの実装 492
18.3.10 フィルタ初期化の例 494
18.3.11 フィルタ演算の例 495
18.3.12 フィルタ設計のヒント 497
18.4 FMAC レジスタ 498
18.4.1 FMAC X1 バッファ設定レジスタ（FMAC_X1BUFCFG） 498
18.4.2 FMAC X2 バッファ設定レジスタ（FMAC_X2BUFCFG） 498
18.4.3 FMAC Y バッファ設定レジスタ（FMAC_YBUFCFG） 499
18.4.4 FMAC パラメータ･レジスタ（FMAC_PARAM） 500
18.4.5 FMAC 制御レジスタ（FMAC_CR） 501
18.4.6 FMAC ステータス･レジスタ（FMAC_SR） 502
18.4.7 FMAC ライト･データ･レジスタ（FMAC_WDATA） 504
18.4.8 FMAC 読出しデータ･レジスタ（FMAC_RDATA） 504
18.4.9 FMAC レジスタマップ 505

19 フレキシブル･スタティック･メモリ･コントローラ（FSMC） 506

19.1 概要 506
19.2 FMC の主な特徴 506
19.3 FMC ブロック図 507
19.4 AHB インタフェース 508
19.4.1 サポートされるメモリおよびトランザクション 508
19.5 外部デバイスアドレスマッピング 509
19.5.1 NOR/PSRAM アドレスマッピング 510
19.5.2 NAND Flash メモリ･アドレス･マッピング 511
19.6 NOR 型 Flash ／ PSRAM コントローラ 512
19.6.1 外部メモリインタフェース信号 513
19.6.2 サポートされるメモリおよびトランザクション 515
19.6.3 一般的なタイミング規則 516
19.6.4 NOR Flash/PSRAM コントローラ非同期トランザクション 517
19.6.5 同期トランザクション 534
19.6.6 NOR/PSRAM コントローラレジスタ 540
19.7 NAND 型 Flash コントローラ 548
19.7.1 外部メモリインタフェース信号 548
19.7.2 NAND 型 Flash によってサポートされるメモリとトランザクション 549
19.7.3 NAND Flash メモリのタイミング図 550
19.7.4 NAND Flash 動作 551

19.7.5 NAND Flash のプリウェイト機能 552
19.7.6 エラー訂正コード（ECC）の計算 NAND 型 Flash メモリ 553
19.7.7 NAND Flash コントローラレジスタ 554
19.7.8 FMC レジスタ･マップ 559

20 Quad SPI インタフェース（QUADSPI） 561

20.1 概要 561
20.2 QUADSPI の主な機能 561
20.3 QUADSPI の機能説明 561
20.3.1 QUADSPI ブロック図 561
20.3.2 QUADSPI ピン 562
20.3.3 QUADSPI コマンドシーケンス 563
20.3.4 QUADSPI シングルインタフェースプロトコルモード 565
20.3.5 QUADSPI インダイレクトモード 567
20.3.6 QUADSPI ステータス･フラグのポーリング･モード 568
20.3.7 QUADSPI メモリマップドモード 569
20.3.8 QUADSPI Flash メモリの設定 570
20.3.9 QUADSPI 遅延データサンプリング 570
20.3.10 QUADSPI の設定 570
20.3.11 QUADSPI の使用 571
20.3.12 1 回限りの命令の送信 573
20.3.13 QUADSPI エラー管理 573
20.3.14 QUADSPI の BUSY ビットおよびアボート機能 573
20.3.15 nCS の動作 574
20.4 QUADSPI 割込み 575
20.5 QUADSPI レジスタ 576
20.5.1 QUADSPI 制御レジスタ（QUADSPI_CR） 576
20.5.2 QUADSPI デバイス設定レジスタ（QUADSPI_DCR） 579
20.5.3 QUADSPI ステータスレジスタ（QUADSPI_SR） 580
20.5.4 QUADSPI フラグ･クリア･レジスタ（QUADSPI_FCR） 581
20.5.5 QUADSPI データ長レジスタ（QUADSPI_DLR） 582
20.5.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR） 582
20.5.7 QUADSPI アドレスレジスタ（QUADSPI_AR） 585
20.5.8 QUADSPI オルタネート･バイト･レジスタ（QUADSPI_ABR） 585
20.5.9 QUADSPI データレジスタ（QUADSPI_DR） 586
20.5.10 QUADSPI ポーリング･ステータス･マスク･レジスタ（QUADSPI _PSMKR） 587

20.5.11 QUADSPI ポーリングステータス一致レジスタ（QUADSPI _PSMAR）.....587
20.5.12 QUADSPI ポーリング･インターバル･レジスタ（QUADSPI _PIR）.....588
20.5.13 QUADSPI 低消費電力タイムアウトレジスタ（QUADSPI_LPTR）.....588
20.5.14 QUADSPI レジスタマップ 589

21 A/D コンバータ（ADC）.....590

21.1 概要 590
21.2 ADC の主な機能 591
21.3 A/D コンバータの実装 592
21.4 ADC の機能詳細 593
21.4.1 ADC ブロック図 593
21.4.2 A/D コンバータのピンおよび内部信号 594
21.4.3 A/D コンバータのクロック 595
21.4.4 ADC1/2/3/4/5 のコネクティビティ 597
21.4.5 スレーブ AHB インタフェース 602
21.4.6 ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）.....602
21.4.7 シングルエンド入力および差動入力のチャネル 602
21.4.8 較正（ADCAL、ADCALDIF、ADC_CALFACT）.....603
21.4.9 ADC オン / オフ制御（ADEN、ADDIS、ADRDY）.....606
21.4.10 A/D コンバータ制御ビット書込み時の制約 607
21.4.11 チャネルの選択（SQRx、JSQRx）.....608
21.4.12 チャネル単位でプログラム可能なサンプリング時間（SMPR1、SMPR2）.....609
21.4.13 シングル変換モード（CONT = 0）.....611
21.4.14 連続変換モード（CONT = 1）.....611
21.4.15 変換の開始（ADSTART、JADSTART）.....612
21.4.16 ADC のタイミング 613
21.4.17 実行中の変換の停止（ADSTP、JADSTP）.....613
21.4.18 外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN、JEXTSEL、JEXTEN）.....615
21.4.19 インジェクトチャネルの管理 620
21.4.20 不連続モード（DISCEN、DISCNUM、JDISCEN）.....621
21.4.21 インジェクト変換のコンテキストのキュー 623
21.4.22 プログラム可能な分解能（RES）- 高速変換モード 630
21.4.23 変換の終了、サンプリングフェーズの終了（EOC、JEOC、EOSMP）.....631
21.4.24 変換シーケンスの終了（EOS、JEOS）.....631
21.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）.....632

21.4.26 データ管理 633
21.4.27 動的低電力機能 640
21.4.28 アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx） 644
21.4.29 オーバーサンプリング回路 648
21.4.30 デュアル ADC モード 654
21.4.31 温度センサ 668
21.4.32 VBAT 電源監視 670
21.4.33 内部電圧基準の監視 671
21.5 低電力モードでの A/D コンバータ 672
21.6 A/D コンバータの割込み 673
21.7 ADC レジスタ（ADC ごと） 674
21.7.1 A/D コンバータの割込みおよびステータス･レジスタ（ADC_ISR） 674
21.7.2 ADC 割込み有効レジスタ（ADC_IER） 676
21.7.3 A/D コンバータ制御レジスタ（ADC_CR） 678
21.7.4 A/D コンバータ設定レジスタ（ADC_CFGR） 681
21.7.5 ADC 設定レジスタ 2（ADC_CFGR2） 685
21.7.6 A/D コンバータ･サンプル時間レジスタ 1（ADC_SMPR1） 687
21.7.7 A/D コンバータ･サンプル時間レジスタ 2（ADC_SMPR2） 689
21.7.8 A/D コンバータ･ウォッチドッグ閾値レジスタ 1（ADC_TR1） 690
21.7.9 A/D コンバータ･ウォッチドッグ閾値レジスタ 2（ADC_TR2） 691
21.7.10 A/D コンバータ･ウォッチドッグ閾値レジスタ 3（ADC_TR3） 692
21.7.11 A/D コンバータ･レギュラ･シーケンス･レジスタ 1（ADC_SQR1） 692
21.7.12 A/D コンバータ･レギュラ･シーケンス･レジスタ 2（ADC_SQR2） 693
21.7.13 A/D コンバータ･レギュラ･シーケンス･レジスタ 3（ADC_SQR3） 694
21.7.14 A/D コンバータ･レギュラ･シーケンス･レジスタ 4（ADC_SQR4） 695
21.7.15 A/D コンバータ･レギュラ･データ･レジスタ（ADC_DR） 696
21.7.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR） 696
21.7.17 A/D コンバータ･オフセット y レジスタ（ADC_OFRy） 698
21.7.18 ADC インジェクトチャネル y データレジスタ（ADC_JDRy） 699
21.7.19 A/D コンバータ･アナログ･ウォッチドッグ 2 設定レジスタ（ADC_AWD2CR） 700
21.7.20 A/D コンバータ･アナログ･ウォッチドッグ 3 設定レジスタ（ADC_AWD3CR） 701
21.7.21 A/D コンバータ差動モード選択レジスタ（ADC_DIFSEL） 701
21.7.22 A/D コンバータ較正係数（ADC_CALFACT） 702
21.7.23 A/D コンバータ･ゲイン補正レジスタ（ADC_GCOMP） 703

21.8 ADC 共通レジスタ 704
21.8.1 A/D コンバータ x 共通ステータス･レジスタ（ADCx_CSR）（x = 12 または 345） 704
21.8.2 A/D コンバータ x 共通制御レジスタ（ADCx_CCR）（x = 12 または 345） . . . 705
21.8.3 A/D コンバータ x デュアル･モード用共通レギュラ･データ･レジスタ（ADCx_CDR）（x = 12 または 345） 709
21.9 A/D コンバータ･レジスタ･マップ 710

22 D/A コンバータ（DAC） 713
22.1 概要 713
22.2 DAC の主な機能 713
22.3 D/A コンバータの実装 714
22.4 D/A コンバータの機能詳細 715
22.4.1 D/A コンバータのブロック図 715
22.4.2 DAC ピンおよび内部信号 716
22.4.3 D/A コンバータチャネルイネーブル 721
22.4.4 DAC データフォーマット 721
22.4.5 D/A コンバータによる変換 723
22.4.6 D/A コンバータ出力電圧 724
22.4.7 DAC トリガ選択 724
22.4.8 DMA リクエスト 724
22.4.9 ノイズ生成 725
22.4.10 三角波生成 727
22.4.11 D/A コンバータによるのこぎり波生成 728
22.4.12 DAC チャネルモード 729
22.4.13 DAC 出力バッファ較正 732
22.4.14 デュアル D/A コンバータチャネル変換モード（デュアルチャネルを使用できる場合） 734
22.5 低電力モードでの D/A コンバータ 739
22.6 D/A コンバータの割込み 739
22.7 DAC レジスタ 740
22.7.1 DAC 制御レジスタ（DAC_CR） 740
22.7.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR） 743
22.7.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1） 744
22.7.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1） 745
22.7.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1） . . . 745

22.7.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2）.....746
22.7.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2）.....746
22.7.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2）...747
22.7.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD）....747
22.7.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD）....748
22.7.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD）.....748
22.7.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）.....749
22.7.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）.....749
22.7.14 DAC ステータスレジスタ（DAC_SR）.....750
22.7.15 DAC 較正制御レジスタ（DAC_CCR）.....751
22.7.16 DAC モード制御レジスタ（DAC_MCR）.....752
22.7.17 D/A コンバータ･チャネル 1 サンプル･アンド･ホールドのサンプル時間レジスタ（DAC_SHSR1）.....754
22.7.18 DAC チャネル 2 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2）.....754
22.7.19 DAC サンプルおよびホールド時間レジスタ（DAC_SHHR）.....755
22.7.20 D/A コンバータ･サンプル･アンド･ホールドのリフレッシュ時間レジスタ（DAC_SHRR）.....755
22.7.21 D/A コンバータ･チャネル 1 のこぎり波レジスタ（DAC_STR1）.....756
22.7.22 DAC チャネル 2 のこぎり波レジスタ（DAC_STR2）.....757
22.7.23 DAC のこぎり波モードレジスタ（DAC_STMODR）.....757
22.7.24 D/A コンバータ･レジスタ･マップ.....759

23 電圧基準バッファ（VREFBUF）.....761

23.1 概要.....761
23.2 VREFBUF の機能説明.....761
23.3 VREFBUF のトリミング.....762
23.4 VREFBUF レジスタ.....762
23.4.1 VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR）.....762
23.4.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）.....763
23.4.3 VREFBUF レジスタ･マップ.....763

24 コンパレータ（COMP）.....764

24.1 COMP の概要.....764
24.2 COMP の主な機能.....764
24.3 COMP の機能詳細.....764

24.3.1 COMP ブロック図 764
24.3.2 COMP ピンおよび内部信号 765
24.3.3 COMP のリセットおよびクロック 765
24.3.4 COMP ロック機構 766
24.3.5 COMP ヒステリシス 766
24.3.6 COMP 出力のブランキング 767
24.4 COMP 低電力モード 768
24.5 COMP 割込み 768
24.6 COMP レジスタ 769
24.6.1 コンパレータ x 制御 / ステータスレジスタ（COMP_CxCSR） 769
24.6.2 COMP レジスタマップ 771

25 オペアンプ（OPAMP） 772
25.1 概要 772
25.2 OPAMP の主な機能 772
25.3 OPAMP の機能詳細 772
25.3.1 内部 ADC チャネルへの OPAMP 出力リダイレクション 772
25.3.2 OPAMP のリセットおよびクロック 773
25.3.3 初期設定 773
25.3.4 信号接続 773
25.3.5 OPAMP モード 775
25.3.6 OPAMP PGA ゲイン 780
25.3.7 較正 781
25.3.8 タイマによって制御されたマルチプレクサ･モード 782
25.4 OPAMP 低電力モード 783
25.5 OPAMP レジスタ 784
25.5.1 OPAMP1 制御／ステータス･レジスタ（OPAMP1_CSR） 784
25.5.2 OPAMP2 制御／ステータス･レジスタ（OPAMP2_CSR） 787
25.5.3 OPAMP3 制御／ステータス･レジスタ（OPAMP3_CSR） 790
25.5.4 OPAMP4 制御／ステータス･レジスタ（OPAMP4_CSR） 793
25.5.5 OPAMP5 制御／ステータス･レジスタ（OPAMP5_CSR） 796
25.5.6 OPAMP6 制御／ステータス･レジスタ（OPAMP6_CSR） 799
25.5.7 OPAMP1 タイマ制御モードレジスタ（OPAMPx_TCMR）（x = 1...6） 802
25.5.8 OPAMP2 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6） 803
25.5.9 OPAMP3 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6） 804
25.5.10 OPAMP4 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6） 806

25.5.11 OPAMP5 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6） 807
25.5.12 OPAMP6 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6） 808
25.5.13 OPAMP レジスタ･マップ 810

26 真性乱数発生器（RNG） 812

26.1 概要 812
26.2 RNG の主な機能 812
26.3 RNG の機能説明 813
26.3.1 RNG のブロック図 813
26.3.2 RNG 内部信号 813
26.3.3 乱数の生成 814
26.3.4 RNG 初期化 816
26.3.5 RNG 動作 817
26.3.6 RNG クロック供給 818
26.3.7 エラー管理 818
26.3.8 RNG の低消費電力時の取り扱い 819
26.4 RNG 割込み 819
26.5 RNG 処理時間 820
26.6 RNG エントロピー･ソース検証 820
26.6.1 概要 820
26.6.2 検証条件 820
26.6.3 データ収集 820
26.7 RNG レジスタ 820
26.7.1 RNG 制御レジスタ（RNG_CR） 820
26.7.2 RNG ステータスレジスタ（RNG_SR） 822
26.7.3 RNG データ･レジスタ（RNG_DR） 823
26.7.4 RNG レジスタ･マップ 823

27 高分解能タイマ（HRTIM） 824

27.1 概要 824
27.2 主な特徴 825
27.3 機能詳細 826
27.3.1 概要 826
27.3.2 HRTIM ピンおよび内部信号 828
27.3.3 クロック 831
27.3.4 タイマ A ～ F のタイミングユニット 835

27.3.5 マスタタイマ 857
27.3.6 アップダウンカウントモード 858
27.3.7 セット／リセットイベントの優先順位と狭いパルスの管理 865
27.3.8 外部イベントのグローバル調整 868
27.3.9 タイミングユニットでの外部イベントのフィルタリング 872
27.3.10 遅延保護 878
27.3.11 レジスタのプリロードおよび更新管理 884
27.3.12 「より大きい」比較を備えた PWM モード 888
27.3.13 複数タイマ内またはそれらをまたがったイベントの伝播 889
27.3.14 出力管理 894
27.3.15 バーストモードコントローラ 896
27.3.16 チョッパ 904
27.3.17 フォールト保護 905
27.3.18 補助出力 910
27.3.19 他のタイマまたは HRTIM インスタンスとの HRTIM の同期 913
27.3.20 ADC トリガ 916
27.3.21 DAC トリガ 920
27.3.22 割込み 925
27.3.23 DMA 927
27.3.24 HRTIM の初期化 930
27.3.25 デバッグ 931
27.4 アプリケーションの使用事例 932
27.4.1 降圧コンバータ 932
27.4.2 同期整流式の降圧コンバータ 933
27.4.3 多相コンバータ 934
27.4.4 遷移モード力率補正 935
27.5 HRTIM レジスタ 938
27.5.1 HRTIM マスタ･タイマ制御レジスタ（HRTIM_MCR） 938
27.5.2 HRTIM マスタタイマ割込みステータスレジスタ（HRTIM_MISR） 941
27.5.3 HRTIM マスタタイマ割込みクリアレジスタ（HRTIM_MICR） 942
27.5.4 HRTIM マスタ･タイマ DMA ／割込みイネーブル･レジスタ（HRTIM_MDIER） 943
27.5.5 HRTIM マスタタイマカウンタレジスタ（HRTIM_MCNTR） 945
27.5.6 HRTIM マスタタイマ周期レジスタ（HRTIM_MPER） 945
27.5.7 HRTIM マスタタイマ繰り返しレジスタ（HRTIM_MREP） 946
27.5.8 HRTIM マスタタイマ比較 1 レジスタ（HRTIM_MCMP1R） 946
27.5.9 HRTIM マスタタイマ比較 2 レジスタ（HRTIM_MCMP2R） 947

27.5.10 HRTIM マスタタイマ比較 3 レジスタ（HRTIM_MCMP3R）.....947
27.5.11 HRTIM マスタタイマ比較 4 レジスタ（HRTIM_MCMP4R）.....948
27.5.12 HRTIM タイマ x 制御レジスタ（HRTIM_TIMxCR）（x = A ～ F）.....949
27.5.13 HRTIM タイマ x 割込みステータス･レジスタ（HRTIM_TIMxISR）（x = A ~ F）.....953
27.5.14 HRTIM タイマ x 割込みクリア･レジスタ（HRTIM_TIMxICR）（x = A ~ F）...955
27.5.15 HRTIM タイマ x DMA 割込みイネーブル･レジスタ（HRTIM_TIMxDIER）（x = A ~ F）.....956
27.5.16 HRTIM タイマ x カウンタレジスタ（HRTIM_CNTxR）（x = A ～ F）.....959
27.5.17 HRTIM タイマ x 周期レジスタ（HRTIM_PERxR）（x = A ～ F）.....960
27.5.18 HRTIM タイマ x 繰り返しレジスタ（HRTIM_REPxR）（x = A ～ F）.....961
27.5.19 HRTIM タイマ x 比較 1 レジスタ（HRTIM_CMP1xR）（x = A ～ F）.....962
27.5.20 HRTIM タイマ x 比較 1 合成レジスタ（HRTIM_CMP1CxR）（x = A ～ F）...963
27.5.21 HRTIM タイマ x 比較 2 レジスタ（HRTIM_CMP2xR）（x = A ～ F）.....964
27.5.22 HRTIM タイマ x 比較 3 レジスタ（HRTIM_CMP3xR）（x = A ～ F）.....965
27.5.23 HRTIM タイマ x 比較 4 レジスタ（HRTIM_CMP4xR）（x = A ～ F）.....966
27.5.24 HRTIM タイマ x キャプチャ 1 レジスタ（HRTIM_CPT1xR）（x = A ～ F）...967
27.5.25 HRTIM タイマ x キャプチャ 2 レジスタ（HRTIM_CPT2xR）（x = A ～ F）...968
27.5.26 HRTIM タイマ x デッドタイム･レジスタ（HRTIM_DTxR）（x = A ~ F）.....969
27.5.27 HRTIM タイマ x 出力 1 設定レジスタ（HRTIM_SETx1R）（x = A ～ F）.....971
27.5.28 HRTIM タイマ x 出力 1 リセット･レジスタ（HRTIM_RSTx1R）（x = A ~ F）.....973
27.5.29 HRTIM タイマ x 出力 2 設定レジスタ（HRTIM_SETx2R）（x = A ～ F）.....973
27.5.30 HRTIM タイマ x 出力 2 リセットレジスタ（HRTIM_RSTx2R）（x = A ～ F）.....974
27.5.31 HRTIM タイマ x 外部イベントフィルタリングレジスタ 1（HRTIM_EEFxR1）（x = A ～ F）.....975
27.5.32 HRTIM タイマ x 外部イベントフィルタリングレジスタ 2（HRTIM_EEFxR2）（x = A ～ F）.....977
27.5.33 HRTIM タイマ A リセットレジスタ（HRTIM_RSTAR）.....978
27.5.34 HRTIM タイマ B リセットレジスタ（HRTIM_RSTBR）.....980
27.5.35 HRTIM タイマ C リセットレジスタ（HRTIM_RSTCR）.....982
27.5.36 HRTIM タイマ D リセットレジスタ（HRTIM_RSTDR）.....984
27.5.37 HRTIM タイマ E リセットレジスタ（HRTIM_RSTER）.....986
27.5.38 HRTIM タイマ F リセットレジスタ（HRTIM_RSTFR）.....988
27.5.39 HRTIM タイマ x チョッパレジスタ（HRTIM_CHPxR）（x = A ～ F）.....990
27.5.40 HRTIM タイマ A キャプチャ 1 制御レジスタ（HRTIM_CPT1ACR）.....991
27.5.41 HRTIM タイマ B キャプチャ 1 制御レジスタ（HRTIM_CPT1BCR）.....991
27.5.42 HRTIM タイマ C キャプチャ 1 制御レジスタ（HRTIM_CPT1CCR）.....991

27.5.43 HRTIM タイマ D キャプチャ 1 制御レジスタ（HRTIM_CPT1DCR）.....992
27.5.44 HRTIM タイマ E キャプチャ 1 制御レジスタ（HRTIM_CPT1ECR）.....992
27.5.45 HRTIM タイマ F キャプチャ 1 制御レジスタ（HRTIM_CPT1FCR）.....992
27.5.46 HRTIM タイマ x キャプチャ 2 制御レジスタ（HRTIM_CPT2xCR）（x = A ～ F）.....993
27.5.47 HRTIM タイマ x 出力レジスタ（HRTIM_OUTxR）（x = A ～ F）.....998
27.5.48 HRTIM タイマ x フォールトレジスタ（HRTIM_FLTxR）（x = A ～ F）.....1001
27.5.49 HRTIM タイマ x 制御レジスタ 2（HRTIM_TIMxCR2）（x = A ～ F）.....1002
27.5.50 HRTIM タイマ x 外部イベントフィルタリングレジスタ 3（HRTIM_TIMxEEFR3）（x = A ～ F）.....1005
27.5.51 HRTIM 制御レジスタ 1（HRTIM_CR1）.....1006
27.5.52 HRTIM 制御レジスタ 2（HRTIM_CR2）.....1007
27.5.53 HRTIM 割込みステータスレジスタ（HRTIM_ISR）.....1009
27.5.54 HRTIM 割込みクリアスレジスタ（HRTIM_ICR）.....1010
27.5.55 HRTIM 割込み有効レジスタ（HRTIM_IER）.....1011
27.5.56 HRTIM 出力有効レジスタ（HRTIM_OENR）.....1012
27.5.57 HRTIM 出力無効レジスタ（HRTIM_ODISR）.....1013
27.5.58 HRTIM 出力無効ステータスレジスタ（HRTIM_ODSR）.....1014
27.5.59 HRTIM バーストモード制御レジスタ（HRTIM_BMCR）.....1015
27.5.60 HRTIM バーストモードトリガレジスタ（HRTIM_BMTRGR）.....1017
27.5.61 HRTIM バーストモード比較レジスタ（HRTIM_BMCMPR）.....1019
27.5.62 HRTIM バーストモード周期レジスタ（HRTIM_BMPER）.....1019
27.5.63 HRTIM タイマ外部イベント制御レジスタ 1（HRTIM_EECR1）.....1020
27.5.64 HRTIM タイマ外部イベント制御レジスタ 2（HRTIM_EECR2）.....1022
27.5.65 HRTIM タイマ外部イベント制御レジスタ 3（HRTIM_EECR3）.....1023
27.5.66 HRTIM ADC トリガ 1 レジスタ（HRTIM_ADC1R）.....1024
27.5.67 HRTIM ADC トリガ 2 レジスタ（HRTIM_ADC2R）.....1025
27.5.68 HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R）.....1025
27.5.69 HRTIM ADC トリガ 4 レジスタ（HRTIM_ADC4R）.....1027
27.5.70 HRTIM DLL 制御レジスタ（HRTIM_DLLCR）.....1029
27.5.71 HRTIM フォールト入力レジスタ 1（HRTIM_FLTINR1）.....1030
27.5.72 HRTIM フォールト入力レジスタ 2（HRTIM_FLTINR2）.....1031
27.5.73 HRTIM バースト DMA マスタタイマ更新レジスタ（HRTIM_BDMUPR）...1033
27.5.74 HRTIM バースト DMA タイマ x 更新レジスタ（HRTIM_BDTxUPR）（x = A ～ F）.....1034
27.5.75 HRTIM バースト DMA データレジスタ（HRTIM_BDMADR）.....1035
27.5.76 HRTIM ADC 拡張トリガ･レジスタ（HRTIM_ADCER）.....1036
27.5.77 HRTIM ADC トリガ更新レジスタ（HRTIM_ADCUR）.....1039

27.5.78 HRTIM ADC ポスト･スケーラ･レジスタ 1（HRTIM_ADCPS1）.....1040
27.5.79 HRTIM ADC ポストスケーラレジスタ 2（HRTIM_ADCPS2）.....1041
27.5.80 HRTIM フォールト入力レジスタ 3（HRTIM_FLTINR3）.....1042
27.5.81 HRTIM フォールト入力レジスタ 4（HRTIM_FLTINR4）.....1044
27.5.82 HRTIM レジスタマップ 1045

28 高機能制御タイマ（TIM1/TIM8/TIM20）..... 1055

28.1 TIM1/TIM8/TIM20 概要 1055
28.2 TIM1/TIM8/TIM20 の主な特徴 1055
28.3 TIM1/TIM8/TIM20 機能詳細 1056
28.3.1 ブロック図 1056
28.3.2 TIM1/TIM8/TIM20 ピンおよび内部信号 1057
28.3.3 タイムベース･ユニット 1062
28.3.4 カウンタモード 1064
28.3.5 繰り返しカウンタ 1076
28.3.6 外部トリガ入力 1078
28.3.7 クロック選択 1078
28.3.8 キャプチャ／比較チャネル 1082
28.3.9 入力キャプチャモード 1085
28.3.10 PWM 入力モード 1086
28.3.11 強制出力モード 1087
28.3.12 出力比較モード 1087
28.3.13 PWM モード 1088
28.3.14 非対称 PWM モード 1097
28.3.15 組み合わせ PWM モード 1098
28.3.16 組み合わせ 3 相 PWM モード 1099
28.3.17 相補出力とデッドタイム挿入 1100
28.3.18 ブレーク機能の使用 1103
28.3.19 双方向ブレーク入力 1109
28.3.20 外部イベントによる tim_ocxref 信号のクリア 1110
28.3.21 6 ステップ PWM 生成 1112
28.3.22 ワンパルスモード 1113
28.3.23 再トリガ可能なワンパルス･モード 1114
28.3.24 比較モードでのパルス 1115
28.3.25 エンコーダインタフェースモード 1117
28.3.26 方向ビットの出力 1135
28.3.27 UIF ビットの再配置 1135

28.3.28 タイマ入力 XOR 機能 1135
28.3.29 ホールセンサとのインタフェース 1136
28.3.30 タイマの同期 1137
28.3.31 ADC トリガ 1141
28.3.32 DMA バースト・モード 1142
28.3.33 TIM1/TIM8/TIM20DMA リクエスト 1143
28.3.34 デバッグ・モード 1143
28.4 TIM1/TIM8/TIM20 低電力モード 1143
28.5 TIM1/TIM8/TIM20 割込み 1144
28.6 TIM1/TIM8/TIM20 レジスタ 1145
28.6.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 1、8、20） 1145
28.6.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 1、8、20） 1147
28.6.3 TIMx スレーブ・モード制御レジスタ（TIMx_SMCR）（x = 1、8、20） 1150
28.6.4 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 1、8、20） 1154
28.6.5 TIMx ステータスレジスタ（TIMx_SR）（x = 1、8、20） 1155
28.6.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 1、8、20） 1158
28.6.7 TIMx キャプチャ／比較モード・レジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 1、8、20） 1159
28.6.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 1、8、20） 1161
28.6.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 1、8、20） 1163
28.6.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 1、8、20） 1164
28.6.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8、20） 1167
28.6.12 TIMx カウンタ（TIMx_CNT）（x = 1、8、20） 1171
28.6.13 TIMx プリスケーラ（TIMx_PSC）（x = 1、8、20） 1171
28.6.14 TIMx 自動再ロード・レジスタ（TIMx_ARR）（x = 1、8、20） 1172
28.6.15 TIMx 繰り返しカウンタ・レジスタ（TIMx_RCR）（x = 1、8、20） 1172
28.6.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）（x = 1、8、20） 1173
28.6.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）（x = 1、8、20） 1174
28.6.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）（x = 1、8、20） 1175
28.6.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）（x = 1、8、20） 1176
28.6.20 TIMx ブレークおよびデッドタイム・レジスタ（TIMx_BDTR）（x = 1、8、20） 1177
28.6.21 TIMx キャプチャ／比較レジスタ 5 （TIMx_CCR5）（x = 1、8、20） 1181
28.6.22 TIMx キャプチャ／比較レジスタ 6 （TIMx_CCR6）（x = 1、8、20） 1182

28.6.23 TIMx キャプチャ／比較モード･レジスタ 3（TIMx_CCMR3）（x = 1、8、20）.....1183
28.6.24 TIMx タイマ･デッドタイム･レジスタ 2 （TIMx_DTR2）（x = 1、8、20）...1184
28.6.25 TIMx タイマ･エンコーダ制御レジスタ（TIMx_ECR）（x = 1、8、20）.....1185
28.6.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 1、8、20）.....1186
28.6.27 TIMx オルタネート機能オプション･レジスタ 1 （TIMx_AF1）（x = 1、8、20）.....1187
28.6.28 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 1、8、20）.....1190
28.6.29 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 1、8、20）.....1193
28.6.30 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 1、8、20）.....1194
28.6.31 TIMx レジスタマップ 1195

29 汎用タイマ（TIM2/TIM3/TIM4/TIM5）..... 1198

29.1 TIM2/TIM3/TIM4/TIM5 の概要 1198
29.2 TIM2/TIM3/TIM4/TIM5 の主な特徴 1198
29.3 TIM2/TIM3/TIM4/TIM5 の実装 1199
29.4 TIM2/TIM3/TIM4/TIM5 機能詳細 1200
29.4.1 ブロック図 1200
29.4.2 TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号 1201
29.4.3 タイムベース･ユニット 1205
29.4.4 カウンタモード 1207
29.4.5 クロック選択 1217
29.4.6 キャプチャ／比較チャネル 1221
29.4.7 入力キャプチャモード 1223
29.4.8 PWM 入力モード 1224
29.4.9 強制出力モード 1225
29.4.10 出力比較モード 1225
29.4.11 PWM モード 1227
29.4.12 非対称 PWM モード 1235
29.4.13 組み合わせ PWM モード 1236
29.4.14 外部イベントによる tim_ocxref 信号のクリア 1237
29.4.15 ワンパルスモード 1239
29.4.16 再トリガ可能なワンパルス･モード 1240
29.4.17 比較モードでのパルス 1241
29.4.18 エンコーダインタフェースモード 1243
29.4.19 方向ビットの出力 1261
29.4.20 UIF ビットの再配置 1261

29.4.21 タイマ入力 XOR 機能 1261
29.4.22 タイマと外部トリガの同期 1261
29.4.23 タイマの同期 1266
29.4.24 ADC トリガ 1271
29.4.25 DMA バースト･モード 1271
29.4.26 TIM2/TIM3/TIM4/TIM5DMA リクエスト 1272
29.4.27 デバッグ･モード 1273
29.4.28 TIM2/TIM3/TIM4/TIM5 低電力モード 1273
29.4.29 TIM2/TIM3/TIM4/TIM5 割込み 1273
29.5 TIM2/TIM3/TIM4/TIM5 レジスタ 1274
29.5.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 2 〜 5） 1274
29.5.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 2 〜 5） 1276
29.5.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）（x = 2 〜 5） 1277
29.5.4 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 2 〜 5） 1280
29.5.5 TIMx ステータスレジスタ（TIMx_SR）（x = 2 〜 5） 1281
29.5.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 2 〜 5） 1283
29.5.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 2 〜 5） 1284
29.5.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 2 〜 5） 1286
29.5.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 2 〜 5） 1288
29.5.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 2 〜 5） 1289
29.5.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 2 〜 5） 1290
29.5.12 TIMx カウンタ（TIMx_CNT）（x = 3、4） 1292
29.5.13 TIMx カウンタ（TIMx_CNT）（x = 2、5） 1292
29.5.14 TIMx プリスケーラ（TIMx_PSC）（x = 2 〜 5） 1293
29.5.15 TIMx 自動再ロード･レジスタ（TIMx_ARR）（x = 3、4） 1293
29.5.16 TIMx 自動再ロード･レジスタ（TIMx_ARR）（x = 2、5） 1294
29.5.17 TIMx キャプチャ / 比較レジスタ 1（TIMx_CCR1）（x = 3、4） 1295
29.5.18 TIMx キャプチャ / 比較レジスタ 1（TIMx_CCR1）（x = 2、5） 1296
29.5.19 TIMx キャプチャ / 比較レジスタ 2（TIMx_CCR2）（x = 3、4） 1297
29.5.20 TIMx キャプチャ / 比較レジスタ 2（TIMx_CCR2）（x = 2、5） 1298
29.5.21 TIMx キャプチャ / 比較レジスタ 3（TIMx_CCR3）（x = 3、4） 1299
29.5.22 TIMx キャプチャ / 比較レジスタ 3（TIMx_CCR3）（x = 2、5） 1300
29.5.23 TIMx キャプチャ / 比較レジスタ 4（TIMx_CCR4）（x = 3、4） 1301
29.5.24 TIMx キャプチャ / 比較レジスタ 4（TIMx_CCR4）（x = 2、5） 1302

29.5.25 TIMx タイマエンコーダ制御レジスタ（TIMx_ECR）（x = 2 ～ 5）.....1303
29.5.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 2 ～ 5）.....1304
29.5.27 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 2 ～ 5）.....1305
29.5.28 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 2 ～ 5）.....1306
29.5.29 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 2 ～ 5）.....1307
29.5.30 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 2 ～ 5）.....1308
29.5.31 TIMx レジスタマップ 1309

30 汎用タイマ（TIM15/TIM16/TIM17） 1311

30.1 TIM15/TIM16/TIM17 概要 1311
30.2 TIM15 の主な機能 1311
30.3 TIM16/TIM17 の主な特徴 1312
30.4 TIM15/TIM16/TIM17 機能詳細 1313
30.4.1 ブロック図 1313
30.4.2 TIM15/TIM16/TIM17 のピンと内部信号 1314
30.4.3 タイムベース･ユニット 1317
30.4.4 カウンタモード 1319
30.4.5 繰り返しカウンタ 1323
30.4.6 クロック選択 1324
30.4.7 キャプチャ／比較チャネル 1326
30.4.8 入力キャプチャモード 1329
30.4.9 PWM 入力モード（TIM15 の場合のみ） 1330
30.4.10 強制出力モード 1331
30.4.11 出力比較モード 1331
30.4.12 PWM モード 1333
30.4.13 組み合わせ PWM モード（TIM15 のみ） 1338
30.4.14 相補出力とデッドタイム挿入 1339
30.4.15 ブレーク機能の使用 1342
30.4.16 双方向ブレーク入力 1346
30.4.17 外部イベントによる tim_ocxref 信号のクリア 1348
30.4.18 6 ステップ PWM 生成 1349
30.4.19 ワンパルスモード 1350
30.4.20 再トリガ可能なワンパルス･モード（OPM）（TIM15 のみ） 1351
30.4.21 UIF ビットの再配置 1352
30.4.22 タイマ入力 XOR 機能（TIM15 のみ） 1352
30.4.23 外部トリガ同期（TIM15 のみ） 1353

30.4.24 スレーブ･モード - リセット･モードとトリガ･モードの組み合わせ（TIM15 のみ）.....1355
30.4.25 スレーブモード - リセットモードとゲートモードの組み合わせ（TIM15 のみ）.....1355
30.4.26 タイマ同期（TIM15 のみ）.....1356
30.4.27 タイマ出力を他のタイマのトリガとして使用する（TIM16/TIM17 のみ）.....1356
30.4.28 ADC トリガ（TMI15 のみ）.....1356
30.4.29 DMA バースト･モード.....1356
30.4.30 TIM15/TIM16/TIM17DMA リクエスト.....1357
30.4.31 デバッグ･モード.....1357
30.5 TIM15/TIM16/TIM17 低電力モード.....1358
30.6 TIM15/TIM16/TIM17 割込み.....1358
30.7 TIM15 レジスタ.....1359
30.7.1 TIM15 制御レジスタ 1（TIM15_CR1）.....1359
30.7.2 TIM15 制御レジスタ 2（TIM15_CR2）.....1360
30.7.3 TIM15 スレーブモード制御レジスタ（TIM15_SMCR）.....1362
30.7.4 TIM15 DMA ／割込み有効化レジスタ（TIM15_DIER）.....1363
30.7.5 TIM15 ステータスレジスタ（TIM15_SR）.....1364
30.7.6 TIM15 イベント生成レジスタ（TIM15_EGR）.....1366
30.7.7 TIM15 キャプチャ／比較モードレジスタ 1 [オルタネート]（TIM15_CCMR1）.....1367
30.7.8 TIM15 キャプチャ／比較モード･レジスタ 1 [オルタネート]（TIM15_CCMR1）.....1369
30.7.9 TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）.....1371
30.7.10 TIM15 カウンタ（TIM15_CNT）.....1374
30.7.11 TIM15 プリスケーラ（TIM15_PSC）.....1374
30.7.12 TIM15 の自動再ロードレジスタ（TIM15_ARR）.....1375
30.7.13 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）.....1375
30.7.14 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1）.....1376
30.7.15 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2）.....1377
30.7.16 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）.....1377
30.7.17 TIM15 タイマデッドタイムレジスタ 2（TIM15_DTR2）.....1380
30.7.18 TIM15 入力選択レジスタ（TIM15_TISEL）.....1381
30.7.19 TIM15 オルタネート機能レジスタ 1 （TIM15_AF1）.....1382
30.7.20 TIM15 オルタネート機能レジスタ 2 （TIM15_AF2）.....1384
30.7.21 TIM15 DMA 制御レジスタ（TIM15_DCR）.....1385
30.7.22 完全転送の TIM15 DMA アドレス（TIM15_DMAR）.....1385
30.7.23 TIM15 レジスタマップ.....1386

30.8 TIM16/TIM17 レジスタ 1388
30.8.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 16 〜 17）. 1388
30.8.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 16 〜 17）. 1389
30.8.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 16 〜 17）. 1390
30.8.4 TIMx ステータスレジスタ（TIMx_SR）（x = 16 〜 17）. 1391
30.8.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 16 〜 17）. 1392
30.8.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 16 〜 17）. 1393
30.8.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 16 〜 17）. 1394
30.8.8 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16 〜 17）. 1396
30.8.9 TIMx カウンタ（TIMx_CNT）（x = 16 〜 17）. 1399
30.8.10 TIMx プリスケーラ（TIMx_PSC）（x = 16 〜 17）. 1399
30.8.11 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 16 〜 17）. 1400
30.8.12 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）（x = 16 〜 17）. 1400
30.8.13 TIMx キャプチャ／比較レジスタ 1（TIMx_CCR1）（x = 16 〜 17）. 1401
30.8.14 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16 〜 17）. 1402
30.8.15 TIMx タイマ･デッドタイム･レジスタ 2 （TIMx_DTR2）（x = 16 〜 17）. . . . 1405
30.8.16 TIMx 入力選択レジスタ（TIMx_TISEL）（x = 16 〜 17）. 1406
30.8.17 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 16 〜 17）. 1406
30.8.18 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 16 〜 17）. 1409
30.8.19 TIMx オプション･レジスタ 1（TIMx_OR1）（x = 16 〜 17）. 1409
30.8.20 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 16 〜 17）. 1410
30.8.21 TIM16/TIM17 完全転送用の DMA アドレス（TIMx_DMAR）（x = 16 〜 17）. 1411
30.8.22 TIM16/TIM17 レジスタマップ 1412

31 基本タイマ （TIM6/TIM7）. 1414

31.1 TIM6/TIM7 の概要 1414
31.2 TIM6/TIM7 の主な特徴 1414
31.3 TIM6/TIM7 機能詳細 1415
31.3.1 TIM6/TIM7 ブロック図 1415
31.3.2 TIM6/TIM7 内部信号 1415
31.3.3 TIM6/TIM7 クロック 1416
31.3.4 タイムベースユニット 1416
31.3.5 カウントモード 1418
31.3.6 UIF ビットの再配置 1426

31.3.7 ADC トリガ 1427
31.3.8 TIM6/TIM7DMA リクエスト 1427
31.3.9 デバッグ･モード 1427
31.3.10 TIM6/TIM7 低消費電力モード 1427
31.3.11 TIM6/TIM7 割込み 1427
31.4 TIM6/TIM7 レジスタ 1428
31.4.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 6 ~ 7） 1428
31.4.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 6 ~ 7） 1429
31.4.3 TIMx DMA ／割込み有効レジスタ（TIMx_DIER）（x = 6 ~ 7） 1430
31.4.4 TIMx ステータスレジスタ（TIMx_SR）（x = 6 ~ 7） 1430
31.4.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 6 ~ 7） 1431
31.4.6 TIMx カウンタ（TIMx_CNT）（x = 6 ~ 7） 1431
31.4.7 TIMx プリスケーラ（TIMx_PSC）（x = 6 ~ 7） 1432
31.4.8 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 6 ~ 7） 1432
31.4.9 TIMx レジスタマップ 1433

32 低電力タイマ（LPTIM） 1434
32.1 概要 1434
32.2 LPTIM の主な機能 1434
32.3 LPTIM の実装 1435
32.4 LPTIM の機能詳細 1435
32.4.1 LPTIM ブロック図 1435
32.4.2 LPTIM 入力およびトリガマッピング 1435
32.4.3 LPTIM のリセットとクロック 1436
32.4.4 グリッチフィルタ 1437
32.4.5 プリスケーラ 1438
32.4.6 トリガマルチプレクサ 1438
32.4.7 動作モード 1439
32.4.8 タイムアウト機能 1440
32.4.9 波形生成 1441
32.4.10 レジスタの更新 1442
32.4.11 カウンタモード 1442
32.4.12 タイマの有効化 1443
32.4.13 タイマカウンタのリセット 1443
32.4.14 エンコーダモード 1443
32.4.15 デバッグモード 1445

32.5 LPTIM 低電力モード 1445
32.6 LPTIM 割込み 1446
32.7 LPTIM レジスタ 1447
32.7.1 LPTIM 割込みおよびステータス·レジスタ（LPTIM_ISR）. 1447
32.7.2 LPTIM 割込みクリア·レジスタ（LPTIM_ICR）. 1448
32.7.3 LPTIM 割込み有効レジスタ（LPTIM_IER）. 1449
32.7.4 LPTIM 設定レジスタ（LPTIM_CFGR）. 1450
32.7.5 LPTIM 制御レジスタ（LPTIM_CR）. 1453
32.7.6 LPTIM 比較レジスタ（LPTIM_CMP）. 1454
32.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）. 1454
32.7.8 LPTIM カウンタレジスタ（LPTIM_CNT）. 1455
32.7.9 LPTIM オプション·レジスタ（LPTIM_OR）. 1456
32.7.10 LPTIM レジスタマップ 1457

33 赤外線インタフェース（IRTIM）. 1458

34 AES ハードウェア·アクセラレータ（AES）. 1459
34.1 概要 1459
34.2 AES の主な特徴 1459
34.3 AES の実装 1460
34.4 AES 機能詳細 1460
34.4.1 AES ブロック図 1460
34.4.2 AES 内部信号 1460
34.4.3 AES 暗号コア 1461
34.4.4 暗号操作を行うための AES の手順 1466
34.4.5 AES 復号ラウンド·キーの準備 1468
34.4.6 AES 暗号文借用（CTS）およびデータ·パディング 1469
34.4.7 AES タスクのサスペンドとレジューム 1470
34.4.8 AES 基本連鎖モード（ECB、CBC）. 1470
34.4.9 AES カウンタ（CTR）モード 1475
34.4.10 AES ガロア／カウンタモード（GCM）. 1478
34.4.11 AES ガロア·メッセージ認証コード（GMAC）. 1484
34.4.12 CBC-MAC 付き AES カウンタ（CCM）. 1485
34.4.13 AES データ·レジスタおよびデータ·スワッピング 1491
34.4.14 AES キーレジスタ 1493
34.4.15 AES 初期化ベクタ·レジスタ 1493
34.4.16 AES DMA インタフェース 1493

34.4.17 AES エラー管理 1495
34.5 AES 割込み 1495
34.6 AES 処理遅延時間 1496
34.7 AES レジスタ 1497
34.7.1 AES 制御レジスタ （AES_CR） 1497
34.7.2 AES ステータス･レジスタ（AES_SR） 1499
34.7.3 AES データ入力レジスタ（AES_DINR） 1501
34.7.4 AES データ出力レジスタ（AES_DOUTR） 1502
34.7.5 AES キー･レジスタ 0（AES_KEYR0） 1503
34.7.6 AES キー･レジスタ 1（AES_KEYR1） 1503
34.7.7 AES キー･レジスタ 2（AES_KEYR2） 1504
34.7.8 AES キー･レジスタ 3（AES_KEYR3） 1504
34.7.9 AES 初期化ベクタ･レジスタ 0（AES_IVR0） 1504
34.7.10 AES 初期化ベクタ･レジスタ 1（AES_IVR1） 1505
34.7.11 AES 初期化ベクタ･レジスタ 2（AES_IVR2） 1505
34.7.12 AES 初期化ベクタ･レジスタ 3（AES_IVR3） 1505
34.7.13 AES キー･レジスタ 4（AES_KEYR4） 1506
34.7.14 AES キー･レジスタ 5（AES_KEYR5） 1506
34.7.15 AES キー･レジスタ 6（AES_KEYR6） 1506
34.7.16 AES キー･レジスタ 7（AES_KEYR7） 1507
34.7.17 AES サスペンド･レジスタ（AES_SUSPxR） 1507
34.7.18 AES レジスタ･マップ 1508

35 リアルタイム･クロック（RTC） 1510

35.1 概要 1510
35.2 RTC の主な機能 1510
35.3 RTC の機能説明 1511
35.3.1 RTC ブロック図 1511
35.3.2 RTC ピンおよび内部信号 1512
35.3.3 RTC および TAMP によって制御される GPIO 1513
35.3.4 クロックとプリスケーラ 1514
35.3.5 リアルタイムクロックとカレンダ 1515
35.3.6 プログラム可能なアラーム 1516
35.3.7 周期的自動ウェイクアップ 1516
35.3.8 RTC の初期化と設定 1517
35.3.9 カレンダの読出し 1518

35.3.10 RTC のリセット 1519
35.3.11 RTC の同期 1520
35.3.12 RTC リファレンスクロック検出 1520
35.3.13 RTC の高精度デジタル較正 1521
35.3.14 タイムスタンプ機能 1523
35.3.15 較正クロック出力 1523
35.3.16 タンパおよびアラーム出力 1524
35.4 RTC 低電力モード 1524
35.5 RTC 割込み 1525
35.6 RTC レジスタ 1526
35.6.1 RTC 時刻レジスタ（RTC_TR） 1526
35.6.2 RTC 日付レジスタ（RTC_DR） 1527
35.6.3 RTC サブセカンドレジスタ（RTC_SSR） 1528
35.6.4 RTC 初期化制御およびステータスレジスタ（RTC_ICSR） 1528
35.6.5 RTC プリスケーラレジスタ（RTC_PRER） 1530
35.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR） 1530
35.6.7 RTC 制御レジスタ（RTC_CR） 1531
35.6.8 RTC 書込み保護レジスタ（RTC_WPR） 1534
35.6.9 RTC 較正レジスタ（RTC_CALR） 1535
35.6.10 RTC シフト制御レジスタ（RTC_SHIFTR） 1536
35.6.11 RTC タイムスタンプ時刻レジスタ（RTC_TSTR） 1537
35.6.12 RTC タイムスタンプ日付レジスタ（RTC_TSDR） 1537
35.6.13 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR） 1538
35.6.14 RTC アラーム A レジスタ（RTC_ALRMAR） 1538
35.6.15 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR） 1539
35.6.16 RTC アラーム B レジスタ（RTC_ALRMBR） 1540
35.6.17 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR） 1541
35.6.18 RTC ステータスレジスタ（RTC_SR） 1542
35.6.19 RTC マスク済み割込みステータスレジスタ（RTC_MISR） 1543
35.6.20 RTC ステータス･クリア･レジスタ（RTC_SCR） 1544
35.6.21 RTC レジスタマップ 1545

36 タンパおよびバックアップレジスタ（TAMP） 1547

36.1 概要 1547
36.2 TAMP の主な機能 1547
36.3 TAMP の機能詳細 1548

36.3.1 TAMP ブロック図 1548
36.3.2 TAMP ピンおよび内部信号 1549
36.3.3 TAMP レジスタ書込み保護 1549
36.3.4 タンパ検出 1550
36.4 TAMP の低電力モード 1552
36.5 TAMP 割込み 1552
36.6 TAMP レジスタ 1553
36.6.1 TAMP 制御レジスタ 1（TAMP_CR1） 1553
36.6.2 TAMP 制御レジスタ 2（TAMP_CR2） 1554
36.6.3 TAMP フィルタ制御レジスタ（TAMP_FLTCR） 1555
36.6.4 TAMP 割込み有効レジスタ（TAMP_IER） 1556
36.6.5 TAMP ステータスレジスタ （TAMP_SR） 1557
36.6.6 TAMP マスク済み割込みステータスレジスタ（TAMP_MISR） 1558
36.6.7 TAMP ステータスクリアレジスタ（TAMP_SCR） 1559
36.6.8 TAMP バックアップ x レジスタ（TAMP_BKPxR） 1560
36.6.9 TAMP レジスタマップ 1561

37 USART/UART（Universal synchronous / asynchronous receiver transmitter） . . . 1562

37.1 USART の概要 1562
37.2 USART の主な機能 1563
37.3 USART の拡張機能 1564
37.4 USART の実装 1564
37.5 USART の機能詳細 1565
37.5.1 USART ブロック図 1565
37.5.2 USART 信号 1566
37.5.3 USART キャラクタの説明 1567
37.5.4 USART の FIFO と閾値 1569
37.5.5 USART トランスミッタ 1569
37.5.6 USART レシーバ 1573
37.5.7 USART ボーレート生成 1580
37.5.8 クロック偏差に対する USART レシーバの許容誤差 1581
37.5.9 USART 自動ボー･レート検出 1582
37.5.10 USART マルチプロセッサ通信 1584
37.5.11 USART Modbus 通信 1586
37.5.12 USART パリティ制御 1586

37.5.13 USART LIN（Local Interconnection Network）モード 1587
37.5.14 USART 同期モード 1589
37.5.15 USART 単線半二重通信 1593
37.5.16 USART レシーバタイムアウト 1593
37.5.17 USART スマートカードモード 1594
37.5.18 USART IrDA SIR ENDEC ブロック 1598
37.5.19 USART および DMA を使用した連続通信 1600
37.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効 1602
37.5.21 USART 低電力管理 1605
37.6 低電力モードでの USART 1608
37.7 USART 割込み 1609
37.8 USART レジスタ 1611
37.8.1 USART 制御レジスタ 1[オルタネート]（USART_CR1）.....1611
37.8.2 USART 制御レジスタ 1[オルタネート]（USART_CR1）.....1615
37.8.3 USART 制御レジスタ 2（USART_CR2）.....1619
37.8.4 USART 制御レジスタ 3（USART_CR3）.....1623
37.8.5 USART ボー･レート･レジスタ（USART_BRR）.....1627
37.8.6 USART ガード時間およびプリスケーラレジスタ（USART_GTPR）.....1628
37.8.7 USART レシーバタイムアウトレジスタ（USART_RTOR）.....1629
37.8.8 USART リクエストレジスタ（USART_RQR）.....1630
37.8.9 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）.....1631
37.8.10 USART 割込みおよびステータス･レジスタ [オルタネート]（USART_ISR）.....1637
37.8.11 USART 割込みフラグクリアレジスタ（USART_ICR）.....1642
37.8.12 USART レシーバデータレジスタ（USART_RDR）.....1643
37.8.13 USART トランスミッタデータレジスタ（USART_TDR）.....1644
37.8.14 USART プリスケーラ･レジスタ（USART_PRESC）.....1645
37.8.15 USART レジスタマップ 1646

38 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）.... 1648

38.1 LPUART の概要 1648
38.2 LPUART の主な機能 1649
38.3 LPUART の実装 1650
38.4 LPUART の機能詳細 1651
38.4.1 LPUART ブロック図 1651
38.4.2 LPUART 信号 1652

38.4.3 LPUART キャラクタの説明 1652
38.4.4 LPUART の FIFO と閾値 1654
38.4.5 LPUART トランスミッタ 1654
38.4.6 LPUART レシーバ 1657
38.4.7 LPUART ボーレート生成 1661
38.4.8 クロック偏差に対する LPUART レシーバの許容誤差 1663
38.4.9 LPUART マルチプロセッサ通信 1663
38.4.10 LPUART パリティ制御 1666
38.4.11 LPUART 単線半二重通信 1667
38.4.12 DMA および LPUART を使用した連続通信 1667
38.4.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効 1670
38.4.14 LPUART 低電力管理 1672
38.5 低消費電力モードでの LPUART 1675
38.6 LPUART 割込み 1676
38.7 LPUART レジスタ 1677
38.7.1 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）. 1677
38.7.2 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）. 1680
38.7.3 LPUART 制御レジスタ 2（LPUART_CR2） 1683
38.7.4 LPUART 制御レジスタ 3（LPUART_CR3） 1685
38.7.5 LPUART ボー･レート･レジスタ（LPUART_BRR） 1688
38.7.6 LPUART リクエストレジスタ（LPUART_RQR） 1688
38.7.7 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）. . . 1689
38.7.8 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）. . . 1693
38.7.9 LPUART 割込みフラグクリアレジスタ（LPUART_ICR）. 1697
38.7.10 LPUART 受信データレジスタ（LPUART_RDR） 1698
38.7.11 LPUART 送信データレジスタ（LPUART_TDR） 1698
38.7.12 LPUART プリスケーラ･レジスタ（LPUART_PRESC） 1699
38.7.13 LPUART レジスタマップ 1700

39 シリアル･ペリフェラル･インタフェース／インター IC サウンド（SPI/I2S） 1702

39.1 概要 1702
39.2 SPI の主な機能 1702
39.3 I^2S の主な機能 1703
39.4 SPI/I2S の実装 1703
39.5 SPI の機能詳細 1704

39.5.1 概要 1704
39.5.2 マスタとスレーブの 1 対 1 の通信 1705
39.5.3 標準マルチスレーブ通信 1707
39.5.4 マルチマスタ通信 1708
39.5.5 スレーブ選択（NSS）ピンの管理 1708
39.5.6 通信フォーマット 1710
39.5.7 SPI の設定 1712
39.5.8 SPI を有効にする手順 1713
39.5.9 データの送受信手順 1713
39.5.10 SPI ステータスフラグ 1723
39.5.11 SPI エラーフラグ 1724
39.5.12 NSS パルスモード 1725
39.5.13 TI モード 1725
39.5.14 CRC 計算 1726
39.6 SPI 割込み 1728
39.7 I²S の機能詳細 1729
39.7.1 I²S の概要 1729
39.7.2 サポートされるオーディオプロトコル 1730
39.7.3 起動に関する説明 1737
39.7.4 クロックジェネレータ 1738
39.7.5 I²S マスタモード 1741
39.7.6 I²S スレーブモード 1743
39.7.7 I2S ステータスフラグ 1744
39.7.8 I²S エラーフラグ 1746
39.7.9 DMA の機能 1746
39.8 I²S 割込み 1747
39.9 SPI および I²S レジスタ 1748
39.9.1 SPI 制御レジスタ 1（SPIx_CR1） 1748
39.9.2 SPI 制御レジスタ 2（SPIx_CR2） 1750
39.9.3 SPI ステータスレジスタ（SPIx_SR） 1752
39.9.4 SPI データレジスタ（SPIx_DR） 1754
39.9.5 SPI CRC 多項式レジスタ（SPIx_CRCPR） 1754
39.9.6 SPI Rx CRC レジスタ（SPIx_RXCRCR） 1755
39.9.7 SPI Tx CRC レジスタ（SPIx_TXCRCR） 1755
39.9.8 SPIx_I2S 設定レジスタ（SPIx_I2SCFGR） 1756
39.9.9 SPIx_I2S プリスケーラレジスタ（SPIx_I2SPR） 1757
39.9.10 SPI/I2S レジスタマップ 1759

40 シリアル･オーディオ･インタフェース（SAI） 1760

40.1 概要 1760

40.2 SAI の主な機能 1760

40.3 SAI の実装 1761

40.4 SAI の機能詳細 1762

40.4.1 SAI ブロック図 1762

40.4.2 SAI ピンおよび内部信号 1763

40.4.3 主要な SAI モード 1763

40.4.4 SAI 同期モード 1764

40.4.5 オーディオデータサイズ 1765

40.4.6 フレーム同期 1765

40.4.7 スロットの設定 1768

40.4.8 SAI クロックジェネレータ 1770

40.4.9 内部 FIFO 1773

40.4.10 PDM インタフェース 1775

40.4.11 AC'97 リンク･コントローラ 1783

40.4.12 SPDIF 出力 1784

40.4.13 特有の機能 1786

40.4.14 エラーフラグ 1790

40.4.15 SAI の無効化 1793

40.4.16 SAI DMA インタフェース 1793

40.5 SAI 割込み 1794

40.6 SAI レジスタ 1796

40.6.1 SAI 設定レジスタ 1（SAI_ACR1） 1796

40.6.2 SAI 設定レジスタ 1（SAI_BCR1） 1799

40.6.3 SAI 設定レジスタ 2（SAI_ACR2） 1802

40.6.4 SAI 設定レジスタ 2（SAI_BCR2） 1804

40.6.5 SAI フレーム設定レジスタ（SAI_AFRCR） 1806

40.6.6 SAI フレーム設定レジスタ（SAI_BFRCR） 1807

40.6.7 SAI スロットレジスタ（SAI_ASLOTR） 1808

40.6.8 SAI スロットレジスタ（SAI_BSLOTR） 1809

40.6.9 SAI 割込みマスクレジスタ（SAI_AIM） 1810

40.6.10 SAI 割込みマスク･レジスタ（SAI_BIM） 1812

40.6.11 SAI ステータスレジスタ（SAI_ASR） 1813

40.6.12 SAI ステータスレジスタ（SAI_BSR） 1815

40.6.13 SAI クリアフラグレジスタ（SAI_ACLRFR） 1817

40.6.14 SAI クリアフラグレジスタ（SAI_BCLRFR）.....1818
40.6.15 SAI データレジスタ（SAI_ADR）.....1819
40.6.16 SAI データレジスタ（SAI_BDR）.....1819
40.6.17 SAI PDM 制御レジスタ（SAI_PDMCR）.....1820
40.6.18 SAI PDM ディレイレジスタ（SAI_PDMDLY）.....1821
40.6.19 SAI レジスタ･マップ.....1824

41 I²C（Inter-integrated circuit）インタフェース..... 1825

41.1 概要..... 1825
41.2 I²C の主な機能..... 1825
41.3 I²C の実装..... 1826
41.4 I²C の機能詳細..... 1826
41.4.1 I²C ブロック図.....1827
41.4.2 I²C ピンおよび内部信号.....1828
41.4.3 I²C クロックの要件.....1828
41.4.4 モード選択.....1829
41.4.5 I²C の初期化.....1830
41.4.6 ソフトウェア･リセット.....1834
41.4.7 データ転送.....1835
41.4.8 I²C スレーブモード.....1837
41.4.9 I²C マスタモード.....1846
41.4.10 I2C_TIMINGR レジスタの設定例.....1858
41.4.11 SMBus 固有の機能.....1859
41.4.12 SMBus 初期化.....1862
41.4.13 SMBus：I2C_TIMEOUTR レジスタの設定例.....1864
41.4.14 SMBus スレーブモード.....1865
41.4.15 アドレス一致時に STOP モードからウェイクアップ.....1872
41.4.16 エラー条件.....1872
41.4.17 DMA リクエスト.....1874
41.4.18 デバッグモード.....1875
41.5 I²C 低電力モード..... 1875
41.6 I²C 割込み..... 1876
41.7 I²C レジスタ..... 1877
41.7.1 I²C 制御レジスタ 1（I2C_CR1）.....1877
41.7.2 I²C 制御レジスタ 2（I2C_CR2）.....1880
41.7.3 I²C Own Address 1 レジスタ（I2C_OAR1）.....1882

41.7.4 I2C Own Address 2 レジスタ（I2C_OAR2）.....1883
41.7.5 I2C タイミングレジスタ（I2C_TIMINGR）.....1884
41.7.6 I2C タイムアウトレジスタ（I2C_TIMEOUTR）.....1885
41.7.7 I2C 割込みおよびステータスレジスタ（I2C_ISR）.....1886
41.7.8 I2C 割込みクリア･レジスタ（I2C_ICR）.....1888
41.7.9 I2C PEC レジスタ（I2C_PECR）.....1889
41.7.10 I2C 受信データレジスタ（I2C_RXDR）.....1890
41.7.11 I2C 送信データレジスタ（I2C_TXDR）.....1890
41.7.12 I2C レジスタ･マップ 1891

42 独立型ウォッチドッグ（IWDG）..... 1893

42.1 概要 1893
42.2 IWDG の主な機能 1893
42.3 IWDG の機能説明 1893
42.3.1 IWDG ブロック図 1893
42.3.2 ウィンドウオプション 1894
42.3.3 ハードウェアウォッチドッグ 1895
42.3.4 低電力モード 1895
42.3.5 レジスタのアクセス保護 1895
42.3.6 デバッグモード 1895
42.4 IWDG レジスタ 1896
42.4.1 IWDG キーレジスタ（IWDG_KR）.....1896
42.4.2 IWDG プリスケーラレジスタ（IWDG_PR）.....1897
42.4.3 IWDG 再ロードレジスタ（IWDG_RLR）.....1898
42.4.4 IWDG ステータスレジスタ（IWDG_SR）.....1899
42.4.5 IWDG ウィンドウレジスタ（IWDG_WINR）.....1900
42.4.6 IWDG レジスタマップ 1901

43 システム･ウィンドウ型ウォッチドッグ（WWDG）..... 1902

43.1 概要 1902
43.2 WWDG の主な機能 1902
43.3 WWDG の機能説明 1902
43.3.1 WWDG ブロック図 1903
43.3.2 ウォッチドッグの有効化 1903
43.3.3 ダウンカウンタの制御 1903
43.3.4 ウォッチドッグタイムアウトをプログラムする方法 1904
43.3.5 デバッグ･モード 1905

43.4 WWDG 割込み 1905
43.5 WWDG レジスタ 1905
43.5.1 WWDG 制御レジスタ（WWDG_CR）. 1906
43.5.2 WWDG 設定レジスタ（WWDG_CFR）. 1906
43.5.3 WWDG ステータスレジスタ（WWDG_SR）. 1907
43.5.4 WWDG レジスタマップ 1907

44 FD コントローラ･エリア･ネットワーク（FDCAN）. 1908
44.1 概要 1908
44.2 FDCAN の主な機能 1910
44.3 FDCAN の機能詳細 1911
44.3.1 ビット･タイミング 1912
44.3.2 動作モード 1913
44.3.3 メッセージ RAM 1921
44.3.4 FIFO 確認応答の処理 1929
44.3.5 FDCAN Rx FIFO 要素 1929
44.3.6 FDCAN の Tx バッファ要素 1932
44.3.7 FDCAN Tx イベント FIFO 要素 1933
44.3.8 FDCAN 標準メッセージ ID フィルタ要素 1935
44.3.9 FDCAN 拡張メッセージ ID フィルタ要素 1935
44.4 FDCAN レジスタ 1937
44.4.1 FDCAN コア開放レジスタ（FDCAN_CREL）. 1937
44.4.2 FDCAN エンディアン･レジスタ（FDCAN_ENDN）. 1937
44.4.3 FDCAN データ･ビット･タイミングおよびプリスケーラ･レジスタ（FDCAN_DBTP）1938
44.4.4 FDCAN テスト･レジスタ（FDCAN_TEST）. 1939
44.4.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD）. 1940
44.4.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）. 1941
44.4.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）. 1943
44.4.8 FDCAN タイムスタンプ･カウンタ設定レジスタ（FDCAN_TSCC）. 1944
44.4.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV）. 1945
44.4.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）. 1945
44.4.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）. 1946
44.4.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR）. 1947
44.4.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）. 1947
44.4.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）. 1950

44.4.15 FDCAN 割込みレジスタ（FDCAN_IR）. 1950
44.4.16 FDCAN 割込み有効レジスタ（FDCAN_IE）. 1953
44.4.17 FDCAN 割込みライン選択レジスタ（FDCAN_ILS）. 1955
44.4.18 FDCAN 割込みライン有効レジスタ（FDCAN_ILE）. 1956
44.4.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_RXGFC）. 1956
44.4.20 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）. 1958
44.4.21 FDCAN 高優先メッセージ･ステータス･レジスタ（FDCAN_HPMS）. 1958
44.4.22 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）. 1959
44.4.23 CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）. 1960
44.4.24 FDCAN Rx FIFO 1 ステータス･レジスタ（FDCAN_RXF1S）. 1960
44.4.25 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）. 1961
44.4.26 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）. 1961
44.4.27 FDCAN Tx FIFO ／キュー･ステータス･レジスタ（FDCAN_TXFQS）. 1962
44.4.28 FDCAN Tx バッファ･リクエスト･ペンディング･レジスタ（FDCAN_TXBRP）. 1963
44.4.29 FDCAN Tx バッファ追加リクエスト･レジスタ（FDCAN_TXBAR）. 1964
44.4.30 FDCAN Tx バッファ･キャンセル･リクエスト･レジスタ（FDCAN_TXBCR）. 1964
44.4.31 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）. 1965
44.4.32 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）. 1965
44.4.33 FDCAN Tx バッファ送信割込み有効レジスタ（FDCAN_TXBTIE）. 1966
44.4.34 FDCAN Tx バッファ･キャンセル終了割込みイネーブル･レジスタ（FDCAN_TXBCIE）. 1966
44.4.35 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）. 1967
44.4.36 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）. 1968
44.4.37 FDCAN CFG クロック分周回路レジスタ（FDCAN_CKDIV）. 1969
44.4.38 FDCAN レジスタ･マップ 1970

45 ユニバーサル･シリアル･バス･フルスピード･デバイス･インタフェース（USB）. 1973

45.1 概要 1973
45.2 USB の主な機能 1973
45.3 USB の実装 1973
45.4 USB の機能詳細 1974
45.4.1 USB ブロックの説明 1975
45.5 プログラミングに関する考慮事項 1976
45.5.1 汎用 USB デバイスのプログラミング 1976
45.5.2 システムリセットとパワーオンリセット 1976

45.5.3 ダブルバッファエンドポイント 1981
45.5.4 アイソクロナス転送 1984
45.5.5 サスペンド／レジュームイベント 1985
45.6 USB および USB SRAM レジスタ 1987
45.6.1 共通レジスタ 1987
45.6.2 バッファディスクリプタテーブル 2000
45.6.3 USB レジスタマップ 2004

46 USB Type-C®/USB Power Delivery インタフェース（UCPD） 2006

46.1 概要 2006
46.2 UCPD の主な特徴 2006
46.3 UCPD の実装 2007
46.4 UCPD の機能詳細 2007
46.4.1 UCPD ブロック図 2008
46.4.2 UCPD のリセットおよびクロック 2009
46.4.3 物理層のプロトコル 2010
46.4.4 UCPD BMC トランスミッタ 2016
46.4.5 UCPD BMC レシーバ 2017
46.4.6 UCPD Type-C のプルアップ抵抗（Rp）およびプルダウン抵抗（Rd） 2019
46.4.7 UCPD Type-C 電圧監視とデバウンス 2019
46.4.8 UCPD 高速ロール･スワップ（FRS）信号および検出 2020
46.4.9 UCPD DMA インタフェース 2020
46.4.10 STOP モードからのウェイクアップ 2020
46.4.11 UCPD プログラミング･シーケンス 2020
46.5 UCPD 低電力モード 2024
46.6 UCPD 割込み 2025
46.7 UCPD レジスタ 2026
46.7.1 UCPD 設定レジスタ 1（UCPD_CFGR1） 2026
46.7.2 UCPD 設定レジスタ 2（UCPD_CFGR2） 2028
46.7.3 UCPD 制御レジスタ（UCPD_CR） 2029
46.7.4 UCPD 割込みマスク･レジスタ（UCPD_IMR） 2031
46.7.5 UCPD ステータス･レジスタ（UCPD_SR） 2033
46.7.6 UCPD 割込みクリア･レジスタ（UCPD_ICR） 2036
46.7.7 UCPD Tx オーダード・セット・タイプ･レジスタ（UCPD_TX_ORDSETR） 2037
46.7.8 UCPD Tx ペイロード･サイズ･レジスタ（UCPD_TX_PAYSZR） 2038
46.7.9 UCPD Tx データ･レジスタ（UCPD_TXDR） 2038

46.7.10 UCPD Rx オーダード・セット・レジスタ（UCPD_RX_ORDSETR）.....2039
46.7.11 UCPD Rx ペイロード･サイズ･レジスタ（UCPD_RX_PAYSZR）.....2040
46.7.12 UCPD 受信データ･レジスタ（UCPD_RXDR）.....2040
46.7.13 UCPD Rx オーダード・セット拡張レジスタ 1（UCPD_RX_ORDEXTR1）.....2041
46.7.14 UCPD Rx オーダード・セット拡張レジスタ 2（UCPD_RX_ORDEXTR2）.....2041
46.7.15 UCPD レジスタ･マップ 2042

47 デバッグ･サポート（DBG）.....2044

47.1 概要 2044
47.2 Arm® リファレンス資料 2045
47.3 SWJ デバッグポート（シリアルワイヤと JTAG）.....2045
47.3.1 JTAG-DP または SW-DP の選択メカニズム 2046
47.4 ピン名とデバッグポートピン 2047
47.4.1 SWJ デバッグポートピン 2047
47.4.2 柔軟性の高い SWJ-DP ピンの割り当て 2047
47.4.3 JTAG ピンの内部プルアップ／プルダウン 2048
47.4.4 シリアル･ワイヤの使用と、未使用のデバッグ･ピンを GPIO として解放する方法 2049
47.5 STM32G4 シリーズ JTAG TAP 接続 2049
47.6 ID コードとロック機構 2051
47.6.1 MCU デバイス ID コード 2051
47.6.2 バウンダリ･スキャン TAP 2051
47.6.3 FPU 搭載 Cortex®-M4 TAP 2052
47.6.4 FPU 搭載 Cortex®-M4 JEDEC-106 ID コード 2052
47.7 JTAG デバッグポート 2052
47.8 SW デバッグポート 2054
47.8.1 SW プロトコルの概要 2054
47.8.2 SW プロトコルシーケンス 2054
47.8.3 SW-DP ステートマシン（リセット、アイドル状態、ID コード）.....2055
47.8.4 DP と AP の読出し／書込みアクセス 2055
47.8.5 SW-DP レジスタ 2056
47.8.6 SW-AP レジスタ 2057
47.9 AHB-AP（AHB アクセスポート）- JTAG-DP と SW-DP の両方に有効 2057
47.10 コアデバッグ 2058
47.11 システムリセット中のデバッガホスト接続機能 2058

47.12 FPB（Flash パッチブレークポイント） 2059
47.13 DWT（データウォッチポイントトリガ） 2059
47.14 ITM（計測トレースマクロセル） 2060
47.14.1 概要 2060
47.14.2 タイムスタンプパケット、同期およびオーバーフローパケット 2060
47.15 ETM（Embedded Trace Macrocell™） 2062
47.15.1 概要 2062
47.15.2 信号プロトコル、パケットタイプ 2062
47.15.3 主な ETM レジスタ 2062
47.15.4 設定例 2063
47.16 MCU デバッグコンポーネント（DBGMCU） 2063
47.16.1 低電力モードのデバッグサポート 2063
47.16.2 タイマ、RTC、ウォッチドッグ、および I^2C のデバッグサポート 2064
47.16.3 デバッグ MCU 設定レジスタ（DBGMCU_CR） 2064
47.16.4 デバッグ マイクロコントローラ APB1 フリーズ･レジスタ 1（DBGMCU_APB1FZR1） 2065
47.16.5 デバッグ MCU APB1 フリーズレジスタ 2（DBGMCU_APB1FZR2） 2067
47.16.6 デバッグ MCU APB2 凍結レジスタ（DBGMCU_APB2FZR） 2067
47.17 TPIU（トレースポートインタフェースユニット） 2069
47.17.1 概要 2069
47.17.2 TRACE ピンの割当て 2069
47.17.3 TPUI フォーマッタ 2071
47.17.4 TPUI フレーム同期パケット 2071
47.17.5 同期フレームパケットの送信 2072
47.17.6 同期モード 2072
47.17.7 非同期モード 2072
47.17.8 STM32G4 シリーズ内部の TRACECLKIN 接続 2072
47.17.9 TPIU レジスタ 2073
47.17.10 設定例 2073
47.18 DBG レジスタマップ 2074

48 デバイス電子署名 2075
48.1 ユニークデバイス ID レジスタ（96 ビット） 2075
48.2 Flash サイズデータレジスタ 2076
48.3 パッケージデータレジスタ 2077

49 改版履歴 2078

表の一覧

表 1. STM32G4 シリーズメモリ容量 73
表 2. 製品固有の機能 74
表 3. STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス 82
表 4. CCM SRAM の構成 87
表 5. ブート･モード 89
表 6. メモリ割当てとブートモード／物理的な再割当て 91
表 7. Flash モジュール - 512/256/128 KB のデュアル･バンク構成（読出し幅 64 ビット） 94
表 8. Flash モジュール - 512/256/128 KB のシングル･バンク構成（読出し幅 128 ビット） 95
表 9. CPU クロック（HCLK）周波数によるウェイトステート数 97
表 10. オプションバイトのフォーマット 107
表 11. オプション･バイトの構成 107
表 12. Flash メモリの読出し保護ステータス 118
表 13. アクセス状態 対 保護レベルと実行モード 120
表 14. PCROP 保護 122
表 15. WRP 保護 124
表 16. Flash 割込みリクエスト 126
表 17. Flash インタフェース - レジスタ･マップとリセット値 144
表 18. Flash モジュール - 256/512 KB の構成（読出し幅 64 ビット） 147
表 19. CPU クロック（HCLK）周波数によるウェイトステート数 148
表 20. オプションバイトのフォーマット 157
表 21. オプションバイトの構成 157
表 22. Flash メモリの読出し保護ステータス 163
表 23. アクセス状態 対 保護レベルと実行モード 165
表 24. PCROP 保護 167
表 25. WRP 保護 168
表 26. Flash 割込みリクエスト 169
表 27. Flash インタフェース - レジスタ･マップとリセット値 183
表 28. Flash モジュール - 64/128 KB の構成（読出し幅 64 ビット） 186
表 29. CPU クロック（HCLK）周波数によるウェイトステート数 187
表 30. オプションバイトのフォーマット 196
表 31. オプションバイトの構成 196
表 32. Flash メモリの読出し保護ステータス 202
表 33. アクセス状態 対 保護レベルと実行モード 204
表 34. PCROP 保護 206
表 35. WRP 保護 207
表 36. Flash 割込みリクエスト 208
表 37. Flash インタフェース - レジスタ･マップとリセット値 222
表 38. レンジ 1 ブーストモード設定 228
表 39. PVM の機能 231
表 40. 低消費電力モードの概要 234
表 41. 動作モードに応じた機能 235
表 42. 低消費電力 RUN 238
表 43. SLEEP 240
表 44. 低消費電力 SLEEP 241
表 45. STOP 0 モード 243
表 46. STOP 1 モード 244
表 47. STANDBY モード 246
表 48. SHUTDOWN モード 248
表 49. PWR レジスタマップとリセット値 265

表 50. クロックソースの周波数 277
表 51. RCC レジスタのマップとリセット値 332
表 52. CRS の機能 336
表 53. 低電力モードが CRS に与える影響 340
表 54. 割込み制御ビット 340
表 55. CRS レジスタのマップとリセット値 346
表 56. ポートビット設定表 349
表 57. GPIO レジスタのマップとリセット値 364
表 58. BOOSTEN および ANASWVDD のセット／リセット 369
表 59. SYSCFG レジスタのマップとリセット値. 377
表 60. STM32G4 シリーズペリフェラル相互接続マトリックス 379
表 61. 相互接続 1 381
表 62. 相互接続 12 381
表 63. 相互接続 13 382
表 64. 相互接続 14 382
表 65. 相互接続 19 383
表 66. 相互接続 20 385
表 67. 相互接続 17 386
表 68. 相互接続 15 387
表 69. 相互接続 21 387
表 70. 相互接続 22 388
表 71. 相互接続 23 389
表 72. 相互接続 24 390
表 73. 相互接続 2 391
表 74. 相互接続 3 392
表 75. 相互接続 4 392
表 76. 相互接続 5 393
表 77. 相互接続 6 393
表 78. 相互接続 7 393
表 79. 相互接続 8 394
表 80. 相互接続 9 394
表 81. 相互接続 10 394
表 82. 相互接続 11 395
表 83. 相互接続 18 396
表 84. 相互接続 16 396
表 85. DMA1 および DMA2 の実装 398
表 86. DMA 内部入力／出力信号 400
表 87. プログラム可能なデータ幅およびエンディアンの動作（PINC = MINC = 1 の場合） 405
表 88. DMA 割込みリクエスト 407
表 89. DMA レジスタのマップとリセット値 416
表 90. DMAMUX の構成 419
表 91. DMAMUX：リソースへのマルチプレクサ入力の割り当て 420
表 92. DMAMUX：リソースへのトリガ入力の割り当て 421
表 93. DMAMUX：リソースへの同期入力の割り当て 422
表 94. DMAMUX 信号 423
表 95. DMAMUX 割込み 427
表 96. DMAMUX レジスタのマップとリセット値 432
表 97. STM32G4 シリーズ ベクタテーブル 435
表 98. EXTI のラインの接続 443
表 99. 拡張割込み／イベントコントローラのレジスタマップとリセット値 452
表 100. CRC 内部入力／出力信号 454
表 101. CRC レジスタマップとリセット値 459

表 102. CORDIC 関数 461
表 103. 余弦パラメータ 461
表 104. 正弦パラメータ 462
表 105. 位相パラメータ 462
表 106. ベクトル絶対値のパラメータ 463
表 107. 逆正接パラメータ 463
表 108. 双曲線余弦パラメータ 464
表 109. 双曲線正弦パラメータ 464
表 110. 双曲線逆正接パラメータ 465
表 111. 自然対数パラメータ 466
表 112. 自然対数のスケーリングファクタと対応範囲 466
表 113. 平方根パラメータ 466
表 114. 平方根のスケーリング･ファクタと対応範囲 467
表 115. 精度対繰り返し回数 470
表 116. CORDIC レジスタのマップとリセット値 477
表 117. 読出し手法と書込み手法の有効な組み合わせ 491
表 118. FMAC レジスタのマップとリセット値 505
表 119. NOR/PSRAM バンク選択 510
表 120. NOR/PSRAM 外部メモリアドレス 510
表 121. NAND メモリマッピングおよびタイミングレジスタ 511
表 122. NAND バンク選択 511
表 123. プログラム可能な NOR/PSRAM のアクセスパラメータ 513
表 124. 非マルチプレクス I/O NOR Flash メモリ 513
表 125. 16 ビットマルチプレクス I/O NOR Flash メモリ 514
表 126. 非マルチプレクス I/O PSRAM/SRAM 514
表 127. 16 ビットマルチプレクス I/O PSRAM 515
表 128. NOR Flash/PSRAM：サポートされているメモリおよびトランザクションの例 515
表 129. FMC_BCRx ビットフィールド（モード 1） 518
表 130. FMC_BTRx ビットフィールド（モード 1） 519
表 131. FMC_BCRx ビットフィールド（モード A） 520
表 132. FMC_BTRx ビットフィールド（モード A） 521
表 133. FMC_BWTRx ビットフィールド（モード A） 521
表 134. FMC_BCRx ビットフィールド（モード 2/B） 523
表 135. FMC_BTRx ビットフィールド（モード 2/B） 524
表 136. FMC_BWTRx ビットフィールド（モード 2/B） 524
表 137. FMC_BCRx ビットフィールド（モード C） 526
表 138. FMC_BTRx ビットフィールド（モード C） 526
表 139. FMC_BWTRx ビットフィールド（モード C） 527
表 140. FMC_BCRx ビットフィールド（モード D） 528
表 141. FMC_BTRx ビットフィールド（モード D） 529
表 142. FMC_BWTRx ビットフィールド（モード D） 529
表 143. FMC_BCRx ビットフィールド（マルチプレクス･モード） 531
表 144. FMC_BTRx ビットフィールド（マルチプレクス･モード） 532
表 145. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード） 537
表 146. FMC_BTRx ビットフィールド（同期マルチプレクス読出しモード） 537
表 147. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード） 538
表 148. FMC_BTRx ビットフィールド（同期マルチプレクス書込みモード） 539
表 149. プログラム可能な NAND Flash のアクセスパラメータ 548
表 150. 8 ビット NAND 型 Flash 548
表 151. 16 ビット NAND 型 Flash 549
表 152. サポートされるメモリおよびトランザクション 549
表 153. ECC 結果関連ビット 558

表 154. FMC レジスタ·マップとリセット値 559
表 155. QUADSPI ピン 562
表 156. QUADSPI 割込みリクエスト 575
表 157. QUADSPI レジスタマップとリセット値 589
表 158. A/D コンバータの機能 592
表 159. A/D コンバータの内部入力信号と内部出力信号 594
表 160. A/D コンバータの各入出力ピン 594
表 161. レギュラ外部トリガのトリガ極性の設定 615
表 162. インジェクト外部トリガのトリガ極性の設定 615
表 163. ADC1/2 - レギュラ·チャネルの外部トリガ 616
表 164. ADC1/2 - インジェクトチャネルの外部トリガ 617
表 165. ADC3/4/5 - レギュラチャネルの外部トリガ 618
表 166. ADC3/4/5 - インジェクトチャネルの外部トリガ 619
表 167. TSAR タイミングは分解能に依存 631
表 168. オフセット計算対データ分解能 634
表 169. アナログウォッチドッグチャネル選択 644
表 170. アナログウォッチドッグ 1 の比較 645
表 171. アナログウォッチドッグ 2 および 3 の比較 645
表 172. 最大出力結果対 N と M（灰色のセルは切り詰め） 649
表 173. オーバーサンプリング動作モードの概要 654
表 174. 低電力モードが A/D コンバータに与える影響 672
表 175. ADC ごとの ADC 割込み 673
表 176. DELAY ビット対 A/D コンバータ分解能 708
表 177. A/D コンバータ·グローバル·レジスタ·マップ 710
表 178. A/D コンバータ·レジスタ·マップとリセット値（各 ADC、オフセット = 0x000（マスタ A/D コンバータ）、0x100（スレーブ A/D コンバータ）） 710
表 179. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300 712
表 180. D/A コンバータの機能 714
表 181. D/A コンバータの入出力ピン 716
表 182. D/A コンバータ入出力信号 716
表 183. DAC1 相互接続 717
表 184. DAC2 相互接続 718
表 185. DAC3 相互接続 719
表 186. DAC4 相互接続 720
表 187. データフォーマット（12 ビットデータの場合） 722
表 188. HFSEL の説明 723
表 189. サンプルおよびリフレッシュタイミング 730
表 190. チャネル出力モードの概要 732
表 191. 低電力モードが D/A コンバータに与える影響 739
表 192. D/A コンバータの割込み 739
表 193. D/A コンバータ·レジスタのマップとリセット値 759
表 194. VREF バッファモード 761
表 195. VREFBUF レジスタマップとリセット値 763
表 196. COMPx 非反転入力割付け 765
表 197. COMPx 反転入力割付け 765
表 198. ブランキング ソース 767
表 199. 低電力モードでのコンパレータの動作 768
表 200. COMP レジスタのマップとリセット値 771
表 201. オペアンプの可能な接続 773
表 202. 動作モードと較正 781
表 203. 低消費電力モードが OPAMP に与える影響 783

表 204. OPAMP レジスタのマップとリセット値 810
表 205. RNG 内部入力／出力信号 813
表 206. RNG 割込みリクエスト 819
表 207. RNG レジスタ･マップとリセット値 823
表 208. HRTIM の入出力の概要 828
表 209. 外部イベントのマッピングおよび関連機能 830
表 210. 更新イネーブル入力とソース 830
表 211. 汎用タイマからのバースト･モードのクロック･ソース 830
表 212. フォールト入力 831
表 213. HRTIM DAC トリガ接続 831
表 214. タイマ分解能および最小 PWM 周波数（f_{HRTIM} = 170 MHz の場合） 832
表 215. 周期および比較レジスタの最小値および最大値 836
表 216. タイマ動作モード 836
表 217. タイマ A ～ F におけるイベントマッピング 842
表 218. インタリーブモードの選択 843
表 219. インタリーブモードでの比較 1 ～ 3 の値 844
表 220. デッドタイム分解能と最大の絶対値 855
表 221. ロールオーバーイベントの転送先とモードのプログラミング 862
表 222. UDM ビットの設定によって変わる EExFLTR[3:0] コード 864
表 223. 外部イベントの機能 869
表 224. 出力のセット／リセット遅延時間とジッタ 対 外部イベント動作モード 870
表 225. タイマごとのフィルタリング信号のマッピング 873
表 226. タイマごとのウィンドウイング信号のマッピング（EEFLTR[3:0] = 1111） 875
表 227. HRTIM のプリロード可能な制御レジスタおよび関連する更新ソース 885
表 228. マスタタイマ更新イベントの伝播 890
表 229. TIMx 更新イベントの伝播 890
表 230. 更新を生成できるリセットイベント 892
表 231. タイマのリセットのための更新イベントの伝播 892
表 232. 出力状態のプログラミング（x = A ～ F、y = 1 または 2） 894
表 233. バーストモードのためのタイマ出力のプログラミング 897
表 234. フォールト入力 906
表 235. サンプリングレートおよびフィルタ長 対 FLTFxF[3:0] およびクロック設定 908
表 236. フォールト入力のブランキングイベント 908
表 237. フォールト 1 ～ 6 のカウンタリセットソース 909
表 238. 同期イベントの効果とタイマの動作モードとの関係 915
表 239. DAC デュアルチャネルトリガの例 922
表 240. HRTIM 割込みの概要 926
表 241. HRTIM の DMA リクエストの概要 927
表 242. HRTIM レジスタマップとリセット値 - マスタタイマ 1045
表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F） 1046
表 244. HRTIM レジスタマップとリセット値 - 共通機能 1051
表 245. TIM の入出力ピン 1057
表 246. TIM 内部入力／出力信号 1057
表 247. tim_ti1 入力マルチプレクサへの相互接続 1058
表 248. tim_ti2 入力マルチプレクサへの相互接続 1059
表 249. tim_ti3 入力マルチプレクサへの相互接続 1059
表 250. tim_ti4 入力マルチプレクサへの相互接続 1059
表 251. TIMx 内部トリガ接続 1059
表 252. tim_etr 入力マルチプレクサへの相互接続 1060
表 253. タイマ･ブレーク相互接続 1060
表 254. タイマ･ブレーク 2 相互接続 1061
表 255. システム･ブレーク相互接続 1061

表 256. ocref_clr 入力マルチプレクサへの相互接続 1061
表 257. CCR および ARR レジスタの変更のディザリングパターン 1095
表 258. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン 1096
表 259. タイマ出力と tim_brk/tim_brk2 入力の動作 1108
表 260. ブレーク保護解除条件 1109
表 261. カウント方向とエンコーダ信号（CC1P = CC2P = 0） 1118
表 262. カウント方向 対 エンコーダ信号および極性設定 1122
表 263. DMA リクエスト 1143
表 264. 低消費電力モードが TIM1/TIM8/TIM20 に与える影響 1143
表 265. 割込みリクエスト 1144
表 266. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット 1170
表 267. TIMx レジスタマップとリセット値 1195
表 268. STM32G4 シリーズ汎用タイマ 1199
表 269. TIM の入出力ピン 1201
表 270. TIM 内部入力／出力信号 1201
表 271. tim_ti1 入力マルチプレクサへの相互接続 1202
表 272. tim_ti2 入力マルチプレクサへの相互接続 1202
表 273. tim_ti3 入力マルチプレクサへの相互接続 1203
表 274. tim_ti4 入力マルチプレクサへの相互接続 1203
表 275. TIMx 内部トリガ接続 1203
表 276. tim_etr 入力マルチプレクサへの相互接続 1204
表 277. tim_ocref_clr 入力マルチプレクサへの相互接続 1204
表 278. CCR および ARR レジスタの変更のディザリングパターン 1234
表 279. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン 1235
表 280. カウント方向とエンコーダ信号（CC1P = CC2P = 0） 1244
表 281. カウント方向 対 エンコーダ信号および極性設定 1248
表 282. DMA リクエスト 1272
表 283. 低消費電力モードが TIM2/TIM3/TIM4/TIM5 に与える影響 1273
表 284. 割込みリクエスト 1273
表 285. 標準 tim_ocx チャネルの出力制御ビット 1291
表 286. TIM2/TIM3/TIM4/TIM5 レジスタマップとリセット値 1309
表 287. TIM の入出力ピン 1314
表 288. TIM 内部入力／出力信号 1315
表 289. tim_ti1 入力マルチプレクサへの相互接続 1315
表 290. tim_ti2 入力マルチプレクサへの相互接続 1316
表 291. TIMx 内部トリガ接続 1316
表 292. タイマブレーク相互接続 1316
表 293. システム･ブレーク相互接続 1317
表 294. ocref_clr 入力マルチプレクサへの相互接続 1317
表 295. CCR および ARR レジスタの変更のディザリングパターン 1337
表 296. ブレーク保護解除条件 1346
表 297. DMA リクエスト 1357
表 298. 低消費電力モードが TIM15/TIM16/TIM17 に与える影響 1358
表 299. 割込みリクエスト 1358
表 300. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット（TIM15） 1373
表 301. TIM15 レジスタマップとリセット値 1386
表 302. ブレーク機能を持つ相補 tim_oc1 および tim_oc1n チャネルの出力制御ビット機能（TIM16/TIM17） 1398
表 303. TIM16/TIM17 レジスタマップとリセット値 1412
表 304. TIM 内部入力／出力信号 1415
表 305. TIMx_ARR レジスタの変更のディザリングパターン 1425
表 306. DMA リクエスト 1427

表 307. 低消費電力モードが TIM6/TIM7 に与える影響 1427
表 308. 割込みリクエスト 1427
表 309. TIMx レジスタマップとリセット値 1433
表 310. STM32G4 シリーズ LPTIM 機能 1435
表 311. LPTIM1 外部トリガ接続 1435
表 312. LPTIM1 入力 1 接続 1436
表 313. LPTIM1 入力 2 接続 1436
表 314. プリスケーラ分周比 1438
表 315. エンコーダのカウントシナリオ 1444
表 316. 低電力モードが LPTIM に与える影響 1445
表 317. 割込みイベント 1446
表 318. LPTIM レジスタのマップとリセット値 1457
表 319. AES 内部入力／出力信号 1460
表 320. CTR モード初期化ベクタの定義 1477
表 321. GCM 最終ブロックの定義 1479
表 322. GCM モードでの AES_IVRx レジスタの初期化 1480
表 323. CCM モードでの AES_IVRx レジスタの初期化 1487
表 324. AES_KEYRx レジスタでのキーのエンディアン設定
（128 ビットまたは 256 ビットのキー長） 1493
表 325. AES 割込みリクエスト 1496
表 326. ECB、CBC および CTR での処理遅延時間 1496
表 327. GCM および CCM での処理遅延時間（クロック･サイクル数） 1496
表 328. AES レジスタマップとリセット値 1508
表 329. RTC の入出力ピン 1512
表 330. RTC 内部入力／出力信号 1512
表 331. RTC 相互接続 1512
表 332. PC13 設定 1513
表 333. RTC_OUT の配置 1514
表 334. 低消費電力モードが RTC に与える影響 1524
表 335. RTC ピン機能（モード共通） 1525
表 336. 割込みリクエスト 1525
表 337. RTC レジスタのマップとリセット値 1545
表 338. TAMP の入出力ピン 1549
表 339. TAMP 内部入力／出力信号 1549
表 340. TAMP 相互接続 1549
表 341. 低電力モードが TAMP に与える影響 1552
表 342. 割込みリクエスト 1552
表 343. TAMP レジスタのマップとリセット値 1561
表 344. USART/LPUART の機能 1564
表 345. サンプリングされたデータからのノイズ検出 1578
表 346. BRR [3:0] = 0000 のときの USART レシーバの許容誤差 1582
表 347. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差 1582
表 348. USART フレームのフォーマット 1586
表 349. 低電力モードが USART に与える影響 1608
表 350. USART 割込みリクエスト 1609
表 351. USART レジスタのマップとリセット値 1646
表 352. USART/LPUART の機能 1650
表 353. lpuart_ker_ck_pres = 32.768 kHz でプログラムされたボー･レートの誤差計算 1662
表 354. fCK = 100 MHz でプログラムされたボー･レートの誤差計算 1662
表 355. LPUART レシーバの許容誤差 1663
表 356. LPUART フレームのフォーマット 1666
表 357. 低消費電力モードが LPUART に与える影響 1675

表 358. LPUART 割込みリクエスト 1676
表 359. LPUART レジスタのマップとリセット値 1700
表 360. STM32G4 シリーズ SPI および SPI/I2S の実装 1703
表 361. SPI 割込みリクエスト 1728
表 362. HSE から派生した 48 MHz クロックを使用した場合のオーディオ周波数の精度 1740
表 363. I^2S 割込みリクエスト 1747
表 364. SPI/I2S レジスタマップとリセット値 1759
表 365. STM32G4 シリーズ SAI の機能 1761
表 366. SAI 内部入力／出力信号 1763
表 367. SAI の入出力ピン 1763
表 368. MCLK_x の有効化条件 1770
表 369. クロックジェネレータのプログラミング例 1773
表 370. TDM 設定 1780
表 371. TDM フレームの設定例 1782
表 372. SOPD パターン 1785
表 373. パリティビットの計算 1785
表 374. オーディオサンプリング周波数とシンボルレート 1786
表 375. SAI 割込みソース 1794
表 376. SAI レジスタのマップとリセット値 1824
表 377. I^2C の実装 1826
表 378. I^2C の入出力ピン 1828
表 379. I^2C 内部入力／出力信号 1828
表 380. アナログ･フィルタとデジタル･フィルタの比較 1830
表 381. I2C-SMBus 仕様のデータのセットアップおよびホールド時間 1833
表 382. I^2C 設定 1837
表 383. I2C-SMBus 仕様のクロック･タイミング 1847
表 384. fI2CCLK = 8 MHz でのタイミング設定の例 1858
表 385. fI2CCLK = 16 MHz でのタイミング設定の例 1858
表 386. fI2CCLK = 48 MHz でのタイミング設定の例 1859
表 387. SMBus タイムアウト仕様 1861
表 388. SMBus の PEC 設定 1863
表 389. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms） 1864
表 390. さまざまな I2CCLK 周波数での TIMEOUTB の設定例 1864
表 391. さまざまな I2CCLK 周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs） 1864
表 392. 低電力モードが I^2C に与える影響 1875
表 393. I^2C 割込みリクエスト 1876
表 394. I^2C レジスタマップとリセット値 1891
表 395. IWDG レジスタのマップとリセット値 1901
表 396. WWDG レジスタのマップとリセット値 1907
表 397. CAN サブシステム I/O 信号 1908
表 398. FDCAN での DLC コーディング 1915
表 399. フレーム送信の可能な設定 1927
表 400. Rx FIFO 要素 1930
表 401. Rx FIFO 要素の説明 1930
表 402. Tx バッファおよび FIFO 要素 1932
表 403. Tx バッファ要素の説明 1932
表 404. Tx イベント FIFO 要素 1934
表 405. Tx イベント FIFO 要素の説明 1934
表 406. 標準メッセージ ID フィルタ要素 1935
表 407. 標準メッセージ ID フィルタ要素フィールドの説明 1935
表 408. 拡張メッセージ ID フィルタ要素 1936
表 409. 拡張メッセージ ID フィルタ要素フィールドの説明 1936

表 410. FDCAN レジスタのマップとリセット値 1970
表 411. STM32G4 シリーズ の USB の実装 1973
表 412. ダブルバッファリングバッファフラグの定義 1982
表 413. バルクダブルバッファリングメモリバッファの用途 1982
表 414. アイソクロナスメモリバッファの使用 1984
表 415. レジュームイベント検出 1986
表 416. 受信ステータスエンコード 1998
表 417. エンドポイントタイプエンコード 1998
表 418. エンドポイントの種類の意味 1998
表 419. 送信ステータス･エンコード 1999
表 420. 割り当てられるバッファメモリの定義 2003
表 421. USB レジスタのマップとリセット値 2004
表 422. UCPD の実装 2007
表 423. ピンの UCPD 信号 2008
表 424. UCPD 内部信号 2009
表 425. 4B5B シンボル･エンコード表 2010
表 426. オーダード・セット 2012
表 427. オーダード・セットの検証 2012
表 428. データ･サイズ 2012
表 429. ANAMODE と ANASUBMODE のコーディングおよび TYPEC_VSTATE_CCx とのリンク 2021
表 430. Type-C シーケンス（ソース：3A）、ケーブル／シンク接続済み（CC1 に Rd、CC2 に Ra） 2022
表 431. 低電力モードが UCPD に与える影響 2024
表 432. UCPD 割込みリクエスト 2025
表 433. UCPD レジスタのマップとリセット値 2042
表 434. SWJ デバッグポートピン 2047
表 435. 柔軟性の高い SWJ-DP ピンの割り当て 2047
表 436. JTAG デバッグ･ポートのデータ･レジスタ 2052
表 437. シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ 2053
表 438. パケットリクエスト（8 ビット） 2054
表 439. ACK 応答（3 ビット） 2055
表 440. データ転送（33 ビット） 2055
表 441. SW-DP レジスタ 2056
表 442. FPU 搭載 Cortex®-M4AHB-AP レジスタ 2057
表 443. コアデバッグレジスタ 2058
表 444. 主な ITM レジスタ 2061
表 445. 主な ETM レジスタ 2062
表 446. 非同期 TRACE ピンの割当て 2069
表 447. 同期 TRACE ピンの割当て 2070
表 448. 柔軟性の高い TRACE ピン割り当て 2070
表 449. 重要な TPIU レジスタ 2073
表 450. DBG レジスタのマップとリセット値 2074
表 451. 文書改版履歴 2078
表 452. 日本語版文書改版履歴 2085

図の一覧

図 1. システムアーキテクチャ 78
図 2. メモリマップ 81
図 3. 連続 16 ビット命令の実行（64 ビット読出しデータ幅） 99
図 4. 読出し保護（RDP）レベルの変更 120
図 5. コアデバッグアクセスの無効化の例 125
図 6. 連続 16 ビット命令の実行（64 ビット読出しデータ幅） 150
図 7. 読出し保護（RDP）レベルの変更 165
図 8. コアデバッグアクセスの無効化の例 169
図 9. 連続 16 ビット命令の実行（64 ビット読出しデータ幅） 189
図 10. 読出し保護（RDP）レベルの変更 204
図 11. コアデバッグアクセスの無効化の例 208
図 12. STM32G4 シリーズ電源の概要 225
図 13. ブラウンアウトリセットの波形 230
図 14. PVD の閾値 231
図 15. 低消費電力モードの可能な遷移 233
図 16. 簡略化されたリセット回路図 268
図 17. クロックツリー 272
図 18. HSE/LSE クロックソース 273
図 19. TIM15 のキャプチャモードを使用した周波数測定 280
図 20. TIM16 のキャプチャモードを使用した周波数測定 280
図 21. TIM17 のキャプチャモードを使用した周波数測定 281
図 22. TIM5 のキャプチャモードを使用した周波数測定 281
図 23. CRS のブロック図 337
図 24. CRS カウンタの動作 338
図 25. I/O ポートビットの基本構成 348
図 26. 5 V トレラント I/O ポート･ビットの基本構成 348
図 27. 入力フローティング／プルアップ／プルダウン設定 353
図 28. 出力設定 354
図 29. オルタネート機能設定 354
図 30. ハイ･インピーダンス･アナログ設定 355
図 31. DMA のブロック図 399
図 32. DMAMUX のブロック図 422
図 33. DMAMUX リクエスト･ライン･マルチプレクサ･チャネルの同期モード 425
図 34. DMA リクエスト･ライン･マルチプレクサ･チャネルのイベント生成 425
図 35. 設定可能な割込み／イベントブロック図 440
図 36. 外部割込み／イベント GPIO の配置 442
図 37. CRC 計算ユニットのブロック図 454
図 38. 三角関数に対する CORDIC の収束 468
図 39. 双曲線関数に対する CORDIC の収束 469
図 40. 平方根に対する CORDIC の収束 470
図 41. ブロック図 479
図 42. 入力バッファ領域 481
図 43. サーキュラ入力バッファ 482
図 44. サーキュラ入力バッファの操作 483
図 45. サーキュラ出力バッファ 484
図 46. サーキュラ出力バッファの操作 485
図 47. FIR フィルタの構造 487
図 48. IIR フィルタの構造（直接型 1） 489
図 49. X1 バッファの初期化 494

図 50. フィルタ例 1 495
図 51. フィルタ例 2 496
図 52. FMC ブロック図 507
図 53. FMC メモリバンク 510
図 54. モード 1 読出しアクセス波形 517
図 55. モード 1 書込みアクセス波形 518
図 56. モード A 読出しアクセス波形 519
図 57. モード A 書込みアクセス波形 520
図 58. モード 2 およびモード B 読出しアクセス波形 522
図 59. モード 2 書込みアクセス波形 522
図 60. モード B 書込みアクセス波形 523
図 61. モード C 読出しアクセス波形 525
図 62. モード C 書込みアクセス波形 525
図 63. モード D 読出しアクセス波形 527
図 64. モード D 書込みアクセス波形 528
図 65. マルチプレクス読出しアクセス波形 530
図 66. マルチプレクス書込みアクセス波形 531
図 67. 読出しアクセス中の非同期ウェイト波形 533
図 68. 書込みアクセス中の非同期ウェイト波形 534
図 69. 待ち設定波形 536
図 70. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM） 536
図 71. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM） 538
図 72. 共通メモリアクセスの NAND Flash コントローラ波形 550
図 73. 'CE don't care' の NAND Flash へのアクセス 552
図 74. QUADSPI ブロック図（デュアル Flash モードが無効のとき） 561
図 75. QUADSPI ブロック図（デュアル Flash モードが有効のとき） 562
図 76. クワッドモードにおける読出しコマンドの例 563
図 77. クワッドモードにおける DDR コマンドの例 566
図 78. nCS - CKMODE = 0（T = CLK 周期）のとき 574
図 79. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき 574
図 80. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき 574
図 81. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき 575
図 82. ADC ブロック図 593
図 83. A/D コンバータのクロック構成 596
図 84. ADC1 接続性 597
図 85. ADC2 接続性 598
図 86. ADC3 接続性 599
図 87. ADC4 接続性 600
図 88. ADC5 接続性 601
図 89. A/D コンバータの較正 604
図 90. A/D コンバータの較正係数の更新 605
図 91. シングルエンド･チャネルと差動チャネルの混在 605
図 92. A/D コンバータの有効化と無効化 607
図 93. バルブモードのタイミング図 610
図 94. アナログ / デジタル変換時間 613
図 95. 実行中のレギュラ変換の停止 614
図 96. 実行中のレギュラ変換とインジェクト変換の停止 614
図 97. ADC マスタと ADC スレーブ間でのトリガの共有 616
図 98. インジェクト変換の遅延時間 621
図 99. コンテキストの JSQR キューの例（シーケンスの変更） 624
図 100. コンテキストの JSQR キューの例（トリガの変更） 624
図 101. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例 625

図 102. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例 625
図 103. 空のキューがあるコンテキストの JSQR キューの例（JQM = 0 の場合） 626
図 104. 空のキューがあるコンテキストの JSQR キューの例（JQM = 1 の場合） 627
図 105. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に JADSTP が発生した場合 627
図 106. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に JADSTP が発生し、新しいトリガが発生した場合。 628
図 107. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に外部で JADSTP が発生した場合 628
図 108. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 1） 629
図 109. ADDIS=1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0） 629
図 110. ADDIS = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 1） 630
図 111. シーケンスのシングル変換、ソフトウェアトリガ 632
図 112. シーケンスの連続変換、ソフトウェア･トリガ 632
図 113. シーケンスのシングル変換、ハードウェア･トリガ 633
図 114. シーケンスの連続変換、ハードウェア･トリガ 633
図 115. 右詰め（オフセット無効、符号なしの値） 635
図 116. 右詰め（オフセット有効、符号付きの値） 635
図 117. 左詰め（オフセット無効、符号なしの値） 636
図 118. 左詰め（オフセット有効、符号付きの値） 636
図 119. オーバーラン（OVR）の例 638
図 120. AUTODLY = 1、連続モードのレギュラ変換、ソフトウェア･トリガ 641
図 121. AUTODLY = 1、インジェクト変換によって中断されたレギュラ･ハードウェア変換（DISCEN = 0、JDISCEN = 0） 641
図 122. AUTODLY = 1、インジェクト変換によって中断されたレギュラ･ハードウェア変換（DISCEN = 1、JDISCEN = 1） 642
図 123. AUTODLY = 1、インジェクト変換によって中断されたレギュラ連続変換 643
図 124. 自動インジェクト･モード（JAUTO = 1）の AUTODLY = 1 643
図 125. アナログ･ウォッチドッグによって保護される領域 644
図 126. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル） 646
図 127. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合） 647
図 128. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル） 647
図 129. ADCy_AWDx_OUT 信号生成（すべてのインジェクト･チャネル） 647
図 130. 20 ビットから 16 ビットへの結果の切り詰め 649
図 131. 5 ビット･シフトと丸めの数値例 649
図 132. オーバーサンプリングモードのトリガ（TROVS ビット =1） 651
図 133. レギュラ･オーバーサンプリング･モード（4x の比率） 652
図 134. レギュラおよびインジェクトオーバーサンプリングモードの同時使用 652
図 135. インジェクションでのトリガレギュラオーバーサンプリング 653
図 136. 自動インジェクトモードでのオーバーサンプリング 653
図 137. デュアル ADC ブロック図 (1) 656
図 138. 4 チャネルのインジェクト同時モード：デュアル ADC モード 657
図 139. 16 チャネルのレギュラ同時モード：デュアル A/D コンバータ･モード 659
図 140. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 660
図 141. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード 661
図 142. インジェクションによるインタリーブ変換 661
図 143. オルタネートトリガ：各 ADC のインジェクトグループ 662
図 144. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC） 663
図 145. オルタネート + レギュラ同時 664
図 146. インジェクト変換中にトリガが発生した場合 664
図 147. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0 665

図 148. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割込み優先 665
図 149. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割込み優先 665
図 150. レギュラ同時モードにおける DMA リクエスト（MDMA = 00） 666
図 151. レギュラ同時モードにおける DMA リクエスト（MDMA = 10） 667
図 152. インタリーブ･モードにおける DMA リクエスト（MDMA = 10） 667
図 153. 温度センサチャネルのブロック図 669
図 154. VBAT チャネルのブロック図 670
図 155. VREFINT チャネルのブロック図 671
図 156. デュアル･チャネル D/A コンバータのブロック図 715
図 157. シングル DAC チャネルモードのデータレジスタ 721
図 158. デュアル DAC チャネルモードのデータレジスタ 722
図 159. トリガ無効（TEN = 0）時の変換タイミング図 723
図 160. DAC LFSR レジスタ計算アルゴリズム 726
図 161. LFSR 波形生成による DAC 変換（SW トリガ有効） 726
図 162. DAC 三角波生成 727
図 163. 三角波生成による DAC 変換（SW トリガ有効） 727
図 164. D/A コンバータによるのこぎり波生成（STDIR x = 0） 728
図 165. D/A コンバータによるのこぎり波生成（STDIR x = 1） 728
図 166. DAC のこぎり波 STINCTRIG および STRSTTRIG の優先度（STDIR=0） 729
図 167. D/A コンバータのサンプル･アンド･ホールド･モードのフェーズ図 731
図 168. コンパレータブロック図 764
図 169. コンパレータヒステリシス 766
図 170. コンパレータの出力のブランキング 767
図 171. スタンドアロンモード：外部ゲイン設定モード 775
図 172. フォロア設定 776
図 173. PGA モード、内部ゲイン設定（x2/x4/x8/x16/x32/x64）、反転入力は使用されません。 777
図 174. PGA モード、内部ゲイン設定（x2/x4/x8/x16/x32/x64）、フィルタリング用の反転入力 778
図 175. PGA モード、非反転ゲイン設定（x2/x4/x8/x16/x32/x64）または反転ゲイン設定（x-1/x-3/x-7/x-15/x-31/x-63） 779
図 176. PGA モード、非反転ゲイン設定（x2/x4/x8/x16/x32/x64）または反転ゲイン設定（x-1/x-3/x-7/x-15/x-31/x-63）フィルタ 780
図 177. 設定例 780
図 178. タイマによって制御されたマルチプレクサモード 783
図 179. RNG のブロック図 813
図 180. エントロピー･ソース･モデル 814
図 181. RNG 初期化の概要 816
図 182. 高分解能タイマの概要 827
図 183. カウンタおよびキャプチャレジスタのフォーマット 対 クロックプリスケーラ係数 833
図 184. タイマ A ～ F の概要 835
図 185. 連続タイマ動作 837
図 186. シングルショットタイマ動作 837
図 187. タイマリセットの再同期（32 を超えるプリスケーリング比） 839
図 188. 連続モードでの繰り返しレート 対 HRTIM_REPxR の内容 840
図 189. シングルショットモードでの繰り返しカウンタの動作 840
図 190. 比較イベントの出力への作用：比較 1 でセット、比較 2 でリセット 842
図 191. タイマ A のタイミングユニットのキャプチャ回路 846
図 192. 自動遅延の概要（比較 2 のみ） 847
図 193. 自動遅延比較 848
図 194. トリガハーフモードの例 850
図 195. プッシュプルモードのブロック図 851

図 196. プッシュプルモードの例 852
図 197. デッドタイム付きのプッシュプル 853
図 198. デッドタイムが挿入された相補出力 854
図 199. デッドタイム挿入 対 デッドタイムの符号（1 は負のデッドタイムを示す） 854
図 200. 小パルス幅に対する相補出力（SDTRx = SDTFx = 0） 855
図 201. 小パルス幅に対する相補出力（SDTRx = SDTFx = 1） 855
図 202. 小パルス幅に対する相補出力（SDTRx = 0、SDTFx = 1） 856
図 203. 小パルス幅に対する相補出力（SDTRx = 1、SDTFx = 0） 856
図 204. マスタタイマの概要 857
図 205. アップダウンカウントモードでの対称な基本的波形 858
図 206. アップダウンカウントモードでの複雑な対称波形 859
図 207. アップダウンカウントモードでの非対称波形 859
図 208. アップダウンカウントモードでの外部イベントの管理 860
図 209. インタリーブされたアップダウンカウンタ動作 860
図 210. インタリーブされたアップダウンカウンタ動作 861
図 211. プッシュプル・アップダウンモードの例 861
図 212. 「より大きい」比較を備えたアップダウンモード 862
図 213. 周期イベントで出力セットしたアップダウンモード（OUTROM[1:0] = 10 の場合） 863
図 214. アップダウンカウントモードでの繰り返しカウンタの動作 864
図 215. 狭いパルス生成に対する短い間隔のセット／リセットの管理 866
図 216. 外部イベント調整の概要（代表 1 チャネル） 868
図 217. 外部イベントの立ち下がりエッジの遅延時間（カウンタリセットおよび出力セット） 871
図 218. 外部イベントへの遅延時間（外部イベントによる出力リセット） 871
図 219. イベントブランキングモード 872
図 220. イベント延期モード 872
図 221. エッジ対応トリガによる外部トリガブランキング 874
図 222. レベル対応トリガによる外部トリガブランキング 874
図 223. イベントウインドウイングモード 875
図 224. エッジ対応トリガによる外部トリガウィンドウイング 876
図 225. レベル対応トリガによる外部トリガウィンドウイング 876
図 226. 外部イベントのカウンタ – チャネル A 877
図 227. 外部イベントカウンタの累積モード（EEVxRSTM = 1、EEVxCNT = 2） 878
図 228. 遅延アイドルモードへの移行 879
図 229. バーストモードおよび遅延保護優先順位（DIDL = 0） 880
図 230. バーストモードおよび遅延保護優先順位（DIDL = 1） 881
図 231. バランスアイドル保護の例 882
図 232. 再同期されたタイマ更新（HRTIM_TIMBCR の TAU=1） 887
図 233. 「より大きい」PWM モードでの早期ターンオンと早期ターンオフ動作 889
図 234. 出力管理の概要 894
図 235. HRTIM 出力状態と遷移 895
図 236. バーストモードの動作例 896
図 237. 外部イベントによるバーストモードのトリガ 899
図 238. デッドタイムを有効にして IDLESx = 1 の場合の遅延バーストモードへの移行 900
図 239. デッドタイム中の遅延バーストモードへの移行 901
図 240. デッドタイムジェネレータが有効なときのバーストモードの終了 902
図 241. バーストモードのエミュレーション例 904
図 242. キャリア周波数信号の挿入 904
図 243. チョッパモードが有効化された HRTIM 出力 905
図 244. フォールト保護回路（FAULT1 は完全表示、FAULT2 ～ 6 は部分表示） 906
図 245. フォールト信号のフィルタリング（FLTxF[3:0]= 0010：$f_{SAMPLING}$ = fHRTIM、N = 4） 907
図 246. フォールトカウンタの累積モード（FLTxRSTM = 1、FLTxCNT[3:0] = 2） 909
図 247. 補助出力 911

図 248. バーストモード中の補助およびメイン出力（DIDLx = 0）.....912
図 249. バーストモードを終了する時の補助出力上のデッドタイムの歪み 913
図 250. 同期開始モードでのカウンタの動作.....916
図 251. ADC トリガ選択の概要 917
図 252. ADC トリガ.....918
図 253. アップカウントモードでの ADC トリガのポストスケーリング.....919
図 254. アップダウンカウントモードでの ADC トリガのポストスケーリング 919
図 255. 単一の hrtim_dac_trgx 出力への複数の更新の組み合わせ 920
図 256. スロープ補正のための DAC トリガ 923
図 257. DAC トリガの概要.....924
図 258. DMA バーストの概要.....928
図 259. バースト DMA の動作フローチャート 929
図 260. DMA バースト転送後のレジスタ更新.....930
図 261. 降圧コンバータの接続形態.....932
図 262. デュアル降圧コンバータのマネージメント 933
図 263. 出力電流に応じた同期整流.....933
図 264. 同期整流式の降圧 934
図 265. 3 相インターリーブ降圧コンバータ 934
図 266. 3 相インターリーブ降圧コンバータの制御.....935
図 267. 遷移モード PFC.....936
図 268. 遷移モード PFC の波形.....937
図 269. 高機能制御タイマのブロック図 1056
図 270. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図.....1063
図 271. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図.....1063
図 272. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....1065
図 273. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1065
図 274. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....1066
図 275. 内部クロック分周比が N の場合のカウンタのタイミング図 1066
図 276. ARPE=0 の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARR はプリロードされない）.....1067
図 277. ARPE=1 の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロード）.....1068
図 278. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....1069
図 279. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1070
図 280. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....1070
図 281. 内部クロック分周比が N の場合のカウンタのタイミング図 1071
図 282. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図 1071
図 283. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 1073
図 284. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1073
図 285. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図 1074
図 286. 内部クロック分周比が N の場合のカウンタのタイミング図 1074
図 287. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図.....1075
図 288. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図.....1076
図 289. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1077
図 290. 外部トリガ入力ブロック.....1078
図 291. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図.....1079
図 292. tim_ti2 外部クロックの接続例.....1079
図 293. 外部クロックモード 1 の制御回路 1080
図 294. 外部トリガ入力ブロック.....1081
図 295. 外部クロックモード 2 の制御回路 1082
図 296. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）.....1082
図 297. キャプチャ／比較チャネル 1 メイン回路.....1083
図 298. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3 および 4）の出力ステージ....1084

図 299. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ 1084
図 300. PWM 入力モードタイミング 1086
図 301. 出力比較モード、tim_oc1 の反転 1088
図 302. エッジアライン PWM 波形（ARR=8） 1090
図 303. センターアライン PWM 波形（ARR=8） 1091
図 304. ディザリング原理 1092
図 305. ディザリングモードでのデータフォーマットとレジスタのコーディング 1093
図 306. PWM 分解能 対 周波数 1094
図 307. PWM ディザリングのパターン 1095
図 308. センターアライン PWM モードでのデューティサイクルへのディザリングの影響 1096
図 309. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成 1098
図 310. チャネル 1 および 3 における組み合わせ PWM モード 1099
図 311. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号 1100
図 312. 対称のデッドタイム挿入のある相補出力 1101
図 313. 非対称のデッドタイム 1102
図 314. 負のパルスより長い遅延があるときのデッドタイムの波形 1102
図 315. 正のパルスより長い遅延があるときのデッドタイムの波形 1102
図 316. ブレークおよびブレーク 2 回路の概要 1105
図 317. tim_brk（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作 1107
図 318. tim_brk および tim_brk2 のアサート後の PWM 出力状態（OSSI=1） 1108
図 319. tim_brk アサート後の PWM 出力状態（OSSI=0） 1108
図 320. 出力先変更（tim_brk2 リクエストは表示なし） 1110
図 321. tim_ocref_clr 入力選択マルチプレクサ 1111
図 322. TIMx tim_ocxref のクリア 1111
図 323. 6 ステップ生成 COM の例（OSSR=1） 1112
図 324. ワンパルスモードの例 1113
図 325. 再トリガ可能なワンパルス･モード 1115
図 326. パルス発生回路 1115
図 327. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成 1116
図 328. 同時トリガの場合の拡張されたパルス幅 1117
図 329. エンコーダインタフェースモードにおけるカウンタの動作例 1119
図 330. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例 1119
図 331. 直交エンコーダのカウントモード 1120
図 332. 方向プラスクロックエンコーダモード 1121
図 333. 方向クロックエンコーダモード（CC1P = CC2P = 0） 1121
図 334. 方向クロックエンコーダモード（CC1P = CC2P = 1） 1122
図 335. インデックスゲーティングオプション 1123
図 336. ジッタのあるインデックス信号 1123
図 337. IPOS[1:0] = 11 の場合のインデックス生成 1124
図 338. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値 1124
図 339. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値 1125
図 340. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値 1125
図 341. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11） 1126
図 342. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07） 1127
図 343. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01） 1128
図 344. インデックスの方向感受性 1129
図 345. FIDX ビット設定によって変わるカウンタリセット 1129
図 346. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1） 1130
図 347. 方向クロックモードでのインデックスの動作（IPOS[0] = 1） 1130
図 348. 直交エンコードされた信号の状態図 1131
図 349. アップカウントでのエンコーダエラー検出 1132
図 350. ダウンカウントでのエンコーダエラー検出 1133

図 351. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）.....1134
図 352. 3 つの信号上のエッジ間の時間間隔の測定.....1135
図 353. ホールセンサインタフェースの例.....1137
図 354. リセットモードの制御回路.....1138
図 355. ゲートモードの制御回路.....1139
図 356. トリガモードの制御回路.....1140
図 357. 外部クロックモード 2 ＋トリガモードの制御回路.....1141
図 358. 汎用タイマのブロック図.....1200
図 359. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図.....1206
図 360. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図.....1206
図 361. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....1207
図 362. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1208
図 363. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....1208
図 364. 内部クロック分周比が N の場合のカウンタのタイミング図.....1209
図 365. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図.....1209
図 366. ARPE=1（TIMx_ARR はプリロードされる）のときの更新イベント時のカウンタのタイミング図.....1210
図 367. 内部クロック分周比が 1 の場合のカウンタのタイミング図.....1211
図 368. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1211
図 369. 内部クロック分周比が 4 の場合のカウンタのタイミング図.....1212
図 370. 内部クロック分周比が N の場合のカウンタのタイミング図.....1212
図 371. 更新イベント時のカウンタのタイミング図.....1213
図 372. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図.....1214
図 373. 内部クロック分周比が 2 の場合のカウンタのタイミング図.....1215
図 374. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図.....1215
図 375. 内部クロック分周比が N の場合のカウンタのタイミング図.....1216
図 376. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図.....1216
図 377. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図.....1217
図 378. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図.....1218
図 379. tim_ti2 外部クロックの接続例.....1218
図 380. 外部クロックモード 1 の制御回路.....1219
図 381. 外部トリガ入力ブロック.....1220
図 382. 外部クロックモード 2 の制御回路.....1221
図 383. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）.....1221
図 384. キャプチャ／比較チャネル 1 メイン回路.....1222
図 385. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3、および 4）の出力ステージ.....1222
図 386. PWM 入力モードタイミング.....1225
図 387. 出力比較モード、tim_oc1 の反転.....1226
図 388. エッジアライン PWM 波形（ARR=8）.....1228
図 389. センターアライン PWM 波形（ARR=8）.....1229
図 390. ディザリング原理.....1230
図 391. ディザリングモードでのデータフォーマットとレジスタのコーディング.....1231
図 392. PWM 分解能 対 周波数（16 ビットモード）.....1232
図 393. PWM 分解能 対 周波数（32 ビットモード）.....1232
図 394. PWM ディザリングのパターン.....1233
図 395. センターアライン PWM モードでのデューティサイクルへのディザリングの影響.....1234
図 396. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成.....1236
図 397. チャネル 1 および 3 における組み合わせ PWM モード.....1237
図 398. OCREF_CLR 入力選択マルチプレクサ.....1238
図 399. TIMx tim_ocxref のクリア.....1238
図 400. ワンパルス･モードの例.....1239

図 401. 再トリガ可能なワンパルス･モード 1241
図 402. パルス発生回路 1241
図 403. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成 1242
図 404. 同時トリガの下で延長されたパルス幅 1243
図 405. エンコーダインタフェースモードにおけるカウンタの動作例 1245
図 406. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例 1245
図 407. 直交エンコーダのカウントモード 1246
図 408. 方向プラスクロックエンコーダモード 1247
図 409. 方向クロックエンコーダモード（CC1P = CC2P = 0） 1247
図 410. 方向クロックエンコーダモード（CC1P = CC2P = 1） 1248
図 411. インデックスゲーティングオプション 1249
図 412. ジッタのあるインデックス信号 1249
図 413. IPOS[1:0] = 11 の場合のインデックス生成 1250
図 414. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値 1250
図 415. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値 1251
図 416. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値 1251
図 417. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11） 1252
図 418. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07） 1253
図 419. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01） 1254
図 420. インデックスの方向感受性 1255
図 421. FIDX ビット設定によって変わるカウンタリセット 1255
図 422. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1） 1256
図 423. 方向クロックモードでのインデックスの動作（IPOS[0] = 1） 1256
図 424. 直交エンコードされた信号の状態図 1257
図 425. アップカウントでのエンコーダエラー検出 1258
図 426. ダウンカウントでのエンコーダエラー検出 1259
図 427. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0） 1260
図 428. リセットモードの制御回路 1262
図 429. ゲートモードの制御回路 1263
図 430. トリガモードの制御回路 1264
図 431. 外部クロックモード 2 ＋トリガモードの制御回路 1265
図 432. マスタ / スレーブ･タイマの例 1266
図 433. 1 チャネルのみのタイマによるマスタ / スレーブ接続例 1266
図 434. TIM_mstr の tim_oc1ref による TIM_slv のゲート操作 1268
図 435. TIM_mstr の有効化による TIM_slv のゲート操作 1269
図 436. TIM_mstr の更新による TIM_slv のトリガ 1269
図 437. TIM_mstr の有効化による TIM_slv のトリガ 1270
図 438. TIM_mstr の tim_ti1 入力による TIM_mstr と TIM_slv のトリガ 1271
図 439. TIM15 ブロック図 1313
図 440. TIM16/TIM17 ブロック図 1314
図 441. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1318
図 442. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1319
図 443. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1320
図 444. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1321
図 445. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1321
図 446. 内部クロック分周比が N の場合のカウンタのタイミング図 1322
図 447. ARPE=0 の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARR はプリロードされない） 1322
図 448. ARPE=1 の場合の更新イベント時のカウンタのタイミング図
（TIMx_ARR はプリロード） 1323
図 449. モードと TIMx_RCR レジスタの設定に応じた更新レートの例 1324
図 450. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 1325

図 451. tim_ti2 外部クロックの接続例 1325
図 452. 外部クロックモード 1 の制御回路 1326
図 453. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ） 1327
図 454. キャプチャ／比較チャネル 1 メイン回路 1327
図 455. キャプチャ／比較チャネル（チャネル 1）の出力ステージ 1328
図 456. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ 1328
図 457. PWM 入力モードタイミング 1331
図 458. 出力比較モード、tim_oc1 の反転 1332
図 459. エッジアライン PWM 波形（ARR=8） 1334
図 460. ディザリング原理 1335
図 461. ディザリングモードでのデータフォーマットとレジスタのコーディング 1335
図 462. PWM 分解能 対 周波数 1336
図 463. PWM ディザリングのパターン 1337
図 464. チャネル 1 および 2 における組み合わせ PWM モード 1339
図 465. 対称のデッドタイム挿入のある相補出力。 1340
図 466. 非対称のデッドタイム 1341
図 467. 負のパルスより長い遅延があるときのデッドタイムの波形 1341
図 468. 正のパルスより長い遅延があるときのデッドタイムの波形 1341
図 469. ブレーク回路の概要 1343
図 470. tim_brk でのブレークイベントに対する出力の動作 1345
図 471. 出力先変更 1347
図 472. tim_ocref_clr 入力選択マルチプレクサ 1348
図 473. 6 ステップ生成 COM の例（OSSR = 1） 1349
図 474. ワンパルスモードの例 1350
図 475. 再トリガ可能なワンパルスモード 1352
図 476. 2 つの信号上のエッジ間の時間間隔の測定 1352
図 477. リセットモードの制御回路 1353
図 478. ゲートモードの制御回路 1354
図 479. トリガモードの制御回路 1355
図 480. 基本タイマブロック図 1415
図 481. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 1416
図 482. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 1417
図 483. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 1418
図 484. 内部クロック分周比が 1 の場合のカウンタのタイミング図 1419
図 485. 内部クロック分周比が 2 の場合のカウンタのタイミング図 1419
図 486. 内部クロック分周比が 4 の場合のカウンタのタイミング図 1420
図 487. 内部クロック分周比が N の場合のカウンタのタイミング図 1420
図 488. ARPE=0（TIMx_ARR がプリロードされない）の場合の更新イベント時のカウンタのタイミング図 1421
図 489. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図 1422
図 490. ディザリング原理 1423
図 491. ディザリングモードでのデータフォーマットとレジスタのコーディング 1424
図 492. FCnt 分解能 対 周波数 1425
図 493. PWM ディザリングのパターン 1425
図 494. 低電力タイマのブロック図 1435
図 495. グリッチフィルタのタイミング図 1437
図 496. LPTIM 出力波形、シングルカウントモードの設定 1439
図 497. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット） 1439
図 498. LPTIM 出力波形、連続カウントモードの設定 1440
図 499. 波形生成 1441

図 500. エンコーダモードのカウントシーケンス 1445
図 501. TIM16 および TIM17 との IRTIM 内部ハードウェア接続 1458
図 502. AES ブロック図 1460
図 503. ECB の暗号化と復号化の原理 1462
図 504. CBC の暗号化と復号化の原理 1463
図 505. CTR の暗号化と復号化の原理 1464
図 506. GCM の暗号化と認証の原理 1465
図 507. GMAC の認証の原理 1465
図 508. CCM の暗号化と認証の原理 1466
図 509. ECB ／ CBC 復号（モード 2）のための暗号化キー派生 1469
図 510. サスペンドモードの管理例 1470
図 511. ECB 暗号化 1471
図 512. ECB 復号化 1471
図 513. CBC 暗号化 1472
図 514. CBC 復号化 1472
図 515. ECB/CBC 暗号化（モード 1） 1473
図 516. ECB/CBC 復号化（モード 3） 1474
図 517. CTR モードでのメッセージ構造 1476
図 518. CTR 暗号化 1476
図 519. CTR 復号化 1477
図 520. GCM でのメッセージ構造 1478
図 521. GCM 認証暗号化 1480
図 522. GMAC モードでのメッセージ構造 1484
図 523. GMAC 認証モード 1484
図 524. CCM モードでのメッセージ構造 1485
図 525. CCM モード認証暗号化 1487
図 526. データ･スワッピングに関する 128 ビット･ブロックの構成 1492
図 527. 入力フェーズでの 128 ビット･データ･ブロックの DMA 転送 1494
図 528. 出力フェーズでの 128 ビットデータブロックの DMA 転送 1494
図 529. RTC ブロック図 1511
図 530. TAMP ブロック図 1548
図 531. USART ブロック図 1565
図 532. ワード長のプログラミング 1568
図 533. 設定可能なストップビット 1570
図 534. 送信時の TC/TXE の動作 1572
図 535. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出 1573
図 536. usart_ker_ck クロック分周回路のブロック図 1576
図 537. データサンプリング（16 倍のオーバーサンプリング） 1578
図 538. データサンプリング（8 倍のオーバーサンプリング） 1578
図 539. アイドルライン検出を使用したミュートモード 1584
図 540. アドレスマーク検出を使用したミュートモード 1585
図 541. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1） 1588
図 542. LIN モードでのブレーク検出とフレーミングエラー検出 1589
図 543. USART の同期マスタ送信の例 1590
図 544. 同期マスタモードでの USART データクロックタイミング図（M ビット = 00） 1590
図 545. 同期マスタ･モードでの USART データ･クロック･タイミング図（M ビット = 01） 1591
図 546. 同期スレーブモードでの USART データクロックタイミング図（M ビット = 00） 1592
図 547. ISO 7816-3 非同期プロトコル 1594
図 548. ストップビット 1.5 個を使用したパリティエラー検出 1596
図 549. IrDA SIR ENDEC ブロック図 1599
図 550. IrDA データ変調（3/16）- 通常モード 1600
図 551. DMA を使用した送信 1601

図 552. DMA を使用した受信 1602
図 553. 2 つの USART 間のハードウェアフロー制御 1602
図 554. RS232 RTS フロー制御 1603
図 555. RS232 CTS フロー制御 1604
図 556. 確認されたウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 1607
図 557. 確認されないウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 1607
図 558. LPUART ブロック図 1651
図 559. LPUART ワード長のプログラミング 1653
図 560. 設定可能なストップビット 1655
図 561. 送信時の TC/TXE の動作 1657
図 562. lpuart_ker_ck クロック分周回路のブロック図 1660
図 563. アイドルライン検出を使用したミュートモード 1664
図 564. アドレスマーク検出を使用したミュートモード 1665
図 565. DMA を使用した送信 1668
図 566. DMA を使用した受信 1669
図 567. 2 つの LPUART 間のハードウェアフロー制御 1670
図 568. RS232 RTS フロー制御 1670
図 569. RS232 CTS フロー制御 1671
図 570. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効） . . . 1674
図 571. 確認されないウェイクアップイベント
（ウェイクアップイベント = アドレス一致、FIFO 無効） 1674
図 572. SPI ブロック図 1704
図 573. 全二重シングルマスタ／シングルスレーブアプリケーション 1705
図 574. 半二重シングルマスタ／シングルスレーブアプリケーション 1705
図 575. 単方向シングルマスタ／シングルスレーブアプリケーション
（送信専用モードのマスタ／受信専用モードのスレーブ） 1706
図 576. マスタと 3 つの独立したスレーブ 1707
図 577. マルチマスタアプリケーション 1708
図 578. ハードウェア／ソフトウェアスレーブ選択管理 1709
図 579. データクロックのタイミング図 1711
図 580. データ長が 8 ビットまたは 16 ビットと等しくない場合のデータ配置 1712
図 581. 送受信用 FIFO でのデータのパッキング 1716
図 582. マスタの全二重通信 1719
図 583. スレーブの全二重通信 1720
図 584. CRC のあるマスタの全二重通信 1721
図 585. パックされたモードでのマスタの全二重通信 1722
図 586. モトローラ SPI マスタモードでの NSSP パルス生成 1725
図 587. TI モードでの転送 1726
図 588. I^2S ブロック図 1729
図 589. I^2S フィリップスプロトコルの波形（16/32 ビットフル精度） 1731
図 590. I^2S フィリップス規格の波形（24 ビットフレーム） 1731
図 591. 0x8EAA33 の送信 1732
図 592. 0x8EAA33 の受信 1732
図 593. I^2S フィリップス規格（32 ビットパケットフレームに拡張された 16 ビット） 1732
図 594. 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例 1732
図 595. MSB 詰め 16 ビットまたは 32 ビット長フル精度 1733
図 596. MSB 詰め 24 ビットフレーム長 1733
図 597. 32 ビットパケットフレームに拡張された MSB 詰め 16 ビット 1733
図 598. LSB 詰め 16 ビットまたは 32 ビットフル精度 1734
図 599. LSB 詰め 24 ビットフレーム長 1734

図 600. 0x3478AE を送信するために必要な動作 1734
図 601. 0x3478AE の受信に必要な動作 1735
図 602. 32 ビットパケットフレームに拡張された LSB 詰め 16 ビット 1735
図 603. 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例 1735
図 604. PCM 規格の波形（16 ビット） 1736
図 605. PCM 規格の波形（32 ビットパケットフレームに拡張された 16 ビット） 1736
図 606. マスタモードでの開始シーケンス 1737
図 607. オーディオサンプリング周波数の定義 1738
図 608. I²S クロックジェネレータのアーキテクチャ 1739
図 609. SAI 機能ブロック図 1762
図 610. オーディオフレーム 1765
図 611. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1） 1767
図 612. FS の機能はフレーム開始（FSDEF = 0） 1768
図 613. スロット･サイズ設定（SAI_xSLOTR で FBOFF = 0 の場合） 1769
図 614. 先頭ビットオフセット 1769
図 615. オーディオブロッククロックジェネレータの概要 1771
図 616. PDM 標準接続とタイミング 1775
図 617. PDM インタフェースの詳細ブロック図 1776
図 618. 起動シーケンス 1777
図 619. TDM の SAI_ADR フォーマット（32 ビットスロット幅） 1778
図 620. TDM の SAI_ADR フォーマット（16 ビットスロット幅） 1779
図 621. TDM の SAI_ADR フォーマット（8 ビットスロット幅） 1780
図 622. AC’97 オーディオフレーム 1783
図 623. SPDIF フォーマット 1784
図 624. SAI_xDR レジスタの順序 1785
図 625. SAI オーディオブロックのデータコンパンディングハードウェア 1788
図 626. 非アクティブ･スロットにおける SD 出力ラインのトライステート化 1789
図 627. I2S のようなプロトコルにおける出力データラインのトライステート化 1790
図 628. オーバーラン検出エラー 1791
図 629. FIFO アンダーランイベント 1791
図 630. I²C ブロック図 1827
図 631. I2C バスプロトコル 1829
図 632. セットアップおよびホールドタイミング 1831
図 633. I²C 初期化フローチャート 1834
図 634. データ受信 1835
図 635. データ送信 1836
図 636. スレーブ初期化フローチャート 1839
図 637. I²C スレーブ･トランスミッタの転送シーケンス･フローチャート（NOSTRETCH = 0） 1841
図 638. I²C スレーブ･トランスミッタの転送シーケンス･フローチャート（NOSTRETCH = 1） 1842
図 639. I²C スレーブ･トランスミッタの転送バス図 1843
図 640. スレーブ･レシーバの転送シーケンス･フローチャート（NOSTRETCH = 0） 1844
図 641. スレーブ･レシーバの転送シーケンス･フローチャート（NOSTRETCH = 1） 1845
図 642. I²C スレーブ･レシーバの転送バス図 1845
図 643. マスタクロック生成 1847
図 644. マスタ初期化フローチャート 1849
図 645. HEAD10R = 0 のときの 10 ビット･アドレス読出しアクセス 1849
図 646. HEAD10R = 1 のときの 10 ビット･アドレス読出しアクセス 1849
図 647. N ≤ 255 バイトの場合の I²C マスタ･トランスミッタの転送シーケンス･フローチャート 1851
図 648. N > 255 バイトの場合の I²C マスタ･トランスミッタの転送シーケンス･フローチャート 1852
図 649. I²C マスタ･トランスミッタの転送バス図 1853
図 650. N ≤ 255 バイトの場合の I²C マスタ･レシーバの転送シーケンス･フローチャート 1855
図 651. N > 255 バイトの場合の I²C マスタ･レシーバの転送シーケンス･フローチャート 1856

図 652. I^2C マスタ･レシーバの転送バス図 1857
図 653. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔 1861
図 654. N バイト + PEC の場合の SMBus スレーブ･トランスミッタの転送シーケンス･フローチャート 1865
図 655. SMBus スレーブ･トランスミッタの転送バス図（SBC = 1） 1866
図 656. N バイト + PEC の場合の SMBus スレーブ･レシーバの転送シーケンス･フローチャート 1867
図 657. SMBus スレーブ･レシーバのバス転送図（SBC = 1） 1868
図 658. SMBus マスタトランスミッタのバス転送図 1869
図 659. SMBus マスタレシーバのバス転送図 1871
図 660. 独立型ウォッチドッグのブロック図 1893
図 661. ウォッチドッグのブロック図 1903
図 662. ウィンドウ型ウォッチドッグのタイミング図 1904
図 663. CAN サブシステム 1909
図 664. FDCAN ブロック図 1911
図 665. ビットタイミング 1912
図 666. トランシーバ遅延測定 1917
図 667. バス･モニタ･モードでのピン制御 1918
図 668. ループバックモードでのピン制御 1920
図 669. メッセージ RAM 設定 1921
図 670. 標準メッセージ ID フィルタパス 1924
図 671. 拡張メッセージ ID フィルタパス 1925
図 672. USB ペリフェラルのブロック図 1974
図 673. パケットバッファ領域とバッファ記述テーブルの位置の例 1978
図 674. UCPD ブロック図 2008
図 675. クロック分周とタイミング要素 2009
図 676. K コード送信 2011
図 677. さまざまなデータ･サイズの送信順序 2013
図 678. パケット･フォーマット 2013
図 679. Hard Reset のライン･フォーマット 2014
図 680. Cable Reset のライン･フォーマット 2014
図 681. BIST テスト･データ･フレーム 2015
図 682. BIST キャリア･モード 2 2016
図 683. UCPD BMC トランスミッタのアーキテクチャ 2016
図 684. UCPD BMC レシーバのアーキテクチャ 2018
図 685. ブロック図 - STM32 マイクロコントローラおよび FPU 搭載 Cortex®-M4 レベルのデバッグ･サポート 2044
図 686. SWJ デバッグポート 2046
図 687. JTAG TAP 接続 2050
図 688. TPIU ブロック図 2069

1 このマニュアルにおける表記の規則

1.1 一般情報

STM32G4 シリーズ デバイスには、Arm®(a) FPU 搭載 Cortex®-M4 コアが搭載されています。

1.2 レジスタに関する略記

レジスタの説明では、次の略記(b) が使用されます。

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

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

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

1.3 用語

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

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

1.4 製品カテゴリの定義

表 1 は、製品ファミリ別のメモリ容量の概要を示します。

このリファレンスマニュアルには、各製品カテゴリの特徴のスーパーセットが記載されています。カテゴリごとの特徴のリストについては、表 2 を参照してください。

表 1. STM32G4 シリーズメモリ容量

メモリ容量	カテゴリ 2	カテゴリ 3	カテゴリ 4
128 KB	STM32G431 STM32G441（AES）	STM32G471 STM32G473 STM32G474	-
256 KB	-	STM32G471 STM32G473 STM32G474	-
512 KB	-	STM32G471 STM32G473 STM32G474 STM32G483 (AES) STM32G484 (AES)	STM32G491 STM32G4A1 (AES)

1.5 使用可能なペリフェラル

表 2：製品固有の機能に、各製品の最大パッケージを考慮して、すべての STM32G4xx シリーズで使用可能なペリフェラルのリストをまとめます。

すべての販売タイプに用意されているペリフェラルとその番号については、該当するデバイスのデータシートを参照してください。

表 2. 製品固有の機能

機能	STM32G474/ STM32G484	STM32G473/ STM32G483	STM32G471	STM32G431/ STM32G441	STM32G491/ STM32G4A1
Flash	512/256/128K、デュアル·バンク	512/256/128K、デュアル·バンク	512/256/128K、デュアル·バンク	128/64/32K、シングル·バンク	512/256K、シングル·バンク
SRAM1	80K、先頭 32K でパリティ·チェック	80K、先頭 32K でパリティチェック	80K、先頭 32K でパリティチェック	16K、SRAM1 全体でパリティチェック	80K、先頭 32K でパリティ·チェック
SRAM2	16K、パリティチェックなし	16K、パリティチェックなし	16K、パリティチェックなし	6K、パリティチェックなし	16K、パリティチェックなし
CCM SRAM	32K、CCM SRAM 全体でパリティチェック	32K、CCM SRAM 全体でパリティチェック	16K、CCM SRAM 全体でパリティチェック	10K、CCM SRAM 全体でパリティチェック	16K、CCM SRAM 全体でパリティチェック
CRS	あり	あり	あり	あり	あり
DMA	2 DMA コントローラ：DMA1：8 チャネル DMA2：8 チャネル	2 DMA コントローラ：DMA1：8 チャネル DMA2：8 チャネル	2 DMA コントローラ：DMA1：8 チャネル DMA2：8 チャネル	2 DMA コントローラ：DMA1：6 チャネル DMA2：6 チャネル	2 DMA コントローラ：DMA1：8 チャネル DMA2：8 チャネル
DMAMUX	あり	あり	あり	あり	あり
Cordic	あり	あり	あり	あり	あり
FMAC	あり	あり	あり	あり	あり
RNG	あり	あり	あり	あり	あり
AES	あり（注）	あり（注）	なし	あり（注）	あり（注）
CRC	あり	あり	あり	あり	あり
FSMC	あり	あり	なし	なし	なし
QUADSPI	あり	あり	あり	なし	あり
ADC	5 x ADC：ADC1/2 はデュアルモードで使用可能 ADC3/4 はデュアルモードで使用可能 ADC5 はシングルモードでのみ使用可能	5 x ADC：ADC1/2 はデュアルモードで使用可能 ADC3/4 はデュアルモードで使用可能 ADC5 はシングルモードでのみ使用可能	3 x ADC：ADC1/2 はデュアルモードで使用可能 ADC3 はシングルモードでのみ使用可能	2 x ADC：ADC1/2 はデュアルモードで使用可能	3 x ADC：ADC1/2 はデュアル·モードで使用可能、ADC3 はシングル·モードのみ使用可能

表 2. 製品固有の機能 （続き）

機能	STM32G474/ STM32G484	STM32G473/ STM32G483	STM32G471	STM32G431/ STM32G441	STM32G491/ STM32G4A1
DAC	7 DAC ch： DAC1_CH1/ DAC1_CH2/ DAC2_CH1：外部 DAC3_CH1/ DAC3/CH2/ DAC4_CH1/ DAC4_CH2：内部	7 DAC ch： DAC1_CH1/ DAC1_CH2/ DAC2_CH1：外部 DAC3_CH1/ DAC3/CH2/ DAC4_CH1/ DAC4_CH2：内部	4 DAC ch： DAC1_CH1/ DAC1_CH2：外部 DAC3_CH1/ DAC3/CH2：内部	4 DAC ch： DAC1_CH1/ DAC1_CH2：外部 DAC3_CH1/ DAC3/CH2：内部	4 DAC ch： DAC1_CH1/ DAC1_CH2：外部 DAC3_CH1/ DAC3/CH2：内部
COMP	7（COMP1..7）	7（COMP1..7）	4（COMP1..4）	4（COMP1..4）	4（COMP1..4）
OPAMP：	6（OPAMP1..6）	6（OPAMP1..6）	4（OPAMP1.2,3,6）	3（OPAMP1..3）	4（OPAMP1.2,3,6）
VREFBUF	あり	あり	あり	あり	あり
HRTIM1	あり	なし	なし	なし	なし
高機能制御タイマ（TIM1/TIM8/ TIM20）	TIM1/8/20	TIM1/8/20	TIM1/8/20	TIM1/8	TIM1/8/20
汎用タイマ（TIM2/TIM3/ TIM4/TIM5）	TIM2/3/4/5	TIM2/3/4/5	TIM2/3/4	TIM2/3/4	TIM2/3/4
汎用タイマ（TIM15/TIM 16/TIM17）	TIM15/16/17	TIM15/16/17	TIM15/16/17	TIM15/16/17	TIM15/16/17
基本タイマ（TIM6/TIM7）	TIM6/7	TIM6/7	TIM6/7	TIM6/7	TIM6/7
低消費電力タイマ（LPTIM1）	あり	あり	あり	あり	あり
赤外線インタフェース（IRTIM）	あり	あり	あり	あり	あり
独立型ウォッチドッグ（Independent watchdog）（IWDG）	あり	あり	あり	あり	あり
システム・ウィンドウ・ウォッチドッグ (WWDG)	あり	あり	あり	あり	あり
RTC および TAMP	あり	あり	あり	あり	あり
I2C	4 x I2C（I2C1..4）	4 x I2C（I2C1..4）	4 x I2C（I2C1..4）	3 x I2C（I2C1..3）	3 x I2C（I2C1..3）

表 2. 製品固有の機能 （続き）

機能	STM32G474/ STM32G484	STM32G473/ STM32G483	STM32G471	STM32G431/ STM32G441	STM32G491/ STM32G4A1
USART/ UART	3 x USART （USART1..3） 2 x UART （UART4,5）	3 x USART （USART1..3） 2 x UART （UART4,5）	3 x USART （USART1..3） 2 x UART （UART4,5）	3 x USART （USART1..3） 1 x UART （UART4）	3 x USART （USART1..3） 2 x UART （UART4,5）
LPUART	1 x LPUART	1 x LPUART	1 x LPUART	1 x LPUART	1 x LPUART
SPI/I2S	4 x SPI/2 x I2S （SPI1..4 - I2S2,3）	4 x SPI/2 x I2S （SPI1..4 - I2S2,3）	4 x SPI/2 x I2S （SPI1..4 - I2S2,3）	3 x SPI/2 x I2S （SPI1..3 - I2S2,3）	3 x SPI/2 x I2S （SPI1..3 - I2S2,3）
SAI	1 x SAI	1 x SAI	1 x SAI	1 x SAI	1 x SAI
FDCAN	3 x FDCAN （FDCAN1..3）	3 x FDCAN （FDCAN1..3）	2 x FDCAN （FDCAN1,2）	1 x FDCAN （FDCAN1）	2 x FDCAN （FDCAN1,2）
USB デバイス	1 x USB デバイス	1 x USB デバイス	1 x USB デバイス	1 x USB デバイス	1 x USB デバイス
UCPD1	1 x UCPD	1 x UCPD	1 x UCPD	1 x UCPD	1 x UCPD

注： AES は STM32G483xx、STM32G484xx、STM32G441x、STM32G4A1 デバイスでのみ使用できます。

2 システムおよびメモリの概要

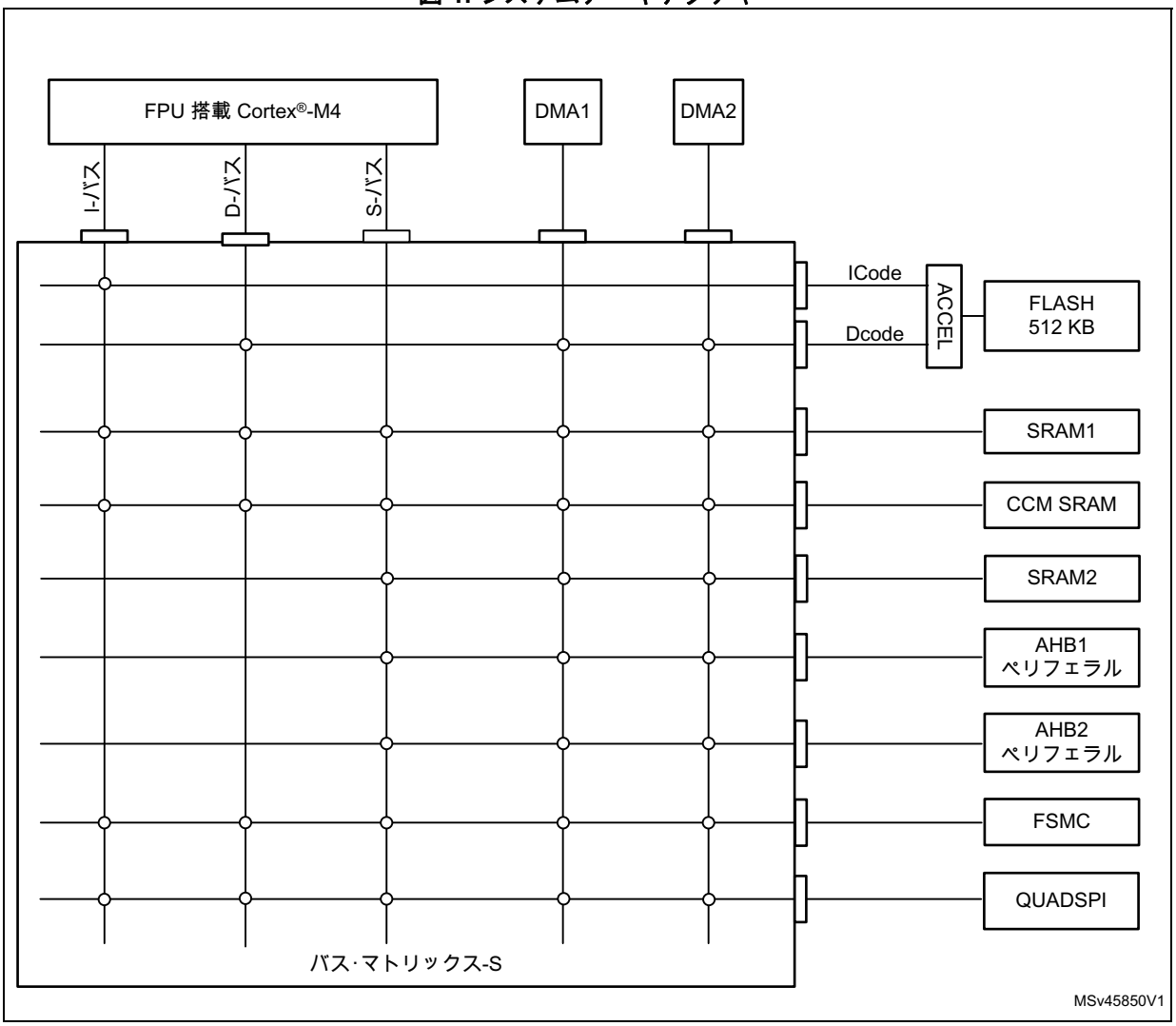
2.1 システムアーキテクチャ

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

- 最大 5 つのマスタ：
  - FPU 搭載 Cortex®-M4 コア I バス
  - FPU 搭載 Cortex®-M4 コア D バス
  - FPU 搭載 Cortex®-M4 コア S バス
  - DMA1
  - DMA2
- 最大 9 つのスレーブ：
  - ICode バス上の内蔵Flashメモリ
  - DCode バス上の内蔵Flashメモリ
  - 内蔵 SRAM1
  - 内蔵 SRAM2
  - 内蔵 CCM SRAM
  - AHB－APB ブリッジおよび APB ペリフェラルを含む AHB1 ペリフェラル（APB1 および APB2 に接続）
  - AHB2 ペリフェラル
  - フレキシブル･スタティック･メモリ･コントローラ（FSMC）
  - QUAD SPI メモリ･インタフェース（QUADSPI）

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

図 1. システムアーキテクチャ



2.1.1 I-バス

このバスによって、FPU 搭載 Cortex®-M4 コアの命令バスがバスマトリックスに接続されます。コアは、このバスを用いて命令をフェッチします。このバスの対象は、コードを保持するメモリです。(内蔵 Flash メモリ、内蔵 SRAM または FSMC もしくは QUADSPI を経由した外部メモリのいずれか)

2.1.2 D-バス

このバスによって、FPU 搭載 Cortex®-M4 コアのデータバスがバスマトリックスに接続されます。このバスは、コアによってリテラルロードとデバッグアクセスに使用されます。このバスの対象は、コードを保持するメモリです。(内蔵 Flash メモリ、内蔵 SRAM または FSMC もしくは QUADSPI を経由した外部メモリのいずれか)

2.1.3 S-バス

このバスによって、FPU 搭載 Cortex®-M4 コアのシステムバスがバスマトリックスに接続されます。コアがこのバスを使用してペリフェラルまたは SRAM 領域内のデータにアクセスします。このバス

の対象となるのは、内蔵 SRAM、APB1 および APB2 ペリフェラルを含む AHB1 ペリフェラル、AHB2 ペリフェラル、および QUADSPI または FSMC を経由した外部メモリです。

CCM SRAM もこのバスでアクセスでき、SRAM1 および SRAM2 と連続的な配置が可能です。

2.1.4 DMA バス

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

2.1.5 バスマトリックス

このバスマトリックスによって、マスタ間のアクセス調停を管理します。調停には、ラウンドロビン方式を使用します。バスマトリックスは 最大 5 つのマスタ（CPU AHB、システムバス、DCode バス、ICode バス、DMA1、DMA2）と最大 9 つのスレーブ（Flash、SRAM1、SRAM2、CCM SRAM、AHB1（APB1 および APB2 を含む）、AHB2、QUADSPIおよび FSMC）で構成されています。

AHB/APB ブリッジ

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

このブリッジに接続されたペリフェラルのアドレスマッピングについては、81 ページのセクション 2.2.2：メモリマップとレジスタ境界アドレスを参照してください。

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

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

2.2 メモリ構成

2.2.1 概要

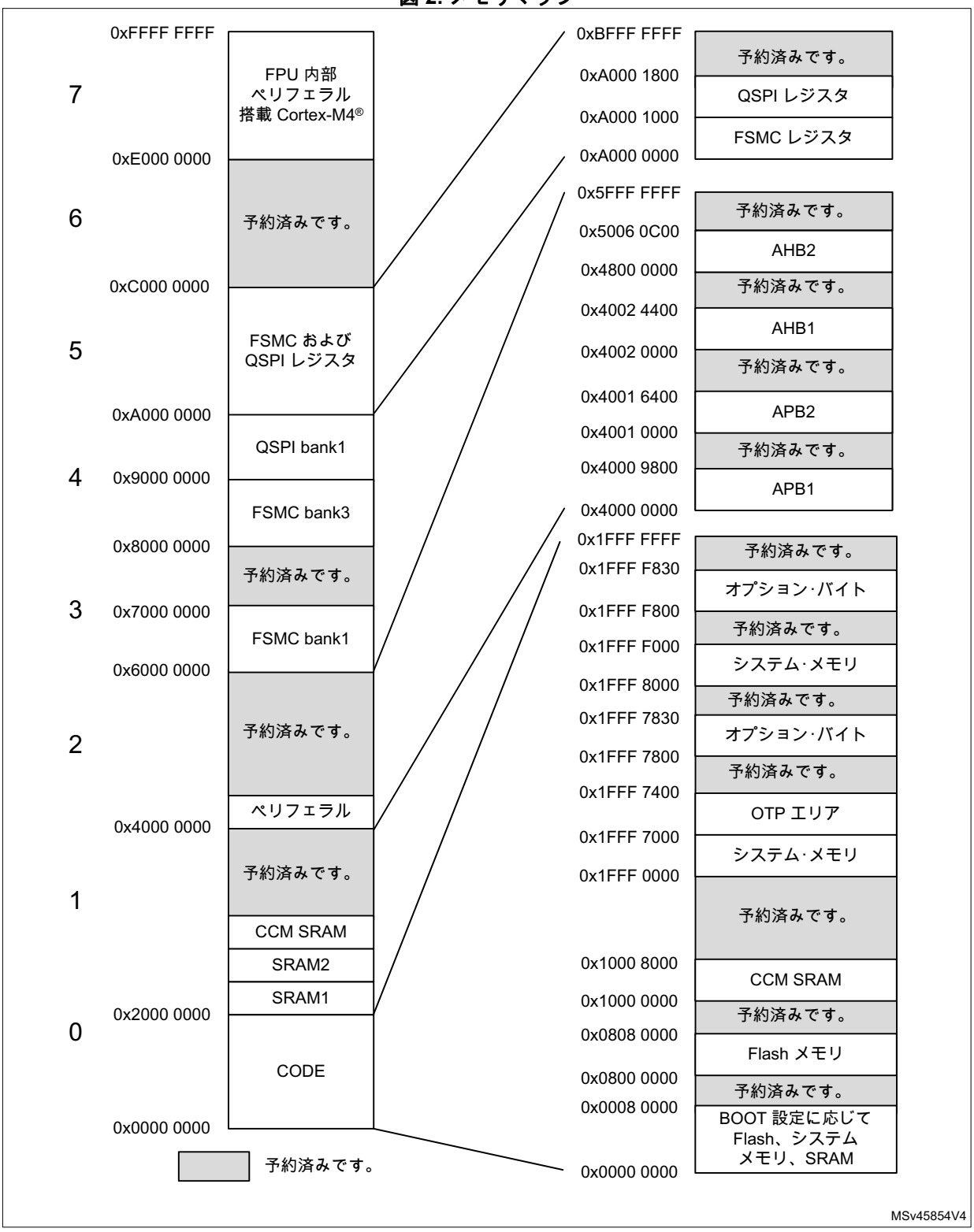
プログラムメモリ、データメモリ、レジスタ、および I/O ポートは、同じ 4 GB のリニアなアドレス空間に配置されています。

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

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

2.2.2 メモリマップとレジスタ境界アドレス

図 2. メモリマップ



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

次の表に、デバイスで使用可能なペリフェラルの境界アドレスを示します。

表 3. STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス(1)

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラル・レジスタ・マップ
-	0xA000 1400 - 0xAFFF FFFF	262 MB	予約済み	-
	0xA000 1000 - 0xA000 13FF	1 KB	QUADSPI 制御レジスタ	セクション 20.5.14 : QUADSPI レジスタマップ
	0xA000 0400 - 0xA000 0FFF	3 KB	予約済み	-
	0xA000 0000 - 0xA000 03FF	1 KB	FSMC	セクション 19.7.8 : FMC レジスタ･マップ
AHB2	0x5006 0C00 - 0x5FFF FFFF	256 MB	予約済み	-
	0x5006 0800 - 0x5006 0BFF	1 KB	RNG	セクション 26.7.4 : RNG レジスタ･マップ
	0x5006 0400 - 0x5006 07FF	1 KB	予約済み	-
	0x5006 0000 - 0x5006 03FF	1 KB	AES	セクション 34.7.18 : AES レジスタ･マップ
	0x5000 1800 - 0x5005 FFFF	377 KB	予約済み	-
	0x5000 1400 - 0x5000 17FF	1 KB	DAC4	セクション 22.7.24:D/A コンバータ･レジスタ･マップ
	0x5000 1000 - 0x5000 13FF	1 KB	DAC3	セクション 22.7.24:D/A コンバータ･レジスタ･マップ
	0x5000 0C00 - 0x5000 0FFF	1 KB	DAC2	セクション 22.7.24:D/A コンバータ･レジスタ･マップ
	0x5000 0800 - 0x5000 0BFF	1 KB	DAC1	セクション 22.7.24:D/A コンバータ･レジスタ･マップ
	0x5000 0400 - 0x5000 07FF	1 KB	ADC3 - ADC4 - ADC5	セクション 21.9 : A/D コンバータ･レジスタ･マップ
	0x5000 0000 - 0x5000 03FF	1 KB	ADC1 - ADC2	セクション 21.9 : A/D コンバータ･レジスタ･マップ
	0x4800 1C00 - 0x4FFF FFFF	127 MB	予約済み	-
	0x4800 1800 - 0x4800 1BFF	1 KB	GPIOG	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 1400 - 0x4800 17FF	1 KB	GPIOF	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 1000 - 0x4800 13FF	1 KB	GPIOE	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 0C00 - 0x4800 0FFF	1 KB	GPIOD	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 0800 - 0x4800 0BFF	1 KB	GPIOC	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 0400 - 0x4800 07FF	1 KB	GPIOB	セクション 9.4.12 : GPIO レジスタマップ
	0x4800 0000 - 0x4800 03FF	1 KB	GPIOA	セクション 9.4.12 : GPIO レジスタマップ

表 3. STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス(1) （続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラル・レジスタ・マップ
AHB1	0x4002 3400 - 0x47FF FFFF	127 MB	予約済み	-
	0x4002 3000 - 0x4002 33FF	1 KB	CRC	セクション 16.4.6 : CRC レジスタマップ
	0x4002 2400 - 0x4002 2FFF	3 KB	予約済み	-
	0x4002 2000 - 0x4002 23FF	1 KB	Flash インタフェース	セクション 5.7.14 : Flash レジスタマップ
	0x4002 1400 - 0x4002 1FFF	3 KB	FMAC	セクション 18.4.9 : FMAC レジスタマップ
	0X4002 1000 - 0x4002 13FF	1 KB	RCC	セクション 7.4.31 : RCC レジスタマップ
	0x4002 0C00 - 0x4002 0FFF	1 KB	CORDIC	セクション 17.4.4 : CORDIC レジスタマップ
	0x4002 0800 - 0x4002 0BFF	1 KB	DMAMUX	セクション 13.6.7 : DMAMUX レジスタ･マップ
	0x4002 0400 - 0x4002 07FF	1 KB	DMA 2	セクション 12.6.7 : DMA レジスタ･マップ
	0x4002 0000 - 0x4002 03FF	1 KB	DMA 1	セクション 12.6.7 : DMA レジスタ･マップ
APB2	0x4001 7800 - 0x4001 FFFF	2 KB	予約済み	-
	0x4001 6800 - 0x4001 77FF	3 KB	HRTIM	セクション 27.5.82 : HRTIM レジスタマップ
	0x4001 5800 - 0x4001 67FF	4 KB	予約済み	-
	0x4001 5400 - 0x4001 57FF	1 KB	SAI1	セクション 40.6.19 : SAI レジスタ･マップ
	0x4001 5000 - 0x4001 53FF	1 KB	TIM20	セクション 28.6.31 : TIMx レジスタマップ
	0x4001 4C00 - 0x4001 4FFF	1 KB	予約済み	-
	0x4001 4800 - 0x4001 4BFF	1 KB	TIM17	セクション 30.8.22 : TIM16/TIM17 レジスタマップ
	0x4001 4400 - 0x4001 47FF	1 KB	TIM16	セクション 30.8.22 : TIM16/TIM17 レジスタマップ
	0x4001 4000 - 0x4001 43FF	1 KB	TIM15	セクション 30.7.23 : TIM15 レジスタマップ
	0x4001 3C00 - 0x4001 3FFF	1 KB	SPI4	セクション 39.9.10 : SPI/I2S レジスタマップ
	0x4001 3800 - 0x4001 3BFF	1 KB	USART1	セクション 37.8.15 : USART レジスタマップ
	0x4001 3400 - 0x4001 37FF	1 KB	TIM8	セクション 28.6.31 : TIMx レジスタマップ
	0x4001 3000 - 0x4001 33FF	1 KB	SPI1	セクション 39.9.10 : SPI/I2S レジスタマップ
	0x4001 2C00 - 0x4001 2FFF	1 KB	TIM1	セクション 28.6.31 : TIMx レジスタマップ
	0x4001 0800 - 0x4001 2BFF	9 KB	予約済み	-
	0x4001 0400 - 0x4001 07FF	1 KB	EXTI	セクション 15.5.13 : EXTI レジスタマップ
	0x4001 0300 - 0x4001 03FF	1 KB	OPAMP :	セクション 25.5.13 : OPAMP レジスタ･マップ
	0x4001 0200 - 0x4001 02FF		COMP	セクション 24.6.2 : COMP レジスタマップ
	0x4001 0030 - 0x4001 01FF		VREFBUF	セクション 23.4.3 : VREFBUF レジスタ･マップ
	0x4001 0000 - 0x4001 0029		SYSCFG	セクション 10.2.11 : SYSCFG レジスタマップ

表 3. STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス(1) （続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラル・レジスタ・マップ
APB1	0x4000 AFFE - 0x4000 FFFF	23 KB	予約済み	-
	0x4000 AC00 - 0x4000 AFFF	1 KB	FDCAN メッセージ RAM	セクション 44.4.38 : FDCAN レジスタ·マップ
	0x4000 A800 - 0x4000 ABFF	1 KB		
	0x4000 A400 - 0x4000 A7FF	1 KB		
	0x4000 A000 - 0x4000 A3FF	1 KB	UCPD1	セクション 46.7.15 : UCPD レジスタ·マップ
	0x4000 8800 - 0x4000 9FFF	6 KB	予約済み	-
	0x4000 8400 - 0x4000 87FF	1 KB	I2C4	セクション 41.7.12 : I2C レジスタ·マップ
	0x4000 8000 - 0x4000 83FF	1 KB	LPUART1	セクション 38.7.13 : LPUART レジスタマップ
	0x4000 7C00 - 0x4000 7FFF	1 KB	LPTIM1	セクション 32.7.10 : LPTIM レジスタマップ
	0x4000 7800 - 0x4000 7BFF	1 KB	I2C3	セクション 41.7.12 : I2C レジスタ·マップ
	0x4000 7400 - 0x4000 77FF	1 KB	予約済み	-
	0x4000 7000 - 0x4000 73FF	1 KB	PWR	セクション 6.4.23:PWR レジスタマップとリセット値の表
	0x4000 6C00 - 0x4000 6FFF	1 KB	FDCAN3	セクション 44.4.38 : FDCAN レジスタ·マップ
	0x4000 6800 - 0x4000 6BFF	1 KB	FDCAN2	セクション 44.4.38 : FDCAN レジスタ·マップ
	0x4000 6400 - 0x4000 67FF	1 KB	FDCAN1	セクション 44.4.38 : FDCAN レジスタ·マップ
	0x4000 6000 - 0x4000 63FF	1 KB	USB SRAM 1 KB	-
	0x4000 5C00 - 0x4000 5FFF	1 KB	USB デバイス FS	セクション 45.6.3 : USB レジスタマップ
	0x4000 5800 - 0x4000 5BFF	1 KB	I2C2	セクション 41.7.12 : I2C レジスタ·マップ
	0x4000 5400 - 0x4000 57FF	1 KB	I2C1	セクション 41.7.12 : I2C レジスタ·マップ
	0x4000 5000 - 0x4000 53FF	1 KB	UART5	セクション 37.8.15 : USART レジスタマップ
	0x4000 4C00 - 0x4000 4FFF	1 KB	UART4	セクション 37.8.15 : USART レジスタマップ
	0x4000 4800 - 0x4000 4BFF	1 KB	USART3	セクション 37.8.15 : USART レジスタマップ
	0x4000 4400 - 0x4000 47FF	1 KB	USART2	セクション 37.8.15 : USART レジスタマップ

表 3. STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス(1) （続き）

バス	境界アドレス	サイズ（バイト）	ペリフェラル	ペリフェラル・レジスタ・マップ
APB1 続き	0x4000 4000 - 0x4000 43FF	1 KB	予約済み	-
	0x4000 3C00 - 0x4000 3FFF	1 KB	SPI3/I2S3	セクション 39.9.10 : SPI/I2S レジスタマップ
	0x4000 3800 - 0x4000 3BFF	1 KB	SPI2/I2S2	セクション 39.9.10 : SPI/I2S レジスタマップ
	0x4000 3400 - 0x4000 37FF	1 KB	予約済み	-
	0x4000 3000 - 0x4000 33FF	1 KB	IWDG	セクション 42.4.6 : IWDG レジスタマップ
	0x4000 2C00 - 0x4000 2FFF	1 KB	WWDG	セクション 43.5.4 : WWDG レジスタマップ
	0x4000 2800 - 0x4000 2BFF	1 KB	RTC & BKP レジスタ	セクション 35.6.21 : RTC レジスタマップ
	0x4000 2400 - 0x4000 27FF	1 KB	TAMP	セクション 36.6.9 : TAMP レジスタマップ
	0x4000 2000 - 0x4000 23FF	1 KB	CRS	セクション 8.7.5 : CRS レジスタマップ
	0x4000 1C00 - 0x4000 1FFF	1 KB	予約済み	-
	0x4000 1800 - 0x4000 1BFF	1 KB	予約済み	-
	0x4000 1400 - 0x4000 17FF	1 KB	TIM7	セクション 29.5.31 : TIMx レジスタマップセクション 31.4.9 : TIMx レジスタマップ
	0x4000 1000 - 0x4000 13FF	1 KB	TIM6	セクション 31.4.9 : TIMx レジスタマップ
	0x4000 0C00 - 0x4000 0FFF	1 KB	TIM5	セクション 29.5.31 : TIMx レジスタマップ
	0x4000 0800 - 0x4000 0BFF	1 KB	TIM4	セクション 29.5.31 : TIMx レジスタマップ
	0x4000 0400 - 0x4000 07FF	1 KB	TIM3	セクション 29.5.31 : TIMx レジスタマップ
	0x4000 0000 - 0x4000 03FF	1 KB	TIM2	セクション 29.5.31 : TIMx レジスタマップ

1. お使いのデバイスで使用可能な GPIO ポートとペリフェラルについては、表 1 : STM32G4 シリーズメモリ容量および表 2 : 製品固有の機能と、デバイスデータシートを参照してください。使用できない GPIO ポートやペリフェラルに対応するメモリ領域は予約済み（グレイでハイライトされています）。

2.3 ビットバンディング

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

STM32G4 シリーズデバイスでは、ペリフェラルレジスタも SRAM もビットバンド領域に割り当てられますので、1 回のビットバンドの書込みと読出し操作が許可されます。操作は FPU 搭載 Cortex®-M4 アクセスでのみ可能であり、他のバスマスタ（DMA など）からはできません。

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

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

ここで、

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

例

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

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

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

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

2.4 内蔵 SRAM

STM32G4 シリーズカテゴリ 3 デバイスは、最大 128 KBの SRAMを搭載しています。

- 80 KB SRAM1（アドレス 0x2000 0000 に配置）
- 16 KB SRAM2（アドレス 0x2001 4000 に配置）
- 32 KB CCM SRAM（アドレス 0x1000 0000 と SRAM2 の最後に配置）

STM32G4 シリーズのカテゴリ 4 デバイスは、最大 112 KBの SRAMを搭載しています。

- 80 KB SRAM1（アドレス 0x2000 0000 に配置）
- 16 KB SRAM2（アドレス 0x2001 4000 に配置）
- 16 KB CCM SRAM（アドレス 0x1000 0000 と SRAM2 の最後に配置）

STM32G4 シリーズカテゴリ 2 デバイスは、最大 32 KBの SRAMを搭載しています。

- 16 KB SRAM1（アドレス 0x2000 0000 に配置）
- 6 KB SRAM2（アドレス 0x2000 4000 に配置）
- 10 KB CCM SRAM（アドレス 0x1000 0000 と SRAM2 の最後に配置）

これらの SRAM には、バイト、ハーフワード（16 ビット）、またはフルワード（32 ビット）によるアクセスが可能です。これらのメモリは、最大システムクロック周波数（ウェイトステートなし）でCPUとDMAの両方でアクセスできます。

CPU は、SRAM1 からのブートが選択されている場合、または物理的な再割当てが選択されている場合に、システムバスまたは ICode／DCode バスを通じて SRAM1 にアクセスできます（SYSCFG コントローラの セクション 10.2.1 : SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP））。SRAM1 実行のパフォーマンスを最大にするには、物理的な再割当てを選択します（ブートまたはソフトウェア選択）。

CCM SRAM はアドレス 0x1000 0000 に配置されます。

ICode バスを通じてアクセスするため、再割当てなく CCM SRAM から最大のパフォーマンスで実行できます。

CCM SRAMは、SRAM2 の最後（カテゴリ 2 デバイスでは 0x2000 5800、カテゴリ 3 デバイスでは 0x2001 8000、カテゴリ 4 デバイスでは 0x2001 8000）の次のアドレスにエイリアスされ、SRAM1 および SRAM2 に連続したアドレス空間を提供します。

2.4.1 パリティチェック

カテゴリ 3 およびカテゴリ 4 デバイスでは、パリティ･チェックは SRAM1 の最初の 32 KB と CCM SRAM 全体に対して実装されます。

カテゴリ 2 デバイスでは、パリティチェックは SRAM1 と CCM SRAM の全体に対して実装されます。

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

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

パリティビットは、SRAM に書き込む際に計算され、格納されます。その後、読み出すときに自動的にチェックされます。1 つのビットがフェイルすると、NMI が生成されます。同じエラーが、TIM1/TIM8/TIM15/TIM16/TIM17/TIM20 の BRK_IN ブレーク入力および hrtim_sys_flt に、セクション 10.2.8：SYSCFG 設定レジスタ 2（SYSCFG_CFGR2）の SPL の制御ビットにより、リンクする事もまたできます。SRAM パリティ・エラー・フラグ（SPF）は セクション 10.2.8：SYSCFG 設定レジスタ 2（SYSCFG_CFGR2）で使用可能です。

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

2.4.2 CCM SRAM 書込み保護

CCM SRAM は、1 KB のページ単位で書込み保護できます。

表 4. CCM SRAM の構成

ページ番号	開始アドレス	終了アドレス
ページ 0	0x1000 0000	0x1000：03FF
ページ 1	0x1000 0400	0x1000：07FF
ページ 2	0x1000 0800	0x1000：0BFF
ページ 3	0x1000：0C00	0x1000：0FFF
ページ 4	0x1000 1000	0x1000：13FF
ページ 5	0x1000 1400	0x1000：17FF
ページ 6	0x1000 1800	0x1000：1BFF
ページ 7	0x1000：1C00	0x1000：1FFF
ページ 8	0x1000 2000	0x1000：23FF
ページ 9	0x1000 2400	0x1000：27FF
ページ 10(1)	0x1000 2800	0x1000：2BFF
ページ 11 (1)	0x1000：2C00	0x1000：2FFF
ページ 12 (1)	0x1000 3000	0x1000：33FF
ページ 13 (1)	0x1000 3400	0x1000：37FF

表 4. CCM SRAM の構成 （続き）

ページ番号	開始アドレス	終了アドレス
ページ 14 (1)	0x1000 3800	0x1000：3BFF
ページ 15 (1)	0x1000：3C00	0x1000：3FFF
ページ 16(2)	0x1000 4000	0x1000：43FF
ページ 17 (2)	0x1000 4400	0x1000：47FF
ページ 18 (2)	0x1000 4800	0x1000：4BFF
ページ 19 (2)	0x1000：4C00	0x1000：4FFF
ページ 20 (2)	0x1000 5000	0x1000：53FF
ページ 21 (2)	0x1000 5400	0x1000：57FF
ページ 22 (2)	0x1000 5800	0x1000：5BFF
ページ 23 (2)	0x1000：5C00	0x1000：5FFF
ページ 24 (2)	0x1000 6000	0x1000：63FF
ページ 25 (2)	0x1000 6400	0x1000：67FF
ページ 26 (2)	0x1000 6800	0x1000：6BFF
ページ 27 (2)	0x1000：6C00	0x1000：6FFF
ページ 28 (2)	0x1000 7000	0x1000：73FF
ページ 29 (2)	0x1000 7400	0x1000：77FF
ページ 30 (2)	0x1000 7800	0x1000：7BFF
ページ 31 (2)	0x1000：7C00	0x1000：7FFF

1. カテゴリ 3 およびカテゴリ 4 デバイスでのみ使用できます。
2. カテゴリ 3 デバイスでのみ使用できます。

書込み保護は、SYSCFG ブロックの セクション 10.2.9：SYSCFG CCM SRAM 書込み保護レジスタ（SYSCFG_SWPR）で有効にすることができます。これは、「1」を 1 回だけ書き込むメカニズムを持つレジスタで、ビットに「1」を書き込むことで、システムリセットでのみ削除/消去可能な書込み保護を SRAM の当該ページに設定します。

2.4.3 CCM SRAM 読出し保護

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

2.4.4 CCM SRAM の消去

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

CCM SRAM 消去は、セクション 10.2.7：SYSCFG CCM SRAM 制御およびステータスレジスタ（SYSCFG_SCSR）の CCMSR ビットをセットすることにより、ソフトウェアでリクエストすることもできます。

2.5 Flashメモリの概要

Flashメモリは 2 つの別個の物理領域で構成されます。

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

Flashインタフェースは、AHB プロトコルに基づいて命令アクセスとデータアクセスを実装します。また、Flashメモリ操作（プログラム／消去）を実行するために必要な、Flashレジスタを使って制御されるロジックも実装します。詳細については セクション 3 : 内蔵 Flash メモリ（Flash）: カテゴリ 3 デバイスの場合、セクション 4 : 内蔵 Flash メモリ（Flash）カテゴリ 4 デバイスの場合、セクション 5 : 内蔵 Flash メモリ（Flash）カテゴリ 2 デバイスの場合 を参照してください。

2.6 ブート設定

2.6.1 ブート設定

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

表 5. ブート･モード

BOOT_LOCK	nBOOT1 Flash_OPTR[23]	nBOOT0 Flash_OPTR[27]	BOOT0 ピン PB8	nSWBOOT0 Flash_OPTR[26]	ブートメモリ空間 エイリアス
1	X	X	X	X	メインFlashメモリ
0	X	X	0	1	メインFlashメモリがブート領域として選択されます。
0	X	1	X	0	メインFlashメモリがブート領域として選択されます。
0	0	X	1	1	内蔵 SRAM1 がブート領域として選択されます。
0	0	0	X	0	内蔵 SRAM1 がブート領域として選択されます。
0	1	X	1	1	システムメモリがブート領域として選択されます。
0	1	0	X	0	システムメモリがブート領域として選択されます。

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

BOOT0 ピンまたはユーザオプションビット（Flash_OPTR レジスタの nSWBOOT0 ビットの値による）および nBOOT1 ビットも、STANDBY モードの終了時に再サンプリングされます。したがって、これらのピンは STANDBY モードのときでも必要なブート･モード設定に保たれる必要があります。

この起動遅延時間が終了すると、CPU はアドレス 0x0000 0000 からスタック最上位の値をフェッチし、0x0000 0004 のブートメモリからコード実行を開始します。

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

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

PB8/BOOT0 GPIO は次で設定されます。

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

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

メインFlashメモリからブートする際、アプリケーションソフトウェアをバンク 1 またはバンク 2 のいずれかからブートできます（カテゴリ 3 デバイスのみ）。デフォルトでは、バンク 1 からのブートが選択されています。

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

異なる RDP レベルのブートローダ機能については、表 13 : アクセス状態 対 保護レベルと実行モードを参照してください。

ユーザFlashメモリからの強制ブート

ブート設定にかかわらず、メイン Flash メモリの一意のエントリポイントからブートを強制することができます。内蔵 Flash メモリ（Flash）のセクションを参照してください。

物理的な再割当て

ブートピンモードを選択したら、アプリケーションソフトウェアはコード領域でアクセス可能なメモリを変更できます（このようにするとシステムバスの代わりに ICode バスを使ってコードを実行できます）。この変更は、SYSCFG コントローラでセクション 10.2.1 : SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP）をプログラミングすることによって実行できます。

その後、次のメモリが再割当て可能となります。

- メインFlashメモリ
- システムメモリ
- 内蔵 SRAM1
- FSMC バンク 1（NOR/PSRAM 1 および 2）

- QUADSPI メモリ

表 6. メモリ割当てとブートモード／物理的な再割当て(1)

アドレス	メインFlash メモリでの ブート／ 再割当て	内蔵 SRAM1 での ブート／ 再割当て	システム メモリでの ブート／ 再割当て	FSMC での 再割当て	QUADSPI での 再割当て
0x2000 0000 - 0x2002 3FFF	SRAM1	SRAM1	SRAM1	SRAM1	SRAM1
0x1FFF 7000 - 0x1FFF FFFF	システム メモリ／OTP／ オプション バイト	システム メモリ／OTP／ オプション バイト	システム メモリ／OTP／ オプション バイト	システム メモリ／OTP／ オプション バイト	システム メモリ／OTP／ オプション バイト
0x1000 8000 - 0x1FFE FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x1000 0000 - 0x1000 7FFF	CCM SRAM	CCM SRAM	CCM SRAM	CCM SRAM	CCM SRAM
0x0808 0000 - 0x0FFF FFFF	予約済み	予約済み	予約済み	予約済み	予約済み
0x0800 0000 - 0x0807 FFFF	Flashメモリ	Flashメモリ	Flashメモリ	Flashメモリ	Flashメモリ
0x0400 0000 - 0x07FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/ PSRAM 2 （128 MB）、 エイリアス	QUADSPI バンク （128 MB）、 エイリアス
0x0010 0000 - 0x03FF FFFF	予約済み	予約済み	予約済み	FSMC バンク 1 NOR/ PSRAM 1 （128 MB）、 エイリアス	QUADSPI バンク （128 MB）、 エイリアス
0x0000 0000 - 0x0007 FFFF (2) (3)	Flash、 エイリアス	SRAM1、 エイリアス	システム メモリ（28 KB）、 エイリアス	FSMC バンク 1 NOR/ PSRAM 1 （128 MB）、 エイリアス	QUADSPI、 エイリアス

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

内蔵ブートローダ

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

3 内蔵 Flash メモリ（Flash）：カテゴリ 3 デバイスの場合

3.1 概要

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

また、命令プリフェッチおよびキャッシュラインでコードの実行を加速します。

3.2 Flash の主な機能

- 最大容量 512 KB の Flash メモリ、書込み中読出し（RWW）機能に対応するデュアル･バンク･アーキテクチャ
- 次の 2 つのデータ幅に対応したFlashメモリ読出し操作
  - シングルバンクモード（DBANK = 0）：128 ビットの読出しアクセス
  - デュアルバンクモード（DBANK = 1）：64 ビットの読出しアクセス
- ページ消去、バンク消去、全体消去（両バンク）を実装

Flashメモリインタフェースの機能：

- Flashメモリ読出し操作
- Flashメモリプログラム／消去操作
- オプション（RDP）で有効化される読出し保護
- 4 つの書込み保護領域（DBANK = 1 のとき各バンクに 2 つ、DBANK = 0 のときメモリ全体に 4 つ）
- 2 つの独自仕様コード読出し保護領域（DBANK = 1 のとき各バンクに 1 つ、DBANK = 0 のときメモリ全体に 2 つ）
- オプションで定義される 2 つのセキュリティ保護可能メモリ領域（DBANK = 1 のとき各バンクに 1 つ、DBANK = 0 のときメモリ全体に 2 つ）
- ICode でプリフェッチ
- 命令キャッシュ：ICode で 4 x 64 または 2 x 128 ビットのキャッシュライン 32 本（1 KB RAM）
- データキャッシュ：ICode で 4 x 64 または 2 x 128 ビットのキャッシュライン 8 本（256 バイト RAM）
- エラーコード訂正 ECC：64 ビットダブルワードごとに 8 ビット
  - DBANK = 1 の場合：8 + 64 = 72 ビット、検出 2 ビット、訂正 1 ビット
  - DBANK = 0 の場合：(8+64) + (8+64) = 144 ビット、検出 2 ビット、訂正 1 ビット
- オプションバイトローダ
- 低電力モード

3.3 Flash の機能説明

3.3.1 Flash メモリの構成

Flashメモリは、以下のような主な特徴を備えています。

- シングルバンクモード（読出し幅 128 ビット）またはデュアルバンクモード（読出し幅 64 ビット）に最大容量 512 KB を構成します。
- BFB2 オプションビットにより、デュアルブートモードに対応しています（デュアルバンクモードのみ）。
- DBANK ビットが設定されている場合はデュアルバンクモード：
  - メインメモリの 2 つのバンクに 512 KB を構成
  - ページサイズ 2 KB
  - 72 ビット幅のデータ読出し（64 ビット + ECC 8 ビット）
  - バンクおよび全体消去
- DBANK ビットがリセットされている場合はシングルバンクモード：
  - メインメモリの 1 つのシングルバンクに 512 KB を構成
  - ページサイズ 4 KB
  - 144 ビット幅のデータ読出し（128 ビット + ECC 2x8 ビット）
  - 全体消去

Flashメモリは、次のように構成されています。

- メインメモリブロックは、デュアルバンク設定ビットに応じて次のように構成されます。
  - デュアルバンクが有効な場合（DBANK ビットが設定されている）、Flashは 256 KB の 2 つのバンクに分割されます。各バンクの構成は以下のとおりです。

    メイン･メモリ･ブロックには 2 KB のページが 128 ページ含まれます。

    各ページには 256 KB の行が 8 行含まれます。
  - デュアルバンクが無効な場合（DBANK ビットがリセットされている）、メインメモリブロックは 512 KB の 1 つのバンクで構成されます。

    メインメモリブロックには 4 KB のページが 128 ページ含まれます。

    各ページには 512 KB の行が 8 行含まれます。
- 情報ブロックには次のものが含まれます。
  - システムメモリブートモードでデバイスがブートするシステムメモリ。この領域はSTMicroelectronics によって予約済みで、次のいずれかのインタフェースを介したFlashメモリの再プログラムに使用するブートローダを含みます。USART、SPI、I2C、FDCAN、USB。これはデバイスの製造時に STMicroelectronics によってプログラムされており、誤った書込み／消去操作から保護されています。詳細については、www.st.com から入手可能な AN2606 を参照してください。
  - OTP（1 度だけプログラムが可能な）、ユーザデータ用の 1 KB（128 ダブルワード）。OTP 領域はバンク 1 でのみ使用できます。OTP データは消去不可で、1 度だけ書込み可能です。1 つのビットだけが 0 の場合、0x0000 0000 0000 0000 の値であってもダブルワード全体を書き込むことはできません。
  - ユーザ設定のオプションバイト。

メモリ構成は、表 28 に示されるメイン領域と情報ブロックに基づきます。

表 7. Flash モジュール - 512/256/128 KB のデュアル･バンク構成（読出し幅 64 ビット）

Flash 領域		Flashメモリアドレス	サイズ（バイト）	名前
メイン･メモリ（512/256/128 KB）	バンク 1(1)（256/128/64 KB）	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
		0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
		0x0800 1000 - 0x0800 17FF	2 K	ページ 2
		0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
		- - - -	- - - -	- - - -
		0x0803 F800 - 0x0803 FFFF	2 K	ページ 127
	バンク 2 (1)（256/128/64 KB）	0x0804 0000 - 0x0804 07FF	2 K	ページ 0
		0x0804 0800 - 0x0804 0FFF	2 K	ページ 1
		0x0804 1000 - 0x0804 17FF	2 K	ページ 2
		0x0804 1800 - 0x0804 1FFF	2 K	ページ 3
		- - - -	- - - -	- - - -
		0x0807 F800 - 0x0807 FFFF	2 K	ページ 127
情報ブロック	バンク 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	バンク 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	バンク 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	バンク 1	0x1FFF 7800 - 0x1FFF 782F	48	オプションバイト
	バンク 2	0x1FFF F800 - 0x1FFF F82F	48	

1. 256 KB デバイスの場合：ページ 0 ~ ページ 63
128 KB デバイスの場合：ページ 0 ~ ページ 31

表 8. Flash モジュール - 512/256/128 KB のシングル･バンク構成（読出し幅 128 ビット）

Flash 領域		Flashメモリアドレス	サイズ（バイト）	名前
メインメモリ(1)（512/256/128 KB）		0x0800 0000 - 0x0800 0FFF	4 K	ページ 0
		0x0800 1000 - 0x0800 1FFF	4 K	ページ 1
		0x0800 2000 - 0x0800 2FFF	4 K	ページ 2
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		-	-	-
		0x0807 F000 - 0x0807 FFFF	4 K	ページ 127
情報ブロック	バンク 1	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	バンク 2	0x1FFF 8000 - 0x1FFF EFFF	28 K	
	バンク 1	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	バンク 1	0x1FFF 7800 - 0x1FFF 782F	48	オプションバイト
	バンク 2	0x1FFF F800 - 0x1FFF F82F	48	

1. 256 KB デバイスの場合：ページ 0 ~ ページ 63
128 KB デバイスの場合：ページ 0 ~ ページ 31

3.3.2 エラーコード訂正（ECC）

デュアルバンクモード（DBANK = 1、データ幅 64 ビット）

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

- 1 つのエラー検出および訂正
- 2 つのエラー検出

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

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

ECC エラーが検出されると、エラーを含むダブルワードのアドレスおよびその関連バンクが Flash_ECCR レジスタの ADDR_ECC[20:0] と BK_ECC に保存されます。ADDR_ECC[2:0] は常にクリアされます。

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

シングル･バンク･モード（DBANK = 0、データ幅 128 ビット）

Flashメモリのデータは 144 ビットワードです。ダブルワード単位で 8 ビット追加されます。ECC メカニズムは次のものをサポートしています。

- 1 つのエラー検出および訂正
- 64 ビットダブルワードごとに 2 つのエラー検出

まず SYSF_ECC ビットを確認し、それがセットされていたら、DBANK = 1 のプログラミングモデルを参照する必要があります（システムFlashは常に 2 つのバンク上にあるため）。そのビットがセットされていない場合は、次のプログラミングモデルを参照してください。

ダブルワード（ビット [63:0] および ビット [127:64]）ごとに ECC があります。

64 LSB ビット（ビット [63:0] ）で 1 つのエラーが検出され訂正されたときに、Flash_ECCR レジスタのフラグ ECCC（ECC 訂正）がセットされます。

64 MSB ビット（ビット [127:64] ）で 1 つのエラーが検出され訂正されたときに、FLASH_ECCR レジスタのフラグ ECCC2（ECC2 訂正）がセットされます。

ECCCIE がセットされている場合は、割込みが生成されます。ECCC および ECCC2 を読み出して、128 ビットデータのどの部分が訂正されたのか（[63:0]、[127:64]、または両方）を確認する必要があります。

64 LSB ビットで 2 つのエラーが検出されたときに、Flash_ECCR レジスタのフラグ ECCD（ECC 検出）がセットされます。

64 MSB ビット（ビット [127:64] ）で 2 つのエラーが検出されたときに、FLASH_ECCR レジスタのフラグ ECCD2（ECC2 検出）がセットされます。

この場合、NMI が生成されます。ECCD および ECCD2 を読み出して、128 ビットデータのどの部分でエラーが検出されたのか（[63:0]、[127:64]、または両方）を確認する必要があります。

ECC エラーが検出されると、2 回失敗したダブルワードのアドレスが Flash_ECCR レジスタの ADDR_ECC[20:0] に保存されます。ADDR_ECC[20:0] には、この 2 回のダブルワードのアドレスが保存されます。

ADDR_ECC[3:0] は常にクリアされます。BK_ECC はこのモードでは使用しません。

ECCC/ECCC2 または ECCD/ECCD2 がセットされていると、新しい ECC エラーが発生した場合に ADDR_ECC が更新されません。Flash_ECCR は、ECC フラグ（ECCC/ECCC2/ECCD/ECCD2）がクリアされた場合のみ更新されます。

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

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

3.3.3 読出しアクセスの遅延

データをFlashメモリから正しく読み出すには、CPU クロック（HCLK）の周波数およびデバイスの内部電圧範囲 V_{CORE} に従って、Flash アクセス制御レジスタ（FLASH_ACR）でウェイトステート（LATENCY）の数を正しくプログラムする必要があります。セクション 6.1.5：ダイナミック電圧スケーリングの管理 を参照してください。表 9 には、ウェイトステートと CPU クロック周波数の対応を示します。

表 9. CPU クロック（HCLK）周波数によるウェイトステート数

ウェイトステート（WS）（LATENCY）	HCLK (MHz)		
	V_{CORE} レンジ 1 ブースト·モード	V_{CORE} レンジ 1 通常モード	V_{CORE} レンジ 2
0 WS（1 CPU サイクル）	≤ 34	≤ 30	≤ 12
1 WS（2 CPU サイクル）	≤ 68	≤ 60	≤ 24
2 WS（3 CPU サイクル）	≤ 102	≤ 90	≤ 26
3 WS（4 CPU サイクル）	≤ 136	≤ 120	-
4 WS（5 CPU サイクル）	≤ 170	≤ 150	-

リセット後、CPU クロック周波数は 16 MHz であり、Flash_ACR レジスタでは 1 ウェイトステート（WS）が設定されます。

CPU 周波数を変更するときは、Flash メモリにアクセスするために必要なウェイトステート数を適切な値にするために、以下のソフトウェア·シーケンスを適用する必要があります。

CPU 周波数の増加：

1. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
2. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が反映されていることを確認します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロックプリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が減少する場合には、最初にそのステップを実行してからもう一方を実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。

CPU 周波数の減少：

1. RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更します。
2. 必要であれば、RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロックプリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が増加する場合には、最初にもう一方のステップを実行してからそのステップを実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。
5. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
6. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が使用されていることを確認します。

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

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

プロセッサのフルパフォーマンスを引き出すため、アクセラレータは命令プリフェッチキューと、64 ビットのFlashメモリからのプログラム実行速度を上昇させるブランチキャッシュを実行します。CoreMark ベンチマークによると、ART アクセラレータによって達成されるパフォーマンスは、最大 CPU 周波数 170 MHz におけるFlashメモリからの 0 ウェイトステートプログラム実行と同等となります。

命令プリフェッチ

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

シングルバンクモード（DBANK オプションビットがリセット）の場合、各Flashメモリ読出し操作では、起動されるプログラムに応じて、32 ビットの命令 4 個または16 ビットの命令 8 個の 128 ビットが提供されます。この 128 ビットの現在の命令ラインは、現在のバッファに保存されますので、連続コードの場合は、その前の読出し命令ラインの実行に 4 個以上の CPU サイクルが必要となります。

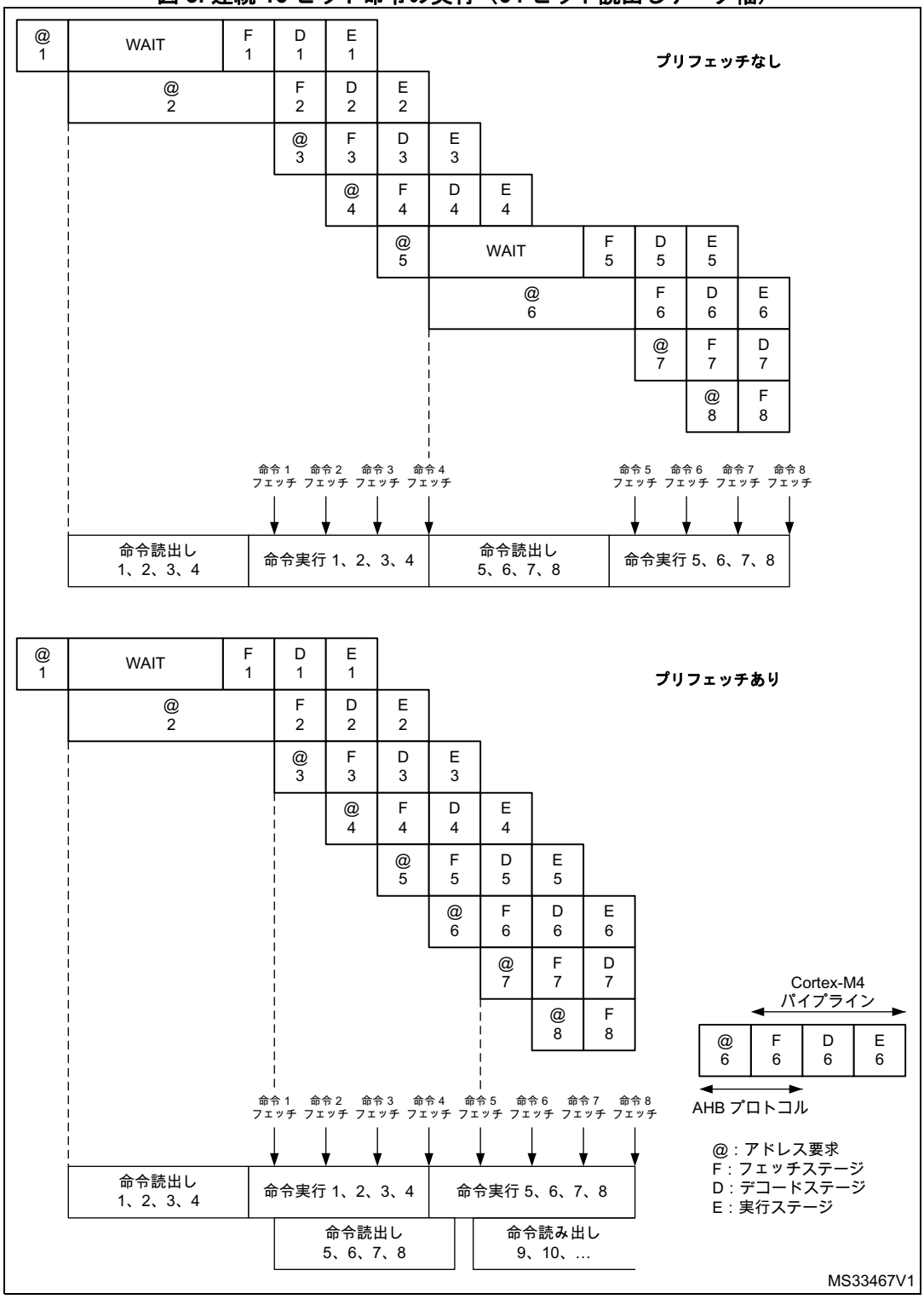
デュアルバンクモード（DBANK オプションビットがセット）の場合、各Flashメモリ読出し操作では、起動されるプログラムに応じて、32 ビットの命令 2 個または16 ビットの命令 4 個の 64 ビットが提供されます。この 64 ビットの現在の命令ラインは、現在のバッファに保存されますので、連続コードの場合は、その前の読出し命令ラインの実行に 2 個以上の CPU サイクルが必要となります。

ICode バスでのプリフェッチを使用すると、CPU によって現在の命令ラインが要求されている間にFlashメモリから次の連続命令ラインを読み出すことができます。

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

図 9 は、Flashメモリへのアクセスに 3 WS が必要なときに、プリフェッチがある場合とない場合の連続 16 ビット命令の実行を示しています。

図 3. 連続 16 ビット命令の実行（64 ビット読出しデータ幅）



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

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

命令キャッシュメモリ（I キャッシュ）

ジャンプによる時間のロスを制限するため、デュアルバンクモードでは 4 x 64 ビットの 32 ライン、シングルバンクモードでは 2 x 128 ビットの 32 ラインを命令キャッシュメモリ内に維持できます。この機能は、Flash アクセス制御レジスタ（FLASH_ACR）の命令キャッシュ有効（ICEN）ビットをセットすると有効にできます。失敗が発生（現在使用している命令ライン、プリフェッチされた命令ラインまたは命令キャッシュメモリに要求されたデータがない）するたびに、ラインの読出しが命令キャッシュメモリにコピーされます。命令キャッシュメモリに含まれるデータの中に CPU が要求するデータがある場合には、全く遅延なしにそのデータが提供されます。すべての命令キャッシュメモリラインが満たされると、LRU (最も長い時間使われていない) ポリシーを使用して命令メモリキャッシュの中で置換するラインを決定します。この機能は、ループを含むコードの場合に特に有用です。

命令キャッシュメモリはシステムリセット後に有効になります。

データキャッシュメモリ（D キャッシュ）

リテラルプールは、CPU パイプラインの実行ステージにおいて DCode バスを通じてFlashメモリからフェッチされます。各 DCode バスの読出しアクセスは、現在のバッファに保存される 64 ビットまたは 128 ビットをフェッチします。その結果として、CPU パイプラインは要求されるリテラルプールが提供されるまでストールされます。リテラルプールによる時間のロスを制限するため、AHB データバス DCodeからのアクセスは、AHB命令バス ICode からのアクセスよりも優先されます。

頻繁に使用されるリテラルプールがある場合は、Flash アクセス制御レジスタ（FLASH_ACR）のデータキャッシュ有効（DCEN）ビットをセットすると、データキャッシュメモリを有効にできます。この機能は、命令キャッシュメモリのように動作しますが、保持されるデータのサイズは、デュアルバンクモードでは 4*64 ビット 8 ライン、シングルバンクモードでは 2*128 ビット 8 ラインに制限されます。

データキャッシュメモリはシステムリセット後に有効になります。

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

オプションバイトブロックのデータは、キャッシュできません。

3.3.5 Flashのプログラムおよび消去操作

STM32G4 シリーズ 内蔵Flashメモリは、インサーキットプログラミングまたはアプリケーション内プログラミングを使用してプログラム可能です。

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

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

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

進行中のFlashメモリ操作は、CPU が同じFlashメモリバンクにアクセスしない限り、CPU をブロックしません。コードまたはデータのフェッチは、1 つのバンクで書込み／消去操作を実施している間にもう 1 つのバンクで可能です（セクション 3.3.8 : デュアル･バンク･モード（DBANK = 1）でのみ使用可能な書込み中読出し（RWW）を参照してください）。

Flashメモリの消去およびプログラムは電圧スケーリングレンジ 1 でのみ可能です。PWR_CR1 の VOS[1:0] ビットは 01b に設定してください。

逆に、Flashメモリへの消去／プログラム操作中に同じFlashメモリバンクを読み出そうとすると、バスがストールします。読出し操作は、プログラム／消去操作が完了すると正しく処理されます。

Flashメモリのロック解除

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

1. Flash キー レジスタ（FLASH_KEYR）に KEY1 = 0x45670123 を書き込みます。
2. Flash_KEYR レジスタに KEY2 = 0xCDEF89AB を書き込みます。

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

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

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

3.3.6 Flashメインメモリの消去シーケンス

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

ページ消去

ページを消去するには、次の手順に従います。

1. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. デュアル･バンク･モード（DBANK オプション･ビットがセットされているモード）では、PER ビットをセットし、消去するページ（PNB）を、関連するバンク（BKER）とともに Flash 制御レジスタ（Flash_CR）で選択します。シングル･バンク･モード（DBANK オプション･ビットがリセットされているモード）では、PER ビットをセットし、消去するページ（PNB）を選択します。Flash 制御レジスタ（Flash_CR）の BKER ビットをクリアされた状態に保つ必要があります。
4. Flash_CR レジスタの STRT ビットをセットします。
5. Flash_SR レジスタの BSY ビットがクリアされるまで待ちます。

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

消去するページが、（WRP または PCROP によって）書込み保護された領域に属している場合、WRPERR がセットされてページ消去リクエストが中止されます。

バンク 1、バンク 2 の全体消去（DBANK = 1 のデュアルバンクモードでのみ使用可能）

バンクの全体消去を実行するには、次の手順に従います。

1. Flash_SR レジスタの BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. Flash制御レジスタ（Flash_CR）の MER1 ビットまたは MER2（バンクによる）をセットします。どちらのバンクも同じ操作で選択できますが、その場合は全体消去に相当します。
4. FLACH_CR レジスタの STRT ビットをセットします。
5. Flashステータスレジスタ（Flash_SR）の BSY ビットがクリアされるまで待ちます。

全体消去

全体消去を実行するには、次の手順に従います。

1. Flash_SR レジスタの BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー·プログラミング·フラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. Flash 制御レジスタ（Flash_CR）の MER1 ビットまたは MER2 ビットをセットします。
4. FLACH_CR レジスタの STRT ビットをセットします。
5. Flashステータスレジスタ（Flash_SR）の BSY ビットがクリアされるまで待ちます。

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

DBANK = 0 のとき、MERA または MERB ビットのみがセットされている場合は、PGSERR がセットされ、消去操作は行われません。

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

3.3.7 Flashメインメモリのプログラミングシーケンス

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

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

ダブルワード（2 x 32 ビットデータ）のプログラムのみ可能です。

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

標準プログラミング

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

1. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中のFlashメインメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。

3. Flash制御レジスタ（Flash_CR）の PG ビットをセットします。
4. メインメモリブロックまたは OTP エリア内の指定したメモリアドレスでデータ書込み操作を実施します。ダブルワードのみプログラム可能です。
   – ダブルワードに合ったアドレスで最初のワードを書き込みます。
   – 2 番目のワードを書き込みます。
5. Flash_SR レジスタの BSY ビットがクリアされるまで待ちます。
6. Flash_SR レジスタの EOP フラグ（プログラミング操作が成功したことを示します）がセットされていることをチェックし、ソフトウェアによってクリアします。
7. プログラミング･リクエストがなくなったら、Flash_SR レジスタの PG ビットをクリアします。

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

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

ECC は、プログラムするダブルワードから計算されます。

1 行（DBANK = 1 の場合の 64 ダブルワード）またはハーフ行（DBANK = 0 の場合の 64 ダブルワード）に対する高速プログラミング

このモードでは、1 行（DBANK = 1 の場合の 64 ダブルワード）またはハーフ行（DBANK = 0 の場合の 64 ダブルワード）のプログラミングが可能であり、プログラム前に Flash 位置を確認する必要をなくしてページプログラミングの時間を削減し、各ダブルワードに対する高電圧の立上がり時間と立下り時間を回避できます。高速プログラミング中は、CPU クロック周波数（HCLK）は少なくとも 8 MHz でなければなりません。

メインメモリのみ高速プログラミングモードでプログラム可能です。

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

1. シングルバンクモード（DBANK = 0）の場合、全体消去を実行します。しない場合、PGSERR がセットされます。高速プログラミングが実行可能であるのは、コードが RAM またはブートローダから実行される場合に限られます。デュアルバンクモード（DBANK = 1）の場合、プログラムするバンクの全体消去を実行します。しない場合、PGSERR がセットされます。
2. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中の Flash メイン･メモリ操作がないことを確認します。
3. 以前のプログラミングによるエラープログラミングフラグをすべて確認してクリアします。
4. Flash制御レジスタ（Flash_CR）の FSTPG ビットをセットします。
5. 64 ダブルワードを書き込んで、1 行またはハーフ行をプログラムします。ダブルワードのみプログラム可能です。
   – ダブルワードに合ったアドレスで最初のワードを書き込みます。
   – 2 番目のワードを書き込みます。
6. Flash_SR レジスタの BSY ビットがクリアされるまで待ちます。
7. Flash_SR レジスタの EOP フラグ（プログラミング操作が成功したことを示します）がセットされていることをチェックし、ソフトウェアによってクリアします。
8. プログラミング･リクエストがなくなったら、Flash_SR レジスタの FSTPG ビットをクリアします。

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

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

64 ダブルワードは連続で書き込む必要があります。高電圧は、すべてのプログラミングに対してFlashで維持されます。連続した 2 つのダブルワードの間の書込みリクエストの最大時間は、最初のダブルワードに対して、Flashがプログラムされている時間（2 x 25 us 前後）です。2 番目のダブルワードがこのプログラムされている時間を越える場合、高速プログラミングは割り込まれ、MISSERR がセットされます。

一行に対する書込み時、高電圧のかかる時間が 8 ms を越えてはなりません。これは 8MHz 以上のクロックシステムで連続で書き込まれた 64 ダブルワードのシーケンスによって保証されます。高速プログラミングがセットされると、内部タイムアウトカウンタが 7 ms をカウントし、タイムアウトを過ぎるとプログラミングが停止します。この場合、FASTERR ビットがセットされます。

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

プログラミングエラー

複数種類のエラーが検出されます。エラーが発生した場合、Flash操作（プログラミングまたは消去）は中止されます。

- **PROGERR**：プログラミングエラー

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

- **SIZERR**：サイズ・プログラミング・エラー

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

- **PGAERR**：アドレス整列プログラミング・エラー

  次のいずれかの条件が発生した場合、PGAERR がセットされます。

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

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

- **PGSERR**：プログラミング・シーケンス・エラー

  次のいずれかの条件が発生した場合、PGSERR がセットされます。

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

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

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

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

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

- 以前のプログラミングエラーによって PROGERR、SIZERR、PGAERR、WRPERR、MISSERR、FASTERR、または PGSERR がセットされた場合にも、PGSERR がセットされます。
- DBANK = 0 のとき、MER1 または MER2 のみがセットされていると PGSERR がセットされます（バンクの全体消去はできません）。

- **WRPERR**：書込み保護エラー

  次のいずれかの条件が発生した場合、WRPERR がセットされます。
  - 書込み保護領域（WRP）、PCROP 領域、またはセキュリティ保護可能メモリ領域でプログラミングまたは消去を試みた場合
  - 1 つ以上のページが WRP または PCROP で保護されているときに バンク消去の実行を試みた場合
  - 読出し保護（RDP）がレベル 1 にセットされている状態で、デバッグ機能が接続されている場合または SRAM やシステムFlashからブートされる場合
  - 読出し保護（RDP）がレベル 2 にセットされている状態で、オプションバイトの変更を試みた場合
- **MISSERR**：高速プログラミングデータミスエラー

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

  高速プログラミング：次のいずれかの条件が発生した場合、FASTERR がセットされます。
  - タイムアウト検出を生成する 7 ms を超える時間に FSTPG ビットがセットされる場合
  - 高速プログラミングが MISSERR、PGAERR、WRPERR、または SIZERR に割り込まれた場合

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

PROGERR、SIZERR、PGAERR、PGSERR、MISSERR（プログラム･エラー･フラグ）、WRPERR（保護エラーフラグ）

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

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

プログラミングとキャッシュ

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

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

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

3.3.8 デュアル･バンク･モード（DBANK = 1）でのみ使用可能な書込み中読出し（RWW）

デュアル･バンク･モードは、DBANK オプション･ビットがセットされた場合のみ使用でき、書込み中読出し操作が可能になります。この機能により、1 つのバンクで消去操作やプログラム操作を実施している間にもう 1 つのバンクから読出し操作を実施できます。

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

バンク 2 ページ消去中のバンク 1 での読出し（またはその逆）

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

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

バンク 2 全体消去中のバンク 1 での読出し（またはその逆）

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

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

バンク 2 プログラミング中のバンク 1 での読出し（またはその逆）

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

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

3.4 Flashオプションバイト

3.4.1 オプションバイトの説明

オプションバイトは、アプリケーション要件によってエンドユーザが設定します。設定の例として、ウォッチドッグをハードウェアまたはソフトウェアモードに選択できます（セクション 5.4.2：オプションバイトのプログラミングを参照してください）。

ダブルワードは、オプションバイトで次に示すように分割されます。

表 10. オプションバイトのフォーマット

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

情報ブロック内のこれらのバイトの構成は、表 31：オプションバイトの構成に示すとおりです。

オプションバイトは、表 31：オプションバイトの構成にリストされたメモリ位置またはオプションバイトレジスタから読み出すことができます。

- Flashオプションレジスタ（Flash_OPTR）
- Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）
- Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）
- Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）
- Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）
- Flash PCROP2 開始アドレスレジスタ（Flash_PCROP2SR）
- Flash PCROP2 終了アドレスレジスタ（Flash_PCROP2ER）
- Flashバンク 2 WRP 領域 A アドレスレジスタ（Flash_WRP2AR）
- （Flashバンク 2 WRP 領域 B アドレスレジスタ（Flash_WRP2BR））

表 11. オプション･バイトの構成

バンク	アドレス	[63:56]	[55:48]	[47:40]	[39:32]	[31:24]	[23:16]	[15:8]	[7:0]
バンク 1	1FFF7800	USER OPT			RDP	USER OPT			RDP
	1FFF7808	未使用		未使用および PCROP1_STRT[14:0]		未使用		未使用および PCROP1_STRT[14:0]	
	1FFF7810	PCROP_RDP および 未使用		未使用および PCROP1_END[14:0]		PCROP_RDP および 未使用		未使用および PCROP1_END[14:0]	
	1FFF7818	未使用	WRP1A_END [6:0]	未使用	WRP1A_STRT [6:0]	未使用	WRP1A_END [6:0]	未使用	WRP1A_STRT [6:0]
	1FFF7820	未使用	WRP2A_END [6:0]	未使用	WRP2A_STRT [6:0]	未使用	WRP2A_END [6:0]	未使用	WRP2A_STRT [6:0]
	1FFF7828	未使用	BOOT_LOCK	未使用	SEC_SIZE1	未使用	BOOT_LOCK	未使用	SEC_SIZE1

表 11. オプション･バイトの構成 （続き）

バンク	アドレス	[63:56]	[55:48]	[47:40]	[39:32]	[31:24]	[23:16]	[15:8]	[7:0]
バンク 2	1FFFF800	未使用							
	1FFFF808	未使用		未使用および PCROP2_STRT[14:0]		未使用		未使用および PCROP2_STRT[14:0]	
	1FFFF810	未使用		未使用および PCROP2_END[14:0]		未使用		未使用および PCROP2_END[14:0]	
	1FFFF818	未使用	WRP2B_END [6:0]	未使用	WRP2B_STRT [6:0]	未使用	WRP2B_END [6:0]	未使用	WRP2B_STRT [6:0]
	1FFFF820	未使用	WRP2B_END [6:0]	未使用	WRP2B_STRT [6:0]	未使用	WRP2B_END [6:0]	未使用	WRP2B_STRT [6:0]
	1FFFF828	未使用			SEC_SIZE2	未使用			SEC_SIZE2

ユーザオプションバイトおよび読出し保護オプションバイト

Flashメモリアドレス：0x1FFF 7800

ST 出荷時の値：0xFFEF F8AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRH_EN	NRST_MODE[1:0]		nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	DBANK	Res.	BFB2	WWDG_SW	IWGD_STDBY	IWDG_STOP	IWDG_SW
	r	r	r	r	r	r	r	r	r		r	r	r	r	r

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **IRH_IN**：PG10の内部リセットホルダ

0：IRH は無効です。

1：IRH は有効です。

ビット 29:28 **NRST_MODE[1:0]**：PG10 パッド･モード

00：入出力をリセット

01：入力のみをリセット

10：GPIO

11：入出力をリセット

ビット 27 **nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24 **SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル
0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。
1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

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

ビット 22 **DBANK**：
0：128 ビット･データ読出し幅でのシングル･バンク･モード
1：64 ビット･データでのデュアル･バンク･モード
このビットに書き込めるのは、PCROP1/2 が無効なときだけです。

ビット 21 予約済みであり、リセット値に保持する必要があります。

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

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択
0：ウィンドウ型ウォッチドッグがハードウェアで起動します。
1：ウィンドウ型ウォッチドッグがソフトウェアで起動します。

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

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

ビット 16 **IWDG_SW**：独立型ウォッチドッグの選択
0：独立型ウォッチドッグがハードウェアで起動します。
1：独立型ウォッチドッグがソフトウェアで起動します。

ビット 15 予約済みであり、リセット値に保持する必要があります。

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

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

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

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

ビット 7:0 **RDP**：読出し保護レベル

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

PCROP1 開始アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7808

リセット値：0xFFFF FFFF（ST 出荷時の値）

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_STRT[14:0]**：PCROP 領域開始オフセット

DBANK = 1
PCROP1_STRT は、バンク 1 の PCROP 領域の最初のダブルワードを含んでいます。
DBANK = 0
PCROP1_STRT は、すべてのメモリの PCROP 領域の最初の 2 つのダブルワードを含んでいます。

PCROP1 終了アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7810

リセット値：0x00FF 0000（ST 出荷時の値）

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

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

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

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

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

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_END[14:0]**：バンク 1 PCROP 領域終了オフセット

DBANK = 1
PCROP1_END は、バンク 1 の PCROP 領域の最後のダブルワードを含んでいます。
DBANK = 0
PCROP1_END は、すべてのメモリの PCROP 領域の最後の 2 つのダブルワードを含んでいます。

WRP1 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7818

リセット値：0xFF00 FFFF（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [6:0]						
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [6:0]						
									r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP1A_END[6:0]**：WRP 最初の領域「A」終了オフセット

DBANK = 1

WRP1A_END は、バンク 1 の WRP 最初の領域の最終ページを含んでいます。

DBANK = 0

WRP1A_END は、すべてのメモリの WRP 最初の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP1A_STRT[6:0]**：WRP 最初の領域「A」開始オフセット

DBANK = 1

WRP1A_STRT は、バンク 1 の WRP 最初の領域の最初のページを含んでいます。

DBANK = 0

WRP1A_STRT は、すべてのメモリの WRP 最初の領域の最初のページを含んでいます。

WRP2 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7820

リセット値：0xFF00 FFFF（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END [6:0]						
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT [6:0]						
									r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP2A_END[6:0]**：WRP 2 番目の領域「B」終了オフセット

DBANK = 1

WRP2A_END は、バンク 1 の WRP 2 番目の領域の最終ページを含んでいます。

DBANK = 0

WRP2A_END は、すべてのメモリの WPR 2 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP2A_STRT[6:0]**：WRP 2 番目の領域開始オフセット

DBANK = 1

WRP2A_STRT は、バンク 1 の WRP 2 番目の領域の最終ページを含んでいます。

DBANK = 0

WRP2A_STRT は、すべてのメモリの WPR 2 番目の領域の最終ページを含んでいます。

セキュリティ保護可能メモリ領域バンク 1 オプションバイト

Flashメモリアドレス：0x1FFF7828

リセット値：0xFF00 FF00（ST 出荷時の値）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [7:0]							
								r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SEC_SIZE1[7:0]**：セキュリティ保護可能なメモリ領域のサイズ
セキュリティ保護可能な Flash メモリのページ数を含んでいます。

PCROP2 開始アドレスオプションバイト

Flashメモリアドレス：0x1FFFF808

リセット値：0xFFFF FFFF（ST 出荷時の値）

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP2_STRT[14:0]**：PCROP 領域開始オフセット
DBANK = 1
PCROP2_STRT は、バンク 2 の PCROP 領域の最初のダブルワードを含んでいます。
DBANK = 0
PCROP2_STRT は、すべてのメモリの PCROP 領域の最初のダブルワードを含んでいます。

PCROP2 終了アドレスオプションバイト

Flashメモリアドレス：0x1FFF F810

リセット値：0x00FF 0000（ST 出荷時の値）

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP2_END[14:0]**：PCROP 領域終了オフセット

DBANK = 1

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

DBANK = 0

PCROP2_END は、すべてのメモリの PCROP 領域の最後の 2 つのダブルワードを含んでいます。

WRP1 領域 B アドレスオプションバイト

Flashメモリアドレス：0x1FFF F818

リセット値：0xFF00 FFFF（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END [6:0]						
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT [6:0]						
									r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP1B_END[6:0]**：WRP 1 番目の領域「B」終了オフセット

DBANK = 1

WRP1B_END は、バンク 2 の WRP 最初の領域の最終ページを含んでいます。

DBANK = 0

WRP1B_END は、すべてのメモリの WRP 3 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP1B_STRT[6:0]**：WRP 1 番目の領域「B」開始オフセット

DBANK = 1

WRP1B_STRT は、バンク 2 の WRP 最初の領域の最初のページを含んでいます。

DBANK = 0

WRP1B_STRT は、すべてのメモリの WRP 3 番目の領域の最初のページを含んでいます。

WRP2 領域 B アドレスオプションバイト

Flashメモリアドレス：0x1FFF F820

リセット値：0xFF00 FFFF（ST 出荷時の値）

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP2B_END[6:0]**：WRP 2 番目の領域「B」終了オフセット

DBANK = 1

WRP2B_END は、バンク 2 の WRP 2 番目の領域の最終ページを含んでいます。

DBANK = 0

WRP2B_END は、すべてのメモリの WRP 4 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP2B_STRT[6:0]**：WRP 2 番目の領域「B」開始オフセット

DBANK = 1

WRP2B_STRT は、バンク 2 の WRP 2 番目の領域の最初のページを含んでいます。

DBANK = 0

WRP2B_STRT は、すべてのメモリの WRP 2 番目の領域の最初のページを含んでいます。

セキュリティ保護可能メモリ領域バンク 2 オプションバイト

Flashメモリアドレス：0x1FFF F828

リセット値：0xFF00 FF00（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE2 [7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SEC_SIZE2[7:0]**：セキュリティ保護可能メモリ領域サイズには、セキュリティ保護可能な Flash メモリのページ数が含まれています。

3.4.2 オプション･バイトのプログラミング

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

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

ソフトウェアによって OPTLOCK ビットをセットすると、ユーザ･オプションを意図せぬ消去やプログラム操作から保護できます。

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

ユーザオプションの変更

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

ユーザオプション値を変更するには、次の手順に従います。

1. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 上記のクリアシーケンスで OPTLOCK オプションロックビットをクリアします。
3. オプション値を次のオプションレジスタに書き込みます。Flashオプションレジスタ（Flash_OPTR）, Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）, Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）, Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）, Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）, Flash PCROP2 開始アドレスレジスタ（Flash_PCROP2SR）, Flash PCROP2 終了アドレスレジスタ（Flash_PCROP2ER）, Flashバンク 2 WRP 領域 A アドレスレジスタ（Flash_WRP2AR）, Flashバンク 2 WRP 領域 B アドレスレジスタ（Flash_WRP2BR）
4. Flash制御レジスタ（Flash_CR）でオプションスタートビット OPTSTRT をセットします。
5. BSY ビットがクリアされるのを待ちます。

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

オプションバイトローディング

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

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

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

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

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

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

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

システムリセットの立ち上がりでは、内部オプションレジスタがソフトウェアで読出しおよび書込み可能なオプションレジスタ（Flash_OPTR、Flash_PCROP1/2SR、Flash_PCROP1/2ER、Flash_WRP1/2AR、Flash_WRP1/2BR）にコピーされます。これらのレジスタはオプション変更にも使用されます。これらのレジスタがユーザによって変更されない場合、システムのオプション状態が反映されます。詳細は、セクション ：ユーザオプションの変更を参照してください。

デュアルバンクモードの有効化（DBANK = 0 から DBANK = 1 への切替え）

あるFlashモードから別のFlashモードへ切り替えるには（たとえば、シングルバンクからデュアルバンクへ）、SRAM からコードを実行するか、あるいはブートローダを使用することを推奨します。メモリ構成が変更されたときにFlashから破損データを読み出さないようにするには、再プログラムする前のFlashメモリへのアクセス（CPU または DMA）は避けるべきです。

- 命令キャッシュとデータキャッシュ、プリフェッチが有効化されている場合、無効化します（Flash_ACR レジスタの PRFTEN ビットと ICEN/DCEN ビットをリセットします）。
- Flash_ACR レジスタの DCRST/ICRST ビットをセットして、命令キャッシュとデータ･キャッシュを一掃します。
- DBANK オプション･ビットをセットし、すべての WRP 書込み保護をクリアします（ユーザ･オプション変更およびオプション･バイト･ローダ手順に従います）。
  - DBANK = 1 で OBL が実行されると、全体消去を実行します。
  - DBANK = 1 のメモリマッピングを使用して、64 ビットモードで新しいコードプログラミングを開始します。
  - 必要であれば、DBANK = 1 スキームを用いて新しい WRP/PCROP を設定します。
  - 必要であれば、PRFTEN と ICEN/DCEN を設定します。

新しいソフトウェアは、バンク設定を使用して起動する準備ができています。

デュアルバンクモードの無効化（DBANK = 1 から DBANK = 0 への切替え）

あるFlashモードから別のFlashモードへ切り替えるには（たとえば、シングルバンクからデュアルバンクへ）、SRAM からコードを実行するか、あるいはブートローダを使用することを推奨します。メモリ構成が変更されたときにFlashから破損データを読み出さないようにするには、再プログラムする前のFlashメモリへのアクセス（CPU または DMA）は避けるべきです。

- 命令キャッシュとデータキャッシュ、プリフェッチが有効化されている場合、無効化します（Flash_ACR レジスタの PRFTEN ビットと ICEN/DCEN ビットをリセットします）。
- Flash_ACR レジスタの DCRST/ICRST ビットをセットして、命令キャッシュとデータ･キャッシュを一掃します。
- DBANK オプション･ビットおよびすべての WRP 書込み保護をクリアします（ユーザ･オプション変更およびオプション･バイト･ローダ手順に従います）。
  - DBANK = 0 で OBL が実行されると、全体消去を実行します。
  - DBANK = 0 のメモリマッピングを使用して、128 ビットモードで新しいコードプログラミングを開始します。
  - 必要であれば、DBANK = 0 スキームを用いて新しい WRP/PCROP を設定します。必要であれば、PRFTEN と ICEN/DCEN を設定します。

新しいソフトウェアは、バンク設定を使用して起動する準備ができています。

3.5 Flashメモリの保護

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

3.5.1 読出し保護（RDP）

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

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

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

Flashメモリは、RDP オプションバイトとその補数に 表 32に示す値のペアが含まれる場合に保護されます。

表 12. Flashメモリの読出し保護ステータス

RDP バイト値	RDP 補数	読出し保護レベル
0xAA	0x55	レベル 0（ST 出荷時の設定）
0xAA または 0xCC を除く任意の値	0x55 および 0x33 以外の任意の値（相補である必要はない）	レベル 1
0xCC	0x33	レベル 2

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

レベル 0：保護なし

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

レベル 1：読出し保護

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

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

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

レベル 2：デバッグなし

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

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

注： **リセット後、デバッグ機能は無効化されます。**

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

読出し保護レベルの変更

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

レベル 1 からレベル 0 に移動するために RDP が値 0xAA に再プログラムされると、PCROP_RDP が Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）でセットされている場合、Flashメインメモリの全体消去が実行されます。バックアップレジスタ（RTC の RTC_BKPxR）および CCM SRAM も消去されます。PCROP 保護を除くユーザオプションは、Flash_OPTR、Flash_WRPxyR（x = 1、2 および y = A または B）からコピーされた以前の値にセットされます。PCROP は無効です。OTP エリアは全体消去に影響されませんので変化しません。

ビット PCROP_RDP が Flash_PCROP1ER でクリアされると、完全全体消去は PCROP によって保護されたページを除いて PCROP がアクティブになっているバンクで連続したページを消去する部分全体消去に代わります。これは、PCROP コードを維持するために実行されます。PCROP が両方のバンクでアクティブになっている場合、両方のバンクがページ消去で消去されます。両方のバンクが消去された場合のみ、オプションが以前の値で再プログラムされます。これは、Flash_PCROPxSR レジスタおよび Flash_PCROPxER レジスタ（x = 1、2）でも同様です。

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

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

図 4. 読出し保護（RDP）レベルの変更

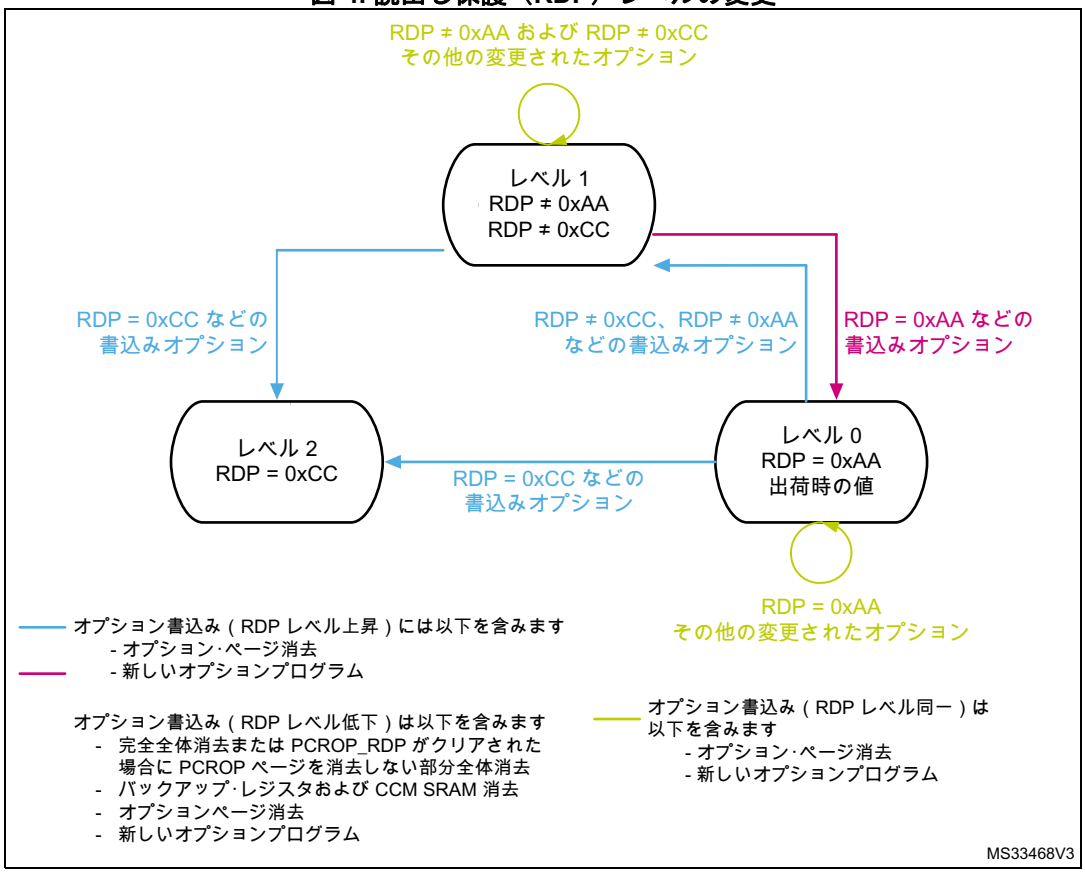


表 13. アクセス状態 対 保護レベルと実行モード

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
Flashメインメモリ	1	可能	可能	可能	不可	不可	不可 (3)
	2	可能	可能	可能	N/A	N/A	N/A
システムメモリ (2)	1	可能	不可	不可	可能	不可	不可
	2	可能	不可	不可	N/A	N/A	N/A
オプションバイト	1	可能	可能(3)	可能	可能	可能 (3)	可能
	2	可能	不可	不可	N/A	N/A	N/A
OTP	1	可能	可能(4)	N/A	不可	不可	N/A
	2	可能	可能 (4)	N/A	N/A	N/A	N/A
バックアップレジスタ	1	可能	可能	N/A	不可	不可	不可(5)
	2	可能	可能	N/A	N/A	N/A	N/A

表 13. アクセス状態 対 保護レベルと実行モード （続き）

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
CCM SRAM	1	可能	可能	N/A	不可	不可	不可(6)
	2	可能	可能	N/A	N/A	N/A	N/A

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

3.5.2 商用コード読出し保護（Proprietary code readout protection: PCROP）

Flash メモリの一部を、第三者の読出しおよび書込みから保護することができます。保護領域は実行専用です。STM32 CPU のみが命令コードとしてアクセスでき、その他すべてのアクセス（DMA、デバッグ、CPU のデータ読出し、書込み、消去）は厳しく禁じられています。DBANK モードに応じて、デュアルバンクモードでバンクごとに 1 つの PCROP ゾーンを指定するか、全メモリに対して 2 つの PCROP ゾーンを指定することができます。追加のオプションビット（PCROP_RDP）で、RDP 保護がレベル 1 からレベル 0 に変更された場合、PCROP 領域を消去するしないを選択できます。(読出し保護レベルの変更を参照)。

各 PCROP 領域は、物理的なFlashバンクベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは PCROP アドレスレジスタのFlash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）、Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）、Flash PCROP2 開始アドレスレジスタ（Flash_PCROP2SR）、Flash PCROP2 終了アドレスレジスタ（Flash_PCROP2ER）で定義されます。

シングルバンクモード（DBANK = 0）の場合

- PCROPx（x = 1、2）領域は、ベース･アドレス + [PCROPx_STRT x 16]（このアドレスを含む）から、ベース･アドレス + [(PCROPx_END + 1) x 16]（このアドレスを含まない）までのアドレス範囲で定義されます。PCROP 領域の最小サイズは 2 ダブルワード（256 ビット）です。

デュアルバンクモード（DBANK = 1）の場合

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

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

- Flashでのブートがバンク 1 で実行される場合、Flash_PCROP1SR および Flash_PCROP1ER レジスタは次のようにプログラムする必要があります。
  - PCROP1_STRT = 0xC5F0
  - PCROP1_END = 0xE000
- 2 つのバンクをスワップする場合、保護をバンク 2 に適用し、Flash_PCROP2SR および Flash_PCROP2ER レジスタは次のようにプログラムする必要があります。
  - PCROP2_STRT = 0xC5F0
  - PCROP2_END = 0xE000

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

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

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

以前の例で、ページでの消去のため、ページ 0x62～0x70 の全ページがページ消去時に保護されています。（0x0806 2000～0x080 70FFF の全アドレスは消去できません）

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

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

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

表 14. PCROP 保護(1)

PCROPx レジスタ値（x = 1、2）	PCROP 保護領域
PCROPx_offset_strt > PCROPx_offset_end	PCROP 領域なし
PCROPx_offset_strt < PCROPx_offset_end	PCROPx_offset_strt と PCROPx_offset_end の間の領域が保護されます。 以下の値を書込み可能です。 – PCROPx_offset_strt に下の値 – PCROPx_offset_end に上の値

1. DBANK = 1 の場合、PCROP 領域の最小サイズは 2 ダブルワード（PCROPx_offset_strt と PCROPx_offset_end）です。
DBANK = 0 の場合、PCROP 領域の最小サイズは 2 ダブルワードの 2 倍（PCROPx_offset_strt と PCROPx_offset_end）です。
DBANK = 1 の場合、PCROP ゾーンにオーバーラップが生じないようにするのはユーザの責任です。

3.5.3 書込み保護（WRP）

Flashメモリのユーザエリアを、不要な書込み操作から保護できます。DBANK のオプションビットの設定に応じて、次のいずれかを指定できます。

- シングルバンクモード（DBANK = 0）の場合：バンクごとに 4 つの書込み保護（WRP）領域をページサイズ（4 KB）で定義できます。
- デュアルバンクモード（DBANK = 1）の場合：バンクごとに 2 つの書込み保護（WRP）領域をページの最小単位（2 KB）で定義できます。

各領域は、物理的なフFlashバンクベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは次の WRP アドレスレジスタで定義されます。Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）、Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）、Flashバンク 2 WRP 領域 A アドレスレジスタ（Flash_WRP2AR）、Flashバンク 2 WRP 領域 B アドレスレジスタ（Flash_WRP2BR）。

デュアルバンクモード（DBANK = 1）

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

シングル･バンク･モード（DBANK = 0）

WRPx「y」領域（x = 1、2、y = A、B）は、次のアドレス範囲で定義されます。ベース･アドレス+ [WRPy_STRT x 0x1000]（このアドレスを含む）からベース･アドレス+ [(WRPy_END+1) x 0x1000]（このアドレスを含まない）まで。

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

- Flashでのブートがバンク 1 で実行される場合、Flash_WRP1AR レジスタは次のようにプログラムする必要があります。
  - WRP1A_STRT = 0x62
  - WRP1A_END = 0x70

  Flash_WRP1BR の WRP1B_STRT と WRP1B_END が代わりに使用できます（バンク 1 の領域「B」）。
- 2 つのバンクをスワップする場合、保護をバンク 2 に適用し、Flash_WRP2AR レジスタは次のようにプログラムする必要があります。
  - WRP2A_STRT = 0x62
  - WRP2A_END = 0x70
  - WRP2A_STRT = 0xC5
  - WRP2A_END = 0xE0

  Flash_WRP2BR の WRP2B_STRT と WRP2B_END が代わりに使用できます（バンク 2 の領域「B」）。

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

Flashメモリの書込み保護されているエリアに消去／プログラム操作の実施を試みると、Flash_SR レジスタで書込み保護エラーフラグ（WRPERR）がセットされます。このフラグは、次の領域への書込みアクセスのすべてでもセットされます。

- OTP エリア
- ICP など書込みできないFlashメモリの部分
- PCROP エリア

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

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

注 : DBANK = 0 の場合、WRP ゾーンにオーバーラップが生じないようにするのはユーザの責任です。

表 15. WRP 保護

WRP レジスタの値（x = 1/2、y = A/B）	WRP 保護領域
WRPxy_STRT = WRPxy_END	ページ WRPxy は保護されます。
WRPxy_STRT > WRPxy_END	WRP 領域なし。
WRPxy_STRT < WRPxy_END	WRPxy_STRT から WRPxy_END までのページは保護されます。

3.5.4 セキュリティ保護可能な領域

セキュリティ保護可能な領域によって、ブート時に 1 回のみ実行可能であり、新たなリセットが発生しない限りは絶対に実行できないコード領域が定義されます。

セキュリティ保護可能な領域の主な目的は、望ましくないアクセスからFlashメモリの特定の領域を保護することです。これは、ファームウェア･ブートの第 1 段階を、そこに含まれる機密性の高い資源（キー）とともに、アプリケーション･コードの他の部分から分離する手段になります。これによって、セキュア･ブートなどの信頼性ソリューションのルートを実装できます。セキュリティ保護可能なメモリ領域は、メイン Flash メモリに配置されています。信頼できるコードを実行するための専用領域です。セキュリティ保護可能な領域が設定されない場合には、残りのメインFlashメモリと同じように動作します。セキュリティ保護可能な領域が設定（Flash_CR レジスタの SEC_PROT1（または SEC_PROT2）ビットがセット）された場合には、セキュリティ保護可能な領域のプログラムまたは消去を試みると書込み保護エラーが発生し（WRPERR フラグがセットされます）、その領域からの読出しを試みると読出しエラーが発生します（RDERR フラグがセットされます）。

セキュリティ保護可能なメモリ領域のサイズは、FLASH_SEC1R（または FLASH_SEC2R）レジスタの SEC_SIZE1[7:0]（または SEC_SIZE2[7:0]）ビット･フィールドによって定義されます。変更は、RDP がレベル 0 の場合にのみ可能です。内容は、RDP をレベル 1 からレベル 0 に変更すると消去されます。PCROP ページとオーバーラップしている場合も同様です。

セキュリティ保護可能なメモリ領域は、次のアドレス範囲で定義されます。

デュアル･バンク構成（DBANK = 1）の場合：

バンク 1 のベース･アドレス（このアドレスを含む）からバンク 1 のベース･アドレス + (0x800 x SEC_SIZE1)（このアドレスを含まない）まで、およびバンク 2 のベース･アドレス（このアドレスを含む）からバンク 2 のベース･アドレス + [0x800 x SEC_SIZE2]（このアドレスを含まない）まで。

シングル･バンク構成（DBANK = 0）の場合：

バンクのベース･アドレス（このアドレスを含む）からバンクのベース･アドレス + [0x1000 x SEC_SIZE1]（このアドレスを含まない）まで。

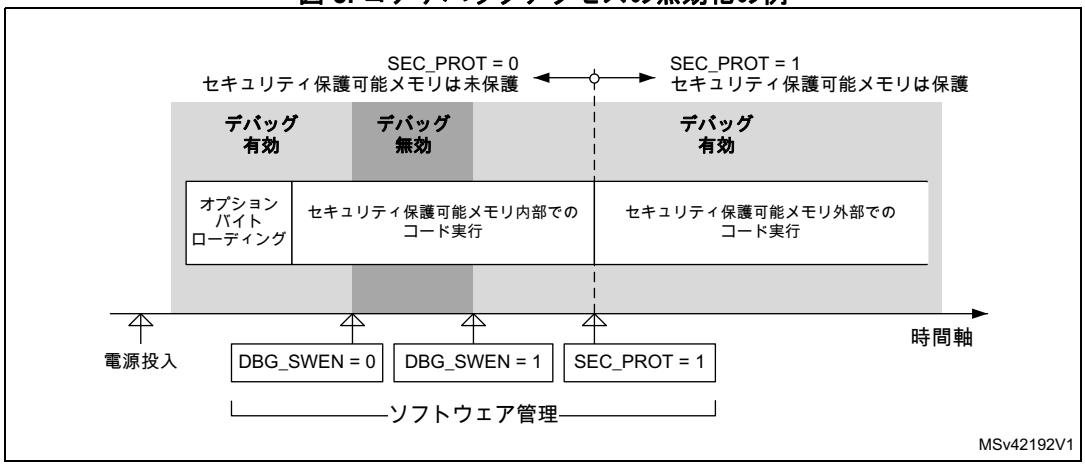
3.5.5 コア･デバッグ･アクセスの無効化

コアへのデバッグアクセスは、セキュリティ保護可能な領域で機密コードを実行したり機密データを処理するために、一時的に無効化できます。

RDP レベル 2 ではデバッガはハードウェアによって無効化されますが、それ以外の RDP レベルでは、デバッガは Flash_ACR レジスタの DBG_SWEN ビットを用いてソフトウェアによって無効化できます。

図 11 に、DBG_SWEN と SEC_PROT のビットの管理例を示します。

図 5. コアデバッグアクセスの無効化の例



3.5.6 Flashメモリからの強制ブート

セキュリティを強化し、信頼のチェーンを構築するため、Flash_SEC1R/Flash_SEC2R レジスタの BOOT_LOCK オプションビットでは、他のブートオプションにかかわらず、システムをメインFlash メモリから強制ブートすることができます。また、BOOT_LOCK ビットはいつでもセットすることができます。ただし、リセットは次の場合にのみ可能です。

- RDP をレベル 0 にセットした場合
- レベル 0 を要求し、完全全体消去を実行しているときに、RDP をレベル 1 にセットした場合

3.6 Flash 割込み

表 16. Flash 割込みリクエスト

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

1. EOP がセットされるのは EOPIE がセットされている場合のみです。
2. OPERR がセットされるのは ERRIE がセットされている場合のみです。

3.7 Flash レジスタ

3.7.1 Flash アクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x00

リセット値：0x0004 0601

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DBG_SWEN**：デバッグソフトウェア有効化

ソフトウェアは、このビットを使用してデバッガを有効／無効にすることができます。

0：デバッガは無効です。

1：デバッガは有効です。

ビット 17:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **SLEEP_PD**：SLEEP モードまたは低電力 SLEEP モード中のFlashパワーダウンモード

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

0：SLEEP モードまたは低電力 SLEEP モード中に Flash をアイドル･モードにします。

1：SLEEP モードまたは低電力 SLEEP モード中に Flash をパワーダウン･モードにします。

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

ビット 13 **RUN_PD**：RUN モードまたは低電力 RUN モード中のFlashパワーダウンモード

このビットは Flash_PDKEYR で書込み保護されています。

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

0：Flash をアイドル･モードにします。

1：Flash をパワーダウン･モードにします。

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

ビット 12 **DCRST**：データ･キャッシュのリセット

0：データ･キャッシュはリセットされません。

1：データ･キャッシュはリセットされます。

このビットに書き込めるのは、データキャッシュが無効のときだけです。

ビット 11 **ICRST**：命令キャッシュのリセット

0：命令キャッシュはリセットされません。

1：命令キャッシュはリセットされます。

このビットに書き込めるのは、命令キャッシュが無効のときだけです。

ビット 10 **DCEN**：データ･キャッシュ有効化
0：データ･キャッシュは無効です。
1：データ･キャッシュは有効です。

ビット 9 **ICEN**：命令キャッシュ有効化
0：命令キャッシュは無効です。
1：命令キャッシュは有効です。

ビット 8 **PRFTEN**：プリフェッチ有効化
0：プリフェッチは無効です。
1：プリフェッチは有効です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **LATENCY[3:0]**：遅延
これらのビットは、Flashアクセス時間に対する SYSCLK（システムクロック）周期の割合を表します。
0000：ウェイトステート 0 個
0001：ウェイトステート 1 個
0010：ウェイトステート 2 個
0011：ウェイトステート 3 個
0100：ウェイトステート 4 個
...1111：ウェイトステート 15 個

3.7.2 Flashパワーダウンキーレジスタ（Flash_PDKEYR）

アドレス･オフセット：0x04

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **PDKEYR[31:0]**：RUN モードにおけるパワーダウン Flash キー
Flash_ACR の RUN_PD ビットをアンロックするには、次の値を連続して書き込む必要があります。
PDKEY1：0x0415 2637
PDKEY2：0xFAFB FCFD

3.7.3 Flash キー レジスタ（FLASH_KEYR）

アドレス･オフセット：0x08

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **KEYR[31:0]**：Flash キー

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

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

3.7.4 Flashオプションキーレジスタ（Flash_OPTKEYR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **OPTKEYR[31:0]**：オプション･バイト･キー

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

KEY1：0x0819 2A3B

KEY2：0x4C5D 6E7F

3.7.5 Flashステータスレジスタ（Flash_SR）

アドレス･オフセット：0x10

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BSY**：ビジー

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

ビット 15 **OPTVERR**：オプション有効性エラー

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

1 を書き込むとクリアされます。

ビット 14 **RDERR**：PCROP 読出しエラー

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

1 を書き込むとクリアされます。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **FASTERR**：高速プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 8 **MISSERR**：高速プログラミング･データ･ミス･エラー

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

1 を書き込むとクリアされます。

ビット 7 **PGSERR**：プログラミング･シーケンス･エラー

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

DBANK = 0 でバンク消去を試みた場合にもセットされます。

1 を書き込むとクリアされます。

ビット 6 **SIZERR**：サイズ･エラー

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

1 を書き込むとクリアされます。

ビット 5 **PGAERR**：プログラミング配置エラー

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

1 を書き込むとクリアされます。

ビット 4 **WRPERR**：書込み保護エラー

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

1 を書き込むとクリアされます。

ビット 3 **PROGERR**：プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OPERR**：操作エラー

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

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

1 を書き込むとクリアされます。

ビット 0 **EOP**：操作終了

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

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

1 を書き込むとクリアされます。

3.7.6 Flash制御レジスタ（Flash_CR）

アドレス･オフセット：0x14

リセット値：0xC000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPT LOCK	SEC_ PROT2	SEC_ PROT1	OBL_ LAUNCH	RD ERRIE	ERR IE	EOP IE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPT STRT	START
rs	rs	rs	rs	rc_w1	rw	rw	rw						rw	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MER2	Res.	Res.	Res.	BKER	Res.	PNB[6:0]							MER1	PER	PG
rw				rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：FLASH_CR ロック

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

ビット 30 **OPTLOCK**：オプションロック

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

ビット 29 **SEC_PROT2**：バンク 2 のセキュリティ保護可能領域保護ビットです。

このビットは、バンク 2 のセキュリティ保護可能な領域へのアクセスをロックするために設定します。セキュリティ保護可能な領域を抜けるときにソフトウェアによってセットされ、1 回に限り書込み可能です。DBANK = 0 の場合は、このビットは無視されます。

ビット 28 **SEC_PROT1**：バンク 1 のセキュリティ保護可能領域保護ビットです。

このビットは、バンク 1 （または DBANK = 0 の場合）のセキュリティ保護可能な領域へのアクセスをロックするために設定します。セキュリティ保護可能な領域を抜けるときにソフトウェアによってセットされ、1 回に限り書込み可能です。

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

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

ビット 26 **RDERRIE**：PCROP 読出しエラー割込み有効化

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

ビット 25 **ERRIE**：エラー割込み有効化

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

ビット 24 **EOPIE**：操作終了割込みイネーブル

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

0：EOP 割込みは無効です。

1：EOP 割込みは有効です。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSTPG**：高速プログラミング

0：高速プログラミングは無効です。

1：高速プログラミングは有効です。

ビット 17 **OPTSTRT**：オプション変更開始

このビットがセットされると、オプション操作がトリガされます。

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

ビット 16 **START**：スタート

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

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

ビット 15 **MER2**：バンク 2 全体消去

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

ビット 14:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **BKER**：バンク消去

DBANK = 1

0：バンク 1 がページ消去対象に選択されます。

1：バンク 2 がページ消去対象に選択されます。

DBANK = 0

予約済みであり、リセット値に保持する必要があります。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:3 **PNB[6:0]**：ページ番号選択

これらのビットは、消去するページを選択します。

00000000：ページ 0

00000001：ページ 1

.....

11111111：ページ 255

ビット 2 **MER1**：バンク 1 全体消去

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

ビット 1 **PER**：ページ消去

0：ページ消去は無効です。

1：ページ消去は有効です。

ビット 0 **PG**：プログラミング

0：Flash プログラミングは無効です。

1：Flash プログラミングは有効です。

3.7.7 Flash ECC レジスタ（Flash_ECCR）

アドレス･オフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ECCD	ECCC	ECCD2	ECCC2	Res.	Res.	Res.	ECCCIE	Res.	SYSF_ECC	BK_ECC	Res.	Res.	ADDR_ECC[18:16]		
rc_w1	rc_w1	rc_w1	rc_w1				rw		r	r			r	r	r

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

ビット 31 **ECCD**：ECC 検出

DBANK = 1

2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が、あらかじめクリアされている場合のみ）。このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

DBANK = 0

64 ビット LSB（ビット 63:0）で 2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が、あらかじめクリアされている場合のみ）。このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

ビット 30 **ECCC**：ECC 訂正

DBANK = 1

1 つの ECC エラーが検出されて訂正されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が、あらかじめクリアされている場合のみ）。ECCCIE がセットされると、割込みが生成されます。

1 を書き込むとクリアされます。

DBANK = 0

64 ビット LSB（ビット 63 : 0）で 1 つの ECC エラーが検出されて訂正されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が、あらかじめクリアされている場合のみ）。

1 を書き込むとクリアされます。

ビット 29 **ECCD2**：ECC2 検出

DBANK = 0

64 ビット MSB（ビット 127 : 64）で 2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます。

このビットがセットされるのは ECCC/ECCC2/ECCD/ECCD2 が、あらかじめクリアされている場合のみです。

このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

DBANK = 1

予約済みであり、リセット値に保持する必要があります。

ビット 28 **ECCC2**：ECC 訂正

DBANK = 0

64 ビット MSB（ビット 127:64）で 1 つの ECC エラーが検出され訂正されたときに、ハードウェアによってセットされます。このビットがセットされます（ECCC/ECCC2/ECCD/ECCD2 が以前にクリアされている場合のみ）。ECCCIE がセットされると、割込みが生成されます。

1 を書き込むとクリアされます。

DBANK = 1

予約済みであり、リセット値に保持する必要があります。

ビット 27:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ECCCIE**：ECC 訂正割込み有効化

0：ECCC 割込み無効

1：ECCC 割込みは有効です。

DBANK = 0

このビットは、FLASH_ECCR レジスタの ECCC または ECCC2 ビットがセットされている場合に割込み生成を有効にします。

DBANK = 1

このビットは、FLASH_ECCR レジスタの ECCC ビットがセットされている場合に割込み生成を有効にします。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SYSF_ECC**：システムFlash ECC 失敗

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

ビット 21 **BK_ECC**：ECC 失敗バンク

DBANK = 1

このビットは、ECC エラー訂正またはダブル ECC エラー検出の対象とするバンクを示します。

0：バンク 1

1：バンク 2

DBANK = 0

SYSF_ECC が 1 の場合、ECC エラー検出の対象とするバンクを指定します。

SYSF_ECC が 0 の場合、予約済みでクリア状態を保持する必要があります。

ビット 20:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **ADDR_ECC[18:0]**：ECC 失敗アドレス

DBANK = 0

このビットは、Flash メモリの中で、ECC エラー訂正またはダブル ECC エラー検出の対象とするアドレスを指定します。

DBANK = 1

このビットは、バンクの中で、ECC エラー訂正またはダブル ECC エラー検出の対象とするアドレスを指定します。

3.7.8 Flash オプション･レジスタ（FLASH_OPTR）

アドレス･オフセット：0x20

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

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRHEN	NRST_MODE [1:0]		n BOOT0	nSW BOOT0	CCM SRAM_ RST	SRAM _PE	nBOOT1	DBANK	Res.	BFB2	WWDG_ SW	IWGD_ STDBY	IWDG_ StOP	IWDG_ SW
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_S HDW	nRST_S TDBY	nRST_S TOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **IRHEN**：内部リセット･ホルダー･イネーブル･ビット

0：内部リセットは、NRST ピン上の単純なパルスとして伝播します。

1：内部リセットは、ロー･レベルと見なされるまで NRST ピンをローになるように駆動します。

ビット 29:28 **NRST_MODE[1:0]**：

00：予約済みです。

01：リセット入力専用：NRST ピンが Low レベルになるとシステム･リセットを生成します。内部リセットが NRST ピンに伝播することはありません。

10：GPIO：標準 GPIO パッドの機能を使用します。内部リセットのみ可能です。

11：双方向リセット：NRST ピンが、リセット入出力モードで設定されます（レガシーモード）。

ビット 27 **nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24 **SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル

0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。

1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

ビット 23 **nBOOT1**：ブート設定

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

ビット 22 **DBANK**：

0：128 ビット･データ読出し幅でのシングル･バンク･モード

1：64 ビット･データでのデュアル･バンク･モード

このビットは、PCROPA/B が無効のときのみ書き込むことができます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **BFB2**：デュアルバンクブート

0：デュアル･バンク･ブートが無効です。
1：デュアル･バンク･ブートが有効です。

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択

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

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

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

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

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

ビット 16 **IWDG_SW**：独立型ウォッチドッグの選択

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

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **nRST_SHDW**

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

ビット 13 **nRST_STDBY**

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

ビット 12 **nRST_STOP**

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **BOR_LEV**：BOR リセットレベル

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

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

ビット 7:0 **RDP**：読出し保護レベル

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

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

3.7.9 Flash PCROP1 開始アドレスレジスタ（Flash_PCROP1SR）

アドレス･オフセット：0x24

リセット値：0xFFFF XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

アクセス：ノーウェイトステート（進行中のFlashメモリ操作がない場合）、ワードアクセス

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_STRT[14:0]**：PCROP 領域開始オフセット

DBANK = 1

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

DBANK = 0

PCROP1_STRT は、すべてのメモリの PCROP 領域の最初の 2 つのダブルワードを含んでいます。

3.7.10 Flash PCROP1 終了アドレスレジスタ（Flash_PCROP1ER）

アドレス･オフセット：0x28

リセット値：0xX000 XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

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

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

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

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

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

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_END[14:0]**：バンク 1 PCROP 領域終了オフセット

DBANK = 1

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

DBANK = 0

PCROP1_END は、すべてのメモリの最初の PCROP 領域の最後の 2 つのダブルワードを含んでいます。

3.7.11 Flashバンク 1 WRP 領域 A アドレスレジスタ（Flash_WRP1AR）

アドレス･オフセット：0x2C

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP1A_END[6:0]**：WRP 最初の領域「A」終了オフセット

DBANK = 1

WRP1A_END は、バンク 1 の WRP 最初の領域の最終ページを含んでいます。

DBANK = 0

WRP1A_END は、すべてのメモリの WRP 最初の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP1A_STRT[6:0]**：WRP 最初の領域「A」開始オフセット

DBANK = 1

WRP1A_STRT は、バンク 1 の WRP 最初の領域の最初のページを含んでいます。

DBANK = 0

WRP1A_STRT は、すべてのメモリの WRP 最初の領域の最初のページを含んでいます。

3.7.12 Flashバンク 1 WRP 領域 B アドレスレジスタ（Flash_WRP1BR）

アドレス･オフセット：0x30

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP1B_END[6:0]**：WRP 2 番目の領域「B」終了オフセット

DBANK = 1

WRP1B_END は、バンク 1 の WRP 2 番目の領域の最終ページを含んでいます。

DBANK = 0

WRP1B_END は、すべてのメモリの WPR 2 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP1B_STRT[6:0]**：WRP 2 番目の領域「B」開始オフセット

DBANK = 1

WRP1B_STRT は、バンク 1 の WRP 2 番目の領域の最終ページを含んでいます。

DBANK = 0

WRP1B_STRT は、すべてのメモリの WPR 2 番目の領域の最終ページを含んでいます。

3.7.13 Flash PCROP2 開始アドレスレジスタ（Flash_PCROP2SR）

アドレス･オフセット：0x44

リセット値：0xFFFF XXXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PCROP2_STRT [14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP2_STRT[14:0]**：PCROP 領域開始オフセット

DBANK = 1

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

DBANK = 0

PCROP2_STRT は、すべてのメモリの PCROP 領域の最初のダブルワードを含んでいます。

3.7.14 Flash PCROP2 終了アドレスレジスタ（Flash_PCROP2ER）

アドレス･オフセット：0x48

リセット値：0x0000 XXXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード･アクセス

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP2_END[14:0]**：PCROP 領域終了オフセット

DBANK = 1

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

DBANK = 0

PCROP2_END は、すべてのメモリの PCROP 領域の最後の 2 つのダブルワードを含んでいます。

3.7.15 Flashバンク 2 WRP 領域 A アドレスレジスタ（Flash_WRP2AR）

アドレス･オフセット：0x4C

リセット値：0x00XX 00XX

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END [6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT [6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP2A_END[6:0]**：WRP 最初の領域「A」終了オフセット

DBANK = 1

WRP2A_END は、バンク 2 の WRP 最初の領域の最終ページを含んでいます。

DBANK = 0

WRP2A_END は、すべてのメモリの WRP 3 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP2A_STRT[6:0]**：WRP 最初の領域「A」開始オフセット

DBANK = 1

WRP2A_STRT は、バンク 2 の WRP 最初の領域の最初のページを含んでいます。

DBANK = 0

WRP2A_STRT は、すべてのメモリの WRP 3 番目の領域の最初のページを含んでいます。

3.7.16 Flashバンク 2 WRP 領域 B アドレスレジスタ（Flash_WRP2BR）

アドレス･オフセット：0x50

リセット値：0x00XX 00XX

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

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **WRP2B_END[6:0]**：WRP 2 番目の領域「B」終了オフセット
DBANK = 1
WRP2B_END は、バンク 2 の WRP 2 番目の領域の最終ページを含んでいます。
DBANK = 0
WRP2B_END は、すべてのメモリの WRP 4 番目の領域の最終ページを含んでいます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **WRP2B_STRT[6:0]**：WRP 2 番目の領域「B」開始オフセット
DBANK = 1
WRP2B_STRT は、バンク 2 の WRP 2 番目の領域の最初のページを含んでいます。
DBANK = 0
WRP2B_STRT は、すべてのメモリの WRP 2 番目の領域の最初のページを含んでいます。

3.7.17 Flash セキュリティ保護可能領域バンク 1 レジスタ（FLASH_SEC1R）

アドレス･オフセット：0x70

リセット値：0xFFFX FFXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ Flash メモリ領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SEC_SIZE1[7:0]**：バンク 1 のセキュリティ保護可能な領域に使用されているページ数がセットされます。

セキュリティ保護可能な領域は 0x0800 0000 から始まり、そのサイズは SEC_SIZE1 * ページ･サイズとなります。このフィールドはレベル 0 でのみ変更可能です。

レベル 1 で変更を試みるとサイレントに失敗し、レジスタ値は変更されません。

3.7.18 Flash セキュリティ保護可能領域バンク 2 レジスタ（FLASH_SEC2R）

アドレス･オフセット：0x74

リセット値：0xFFFF FFXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **SEC_SIZE2[7:0]**：バンク 2 のセキュリティ保護可能な領域に使用されているページ数がセットされます。

セキュリティ保護可能な領域は 0x0804 0000 から始まり、そのサイズは SEC_SIZE2 * ページ･サイズとなります。

DBANK = 0 の場合は、このビットは無視されます。

このフィールドはレベル 0 でのみ変更可能です。レベル 1 で変更を試みると黙然と失敗し、レジスタ値は変更されます。

3.7.19 Flash レジスタマップ

表 17. Flash インタフェース - レジスタ・マップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_SWEN	Res.	Res.	Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY [3:0]			
	リセット値														1				0	0	0	0	1	1	0					0	0	0	1
0x04	FLASH_ PDKEYR	PDKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FLASH_KEYR	KEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FLASH_OPT KEYR	OPTKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FLASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY	OPTVERR	RDERR	Res.	Res.	Res.	Res.	FASTERR	MISSERR	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値																0	0	0					0	0	0	0	0	0	0		0	0
0x14	FLASH_CR	LOCK	OPTLOCK	SEC_PROT2	SEC_PROT1	OBL_LAUNCH	RDERRIE	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPTSTRT	STRT	MER2	Res.	Res.	Res.	BKER	Res.	PNB[6:0]							MER1	PER	PG
	リセット値	1	1	1	1	0	0	0	0						0	0	0	0				0		0	0	0	0	0	0	0	0	0	0
0x18	FLASH_ECCR	ECCD	ECCC	ECCD2	ECCC2	Res.	Res.	Res.	ECCCIE	Res.	SYSF_ECC	BK_ECC	Res.	Res.	ADDR_ECC[18:0]																		
	リセット値	0	0	0	0				0		0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	FLASH_OPTR	Res.	Res.	Res.	Res.	nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	DBANK	Res.	BFB2	WWDG_SW	IWDG_STBY	IWDG_STOP	IWDG_SW	Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_ LEV[2:0]			RDP[7:0]							
	リセット値					X	X	x	x	x	X		x	x	x	x	x		x	x	x		x	x	x	x	x	x	x	x	x	x	x
0x24	FLASH_ PCROP1SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_STRT [14:0]														
	リセット値																		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x28	FLASH_ PCROP1ER	PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_END [14:0]														
	リセット値	x																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x2C	FLASH_ WRP1AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [6:0]						
	リセット値										x	x	x	x	x	x	x										x	x	x	x	x	x	x
0x30	FLASH_ WRP1BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT [6:0]						
	リセット値										x	x	x	x	x	x	x										x	x	x	x	x	x	x

表 17. Flash インタフェース - レジスタ･マップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	FLASH_ PCROP2SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_STRT [14:0]														
	リセット値																		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x48	FLASH_ PCROP2ER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP2_END [14:0]														
	リセット値																		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x4C	FLASH_ WRP2AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_END [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2A_STRT [6:0]						
	リセット値										X	X	X	X	X	X	X										X	X	X	X	X	X	X
0x50	FLASH_ WRP2BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_END [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP2B_STRT [7:0]						
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x70	FLASH_ SEC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [7:0]						
	リセット値																X									X	X	X	X	X	X	X	X
0x74	FLASH_ SEC2R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE2 [7:0]						
	リセット値																									X	X	X	X	X	X	X	X

レジスタ境界アドレスについては、80ページのセクション 2.2を参照してください。

4 内蔵 Flash メモリ（Flash）カテゴリ 4 デバイスの場合

4.1 概要

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

また、命令プリフェッチおよびキャッシュラインでコードの実行を加速します。

4.2 Flash の主な機能

- 最大 512 KB の Flash メモリ（シングル･バンク）
- 64 ビットデータ幅に対応した Flash メモリ読出し操作
- ページ消去と全体消去

Flashメモリインタフェースの機能：

- Flashメモリ読出し操作
- Flashメモリプログラム／消去操作
- オプション（RDP）で有効化される読出し保護
- オプションで選択される 2 つの書込み保護領域
- オプションで定義される独自仕様コード読出し保護領域
- オプションで定義されるセキュリティ保護可能なメモリ領域
- ICode でプリフェッチ
- 命令キャッシュ：ICode で 4 x 64 ビットのキャッシュライン 32 本（1 KB RAM）
- データキャッシュ：DCode で 4 x 64 ビットのキャッシュライン 8 本（256B RAM）
- エラーコード訂正 ECC：64 ビットダブルワードごとに 8 ビット
  - 8 + 64 = 72 ビット、検出 2 ビット、訂正 1 ビット
- オプションバイトローダ
- 低電力モード

4.3 Flash の機能説明

4.3.1 Flash メモリの構成

Flashメモリは、以下のような主な特徴を備えています。

- 最大容量 512 KB（読出し幅 64 ビット）
- メイン･メモリの 1 つのシングル･バンクに 512 KB を構成
- ページサイズ 2 KB
- 72 ビット幅のデータ読出し（64 ビット + ECC 8 ビット）
- ページ消去と全体消去
- 各ページには 256 KB の行が 8 行含まれます。

- 情報ブロックには次のものが含まれます。
  - システムメモリブートモードでデバイスがブートするシステムメモリ。この領域は STMicroelectronics によって予約済みで、次のいずれかのインタフェースを介したFlashメモリの再プログラムに使用するブートローダを含みます。USART、SPI、I2C、USB。これはデバイスの製造時に STMicroelectronics によってプログラムされており、誤った書込み／消去操作から保護されています。詳細については、www.st.com から入手可能な AN2606 を参照してください。
  - OTP（1 度だけプログラムが可能な）、ユーザデータ用の 1 KB（128 ダブルワード）。OTP データは消去不可で、1 度だけ書込み可能です。1 つのビットだけが 0 の場合、0x0000 0000 0000 0000 の値であってもダブルワード全体を書き込むことはできません。
  - ユーザ設定のオプションバイト。

メモリ構成は、表 28 に示されるメイン領域と情報ブロックに基づきます。

表 18. Flash モジュール - 256/512 KB の構成（読出し幅 64 ビット）

Flash 領域	Flashメモリアドレス	サイズ（バイト）	名前
メインメモリ 256/512 KB(1)	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
	0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
	0x0800 1000 - 0x0800 17FF	2 K	ページ 2
	0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
	-	-	-
	-	-	-
	-	-	-
	-	-	-
	0x0807 F800 - 0x0807 FFFF	2 K	ページ 255
情報ブロック	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	0x1FFF 7800 - 0x1FFF 782F	48	オプションバイト

1. 256 KB デバイスの場合：ページ 0 ~ ページ 127

4.3.2 エラーコード訂正（ECC）

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

- 1 つのエラー検出および訂正
- 2 つのエラー検出

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

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

ECC エラーが検出されると、失敗したダブルワードのアドレスが、Flash_ECCR レジスタの ADDR_ECC[20:0] に保存されます。ADDR_ECC[2:0] は常にクリアされます。

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

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

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

4.3.3 読出しアクセスの遅延

データをFlashメモリから正しく読み出すには、CPU クロック（HCLK）の周波数およびデバイスの内部電圧範囲 V_{CORE} に従って、Flash アクセス制御レジスタ（FLASH_ACR）でウェイトステート（LATENCY）の数を正しくプログラムする必要があります。セクション 6.1.5：ダイナミック電圧スケーリングの管理 を参照してください。表 19 には、ウェイトステートと CPU クロック周波数の対応を示します。

表 19. CPU クロック（HCLK）周波数によるウェイトステート数

ウェイトステート（WS）（LATENCY）	HCLK (MHz)		
	V_{CORE} レンジ 1 ブースト·モード	V_{CORE} レンジ 1 通常モード	V_{CORE} レンジ 2
0 WS（1 CPU サイクル）	≤ 34	≤ 30	≤ 12
1 WS（2 CPU サイクル）	≤ 68	≤ 60	≤ 24
2 WS（3 CPU サイクル）	≤ 102	≤ 90	≤ 26
3 WS（4 CPU サイクル）	≤ 136	≤ 120	-
4 WS（5 CPU サイクル）	≤ 170	≤ 150	-

リセット後、CPU クロック周波数は 16 MHz であり、Flash_ACR レジスタでは 1 ウェイトステート（WS）が設定されます。

CPU 周波数を変更するときは、Flash メモリにアクセスするために必要なウェイトステート数を適切な値にするために、以下のソフトウェア·シーケンスを適用する必要があります。

CPU 周波数の増加：

1. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
2. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が反映されていることを確認します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロックプリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が減少する場合には、最初にそのステップを実行してからもう一方を実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。

CPU 周波数の減少：

1. RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更します。
2. 必要であれば、RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロックプリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が増加する場合には、最初にもう一方のステップを実行してからそのステップを実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。
5. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
6. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が使用されていることを確認します。

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

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

プロセッサのフルパフォーマンスを引き出すため、アクセラレータは命令プリフェッチキューと、64 ビットのFlashメモリからのプログラム実行速度を上昇させるブランチキャッシュを実行します。CoreMark ベンチマークによると、ART アクセラレータによって達成されるパフォーマンスは、最大 CPU 周波数 170 MHz におけるFlashメモリからの 0 ウェイトステートプログラム実行と同等となります。

命令プリフェッチ

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

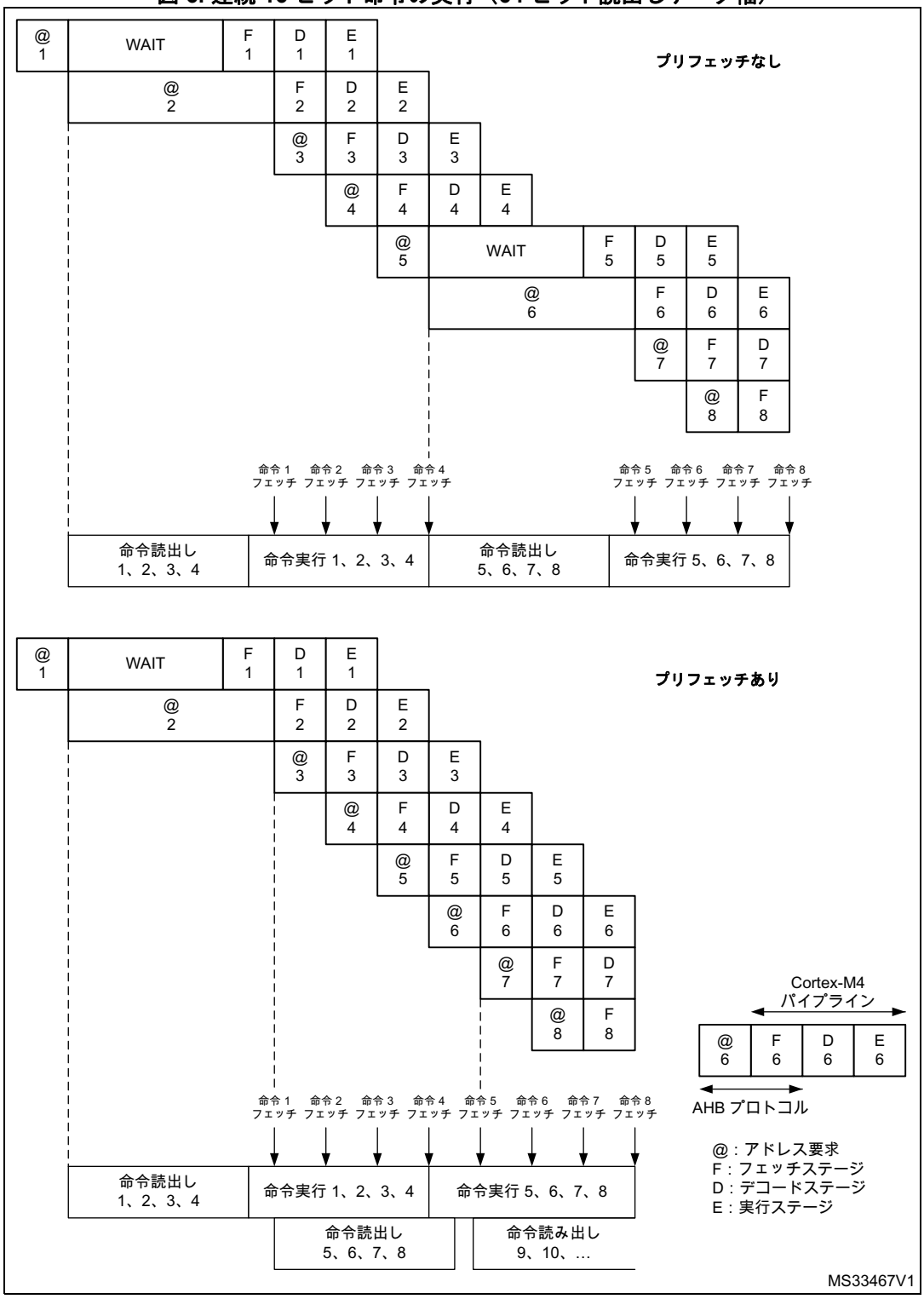
各Flashメモリ読出し操作では、起動されるプログラムに応じて、32 ビットの命令 2 個または 16 ビットの命令 4 個の 64 ビットが提供されます。この 64 ビットの現在の命令ラインは、現在のバッファに保存されますので、連続コードの場合は、その前の読出し命令ラインの実行に 2 個以上の CPU サイクルが必要となります。

ICode バスでのプリフェッチを使用すると、CPU によって現在の命令ラインが要求されている間にFlashメモリから次の連続命令ラインを読み出すことができます。

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

図 9 は、Flashメモリへのアクセスに 3 WS が必要なときに、プリフェッチがある場合とない場合の連続 16 ビット命令の実行を示しています。

図 6. 連続 16 ビット命令の実行（64 ビット読出しデータ幅）



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

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

命令キャッシュメモリ（I キャッシュ）

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

命令キャッシュメモリはシステムリセット後に有効になります。

データキャッシュメモリ（D キャッシュ）

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

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

データキャッシュメモリはシステムリセット後に有効になります。

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

オプションバイトブロックのデータは、キャッシュできません。

4.3.5 Flashのプログラムおよび消去操作

STM32G4 シリーズ 内蔵Flashメモリは、インサーキットプログラミングまたはアプリケーション内プログラミングを使用してプログラム可能です。

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

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

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

Flashメモリの消去およびプログラムは電圧スケーリングレンジ 1 でのみ可能です。PWR_CR1 の VOS[1:0] ビットは 01b に設定してください。

Flashメモリへのプログラム／消去操作中にFlashメモリを読み出そうとすると、バスがストールされます。読出し操作は、プログラム／消去操作が完了すると正しく処理されます。

Flash メモリのロック解除

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

1. Flash キー レジスタ（FLASH_KEYR）に KEY1 = 0x45670123 を書き込みます。
2. Flash_KEYR レジスタに KEY2 = 0xCDEF89AB を書き込みます。

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

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

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

4.3.6 Flashメインメモリの消去シーケンス

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

ページ消去

ページを消去するには、次の手順に従います。

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

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

消去するページが、（WRP または PCROP によって）書込み保護された領域に属している場合、WRPERR がセットされてページ消去リクエストが中止されます。

全体消去

全体消去を実行するには、次の手順に従います。

1. Flash_SR レジスタの BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー･プログラミング･フラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. Flash 制御レジスタ（FLASH_CR）の MER1 ビットをセットします。
4. FLACH_CR レジスタの STRT ビットをセットします。
5. Flashステータスレジスタ（Flash_SR）の BSY ビットがクリアされるまで待ちます。

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

Flash メモリに（WRP または PCROP によって）書込み保護された領域が含まれる場合、WRPERR がセットされて全体消去リクエストが中止されます。

4.3.7 Flashメインメモリのプログラミングシーケンス

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

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

ダブルワード（2 x 32 ビットデータ）のプログラムのみ可能です。

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

標準プログラミング

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

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

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

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

ECC は、プログラムするダブルワードから計算されます。

高速プログラミング

このモードでは、行（32 ダブルワード）をプログラムし、プログラム前に Flash の位置を確認する必要性を取り除くことでページ･プログラミングの時間を削減して、ダブルワードごとに高電圧の立ち上がり時間と立ち下がり時間が発生することを回避できます。高速プログラミング中は、CPU クロック周波数（HCLK）は少なくとも 8 MHz でなければなりません。

メインメモリのみ高速プログラミングモードでプログラム可能です。

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

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

注： **読出し操作が進行している際に、Flash が高速プログラミング･モードでの書込みを試みた場合、システム通知なしでプログラミングが中止されます（エラー･フラグはセットされません）。**

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

32 ダブルワードは連続で書き込む必要があります。高電圧は、すべてのプログラミングに対してFlashで維持されます。連続した 2 つのダブルワードの間の書込みリクエストの最大時間は、最初のダブルワードに対して、Flashがプログラムされている時間（2 x 25 us 前後）です。2 番目のダブルワードがこのプログラムされている時間を越える場合、高速プログラミングは割り込まれ、MISSERR がセットされます。

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

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

プログラミングエラー

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

- **PROGERR**：プログラミングエラー

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

- **SIZERR**：サイズ･プログラミング･エラー

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

- **PGAERR**：整列プログラミング･エラー

  次のいずれかの条件が発生した場合、PGAERR がセットされます。

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

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

- **PGSERR**：プログラミング･シーケンス･エラー

  次のいずれかの条件が発生した場合、PGSERR がセットされます。

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

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

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

  – 全体消去シーケンス：MER1 がセットされたときに PG、FSTPG、および PER がクリアされない場合

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

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

- **WRPERR**：書込み保護エラー

  次のいずれかの条件が発生した場合、WRPERR がセットされます。

  – 書込み保護領域（WRP）、PCROP 領域、またはセキュリティ保護可能メモリ領域でプログラミングまたは消去を試みた場合

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

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

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

- **MISSERR**：高速プログラミングデータミスエラー

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

- **FASTERR**：高速プログラミングエラー

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

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

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

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

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

WRPERR（保護エラーフラグ）

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

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

プログラミングとキャッシュ

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

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

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

4.4 Flashオプションバイト

4.4.1 オプションバイトの説明

オプションバイトは、アプリケーション要件によってエンドユーザが設定します。設定の例として、ウォッチドッグをハードウェアまたはソフトウェアモードに選択できます（セクション 5.4.2：オプションバイトのプログラミングを参照してください）。

ダブルワードは、オプションバイトで次に示すように分割されます。

表 20. オプションバイトのフォーマット

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

情報ブロック内のこれらのバイトの構成は、表 31：オプションバイトの構成に示すとおりです。

オプションバイトは、表 31：オプションバイトの構成にリストされたメモリ位置またはオプションバイトレジスタから読み出すことができます。

- Flashオプションレジスタ（Flash_OPTR）
- Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）
- Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）
- Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）
- Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）

表 21. オプションバイトの構成

アドレス	[63:56]	[55:48]	[47:40]	[39:32]	[31:24]	[23:16]	[15:8]	[7:0]
1FFF7800	USER OPT			RDP	USER OPT			RDP
1FFF7808	未使用		使用されないかつ PCROP1_STRT[15:0]		使用されない		使用されないかつ PCROP1_STRT[15:0]	
1FFF7810	PCROP_RDP および未使用		使用されないかつ PCROP1_END[15:0]		PCROP_RDP かつ使用されない		使用されないかつ PCROP1_END[15:0]	
1FFF7818	未使用	WRP1A_END [7:0]	使用されない	WRP1A_STRT [7:0]	使用されない	WRP1A_END [7:0]	未使用	WRP1A_STRT [7:0]
1FFF7820	未使用	WRP2A_END [7:0]	使用されない	WRP2A_STRT [7:0]	使用されない	WRP2A_END [7:0]	未使用	WRP2A_STRT [7:0]
1FFF7828	未使用	BOOT_LOCK	未使用	SEC_SIZE1	未使用	BOOT_LOCK	未使用	SEC_SIZE1

ユーザ･オプション･バイトおよび読出し保護オプション･バイト

Flash メモリ･アドレス：0x1FFF 7800

ST 出荷時の値：0xFFEF F8AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRH_EN	NRST_MODE[1:0]		nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	PB4_PUPEN	Res.	Res.	WWDG_SW	IWGD_STDBY	IWDG_STOP	IWDG_SW
	r	r	r	r	r	r	r	r	r			r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	r	r	r		r	r	r	r	r	r	r	r	r	r	r

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **IRH_IN**：PG10の内部リセットホルダ

0：IRH は無効です。

1：IRH は有効です。

ビット 29:28 **NRST_MODE[1:0]**：PG10 パッド･モード

00：入出力をリセット

01：入力のみをリセット

10：GPIO

11：入出力をリセット

ビット 27 **nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24 **SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル

0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。

1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

ビット 23 **nBOOT1**：ブート設定

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

ビット 22 **PB4_PUPEN**：PB4 プルアップ有効化

0：USB Power Delivery のデッド･バッテリ機能は有効、NJTRST プルアップは無効です。

1：USB Power Delivery のデッド･バッテリ機能は無効、NJTRST プルアップは有効です。

注： カテゴリ 4 デバイスの場合のみ（他のカテゴリでは予約済み）

ビット 21:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択

0：ハードウェアによるウィンドウ型ウォッチドッグです。

1：ソフトウェアによるウィンドウ型ウォッチドッグです。

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

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

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

ビット 15 予約済みであり、リセット値に保持する必要があります。

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

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

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

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

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

PCROP1 開始アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7808

リセット値：0xFFFF FFFF（ST 出荷時の値）

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PCROP1_STRT[15:0]**：PCROP 領域開始オフセット
PCROP1_STRT は、バンク 1 の PCROP 領域の最初のダブルワードを含んでいます。

PCROP1 終了アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7810

リセット値：0x00FF 0000（ST 出荷時の値）

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

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

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域
このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。
0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。
1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PCROP1_END[15:0]**：バンク 1 PCROP 領域終了オフセット
PCROP1_END は、PCROP 領域の最後のダブルワードを含んでいます。

WRP1 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7818

リセット値：0xFF00 FFFF（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [7:0]							
								r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [7:0]							
								r	r	r	r	r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **WRP1A_END[7:0]**：WRP 最初の領域「A」終了オフセット
WRP1A_END は、WRP の最初の領域の最終ページを含んでいます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

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

WRP2 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7820

リセット値：0xFF00 FFFF（ST 出荷時の値）

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **WRP2A_END[7:0]**：WRP 2 番目の領域「B」終了オフセット
WRP2A_END は、WRP の 2 番目の領域の最終ページを含んでいます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRP2A_STRT[7:0]**：WRP 2 番目の領域開始オフセット
WRP2A_STRT は、WRP の 2 番目の領域の最終ページを含んでいます。

セキュリティ保護可能メモリ領域オプションバイト

Flashメモリアドレス：0x1FFF7828

リセット値：0xFF00FE00（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **SEC_SIZE1[8:0]**：セキュリティ保護可能な領域のサイズ
セキュリティ保護可能な Flash メモリのページ数を含んでいます。

4.4.2 オプション･バイトのプログラミング

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

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

ソフトウェアによって OPTLOCK ビットをセットすると、ユーザ･オプションを意図せぬ消去やプログラム操作から保護できます。

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

ユーザオプションの変更

オプション･バイトは、メイン･メモリのユーザ･アドレスとは異なる方法でプログラムされます。ユーザ･オプション値を変更するには、次の手順に従います。

1. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中の Flash メモリ操作がないことを確認します。
2. 上記のクリアシーケンスで OPTLOCK オプションロックビットをクリアします。
3. オプション値を次のオプションレジスタに書き込みます。Flashオプションレジスタ（Flash_OPTR）、Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）、Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）、Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）、Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）。
4. Flash制御レジスタ（Flash_CR）でオプションスタートビット OPTSTRT をセットします。
5. BSY ビットがクリアされるのを待ちます。

注： **まず両方のユーザオプションバイトページを消去し、すべてのオプションバイトをFlashオプションレジスタに含まれている値でプログラムすると、一方のオプションの値の変更が自動的に実行されます。**

オプションバイトローディング

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

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

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

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

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

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

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

システムリセットの立ち上がりでは、内部オプションレジスタがソフトウェアで読出しおよび書込み可能なオプションレジスタ（Flash_OPTR、Flash_PCROP1SR、Flash_PCROP1ER、Flash_WRP1AR、Flash_WRP1BR）にコピーされます。これらのレジスタはオプション変更にも使用されます。これらのレジスタがユーザによって変更されない場合、システムのオプション状態が反映されます。詳細は、セクション ：ユーザオプションの変更を参照してください。

4.5 Flash メモリの保護

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

4.5.1 読出し保護（RDP）

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

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

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

Flashメモリは、RDP オプションバイトとその補数に 表 32 に示す値のペアが含まれる場合に保護されます。

表 22. Flashメモリの読出し保護ステータス

RDP バイト値	RDP 補数	読出し保護レベル
0xAA	0x55	レベル 0（ST 出荷時の設定）
0xAA または 0xCC を除く任意の値	0x55 および 0x33 以外の任意の値（相補である必要はない）	レベル 1
0xCC	0x33	レベル 2

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

レベル 0：保護なし

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

レベル 1：読出し保護

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

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

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

レベル 2：デバッグなし

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

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

注： **リセット後、デバッグ機能は無効化されます。**

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

読出し保護レベルの変更

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

レベル 1 からレベル 0 に移動するために RDP が値 0xAA に再プログラムされると、PCROP_RDP が Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）でセットされている場合、Flashメインメモリの全体消去が実行されます。バックアップレジスタ（RTC の RTC_BKPxR）および CCM SRAM も消去されます。PCROP 保護を除くユーザオプションは、Flash_OPTR、Flash_WRPxyR（x = 1 および y = A または B）からコピーされた以前の値にセットされます。PCROP は無効です。OTP エリアは全体消去に影響されませんので変化しません。

ビット PCROP_RDP が Flash_PCROP1ER でクリアされると、完全全体消去は、PCROP によって保護されたページを除いて連続したページを消去する部分全体消去に代わります。これは、PCROP コードを維持するために実行されます。Flash メモリが消去された場合のみ、オプションが以前の値で再プログラムされます。これは、Flash_PCROPxSR レジスタおよび Flash_PCROPxER レジスタ（x = 1）でも同様です。

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

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

図 7. 読出し保護（RDP）レベルの変更

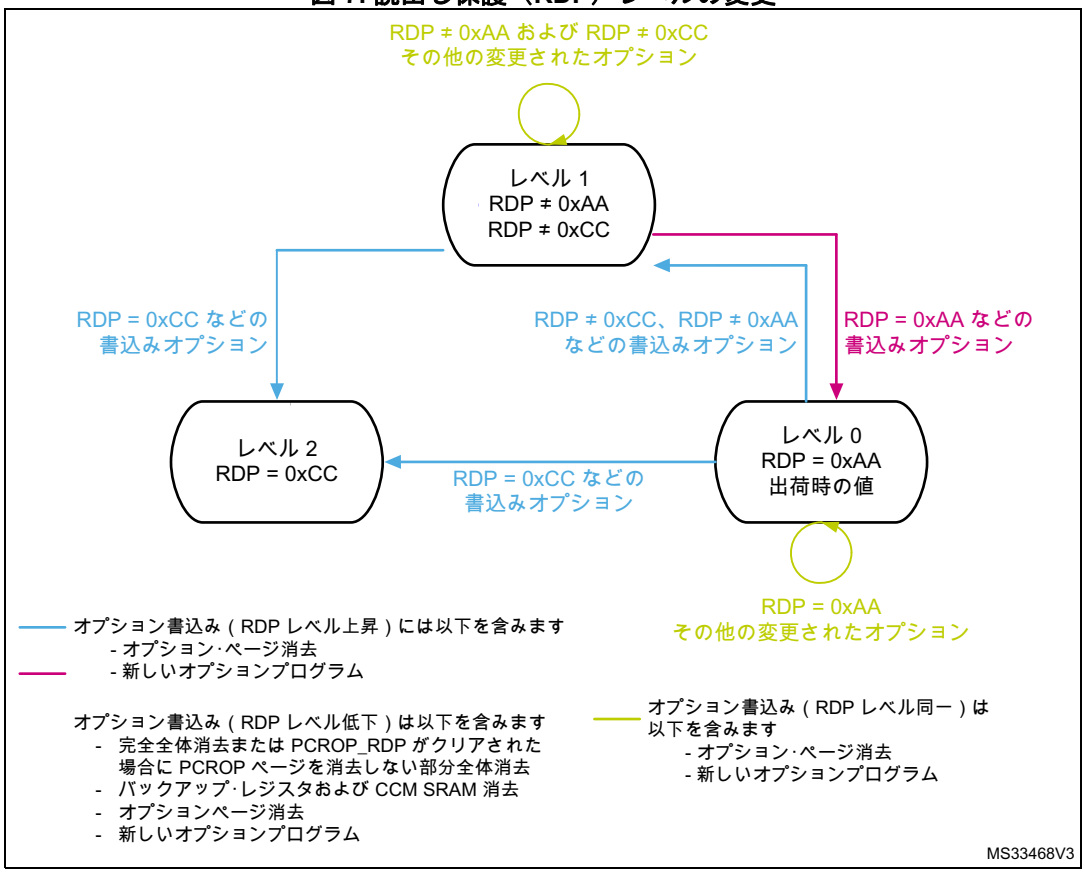


表 23. アクセス状態 対 保護レベルと実行モード

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
Flashメインメモリ	1	可能	可能	可能	不可	不可	不可 (3)
	2	可能	可能	可能	N/A	N/A	N/A
システムメモリ (2)	1	可能	不可	不可	可能	不可	不可
	2	可能	不可	不可	N/A	N/A	N/A
オプションバイト	1	可能	可能(3)	可能	可能	可能 (3)	可能
	2	可能	不可	不可	N/A	N/A	N/A
OTP	1	可能	可能(4)	N/A	不可	不可	N/A
	2	可能	可能 (4)	N/A	N/A	N/A	N/A
バックアップレジスタ	1	可能	可能	N/A	不可	不可	不可(5)
	2	可能	可能	N/A	N/A	N/A	N/A

表 23. アクセス状態 対 保護レベルと実行モード （続き）

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
CCM SRAM	1	可能	可能	N/A	不可	不可	不可(6)
	2	可能	可能	N/A	N/A	N/A	N/A

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

4.5.2 独自仕様コード読出し保護（Proprietary code readout protection: PCROP）

Flash メモリの一部を、第三者の読出しおよび書込みから保護することができます。保護領域は実行専用です。STM32 CPU のみが命令コードとしてアクセスでき、その他すべてのアクセス（DMA、デバッグ、CPU のデータ読出し、書込み、消去）は厳しく禁じられています。PCROP 領域の単位はダブルワード（64 ビット）です。追加のオプションビット（PCROP_RDP）で、RDP 保護がレベル 1 からレベル 0 に変更された場合、PCROP 領域を消去するしないを選択できます。（読出し保護レベルの変更を参照）。

各 PCROP 領域は、物理的なFlashアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは PCROP アドレスレジスタのFlash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）、Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）で定義されます。

- PCROPx （x = 1）領域は、次のアドレス範囲で定義されます。Flash メモリのベース･アドレス + [PCROPx_STRT x 0x8]（このアドレスを含む）からFlash メモリのベース･アドレス + [(PCROPx_END + 1) x 0x8]（このアドレスを含まない）PCROP 領域の最小サイズは 2 ダブルワード（128 ビット）です。

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

- Flashでのブートが選択された場合、Flash_PCROP1SR および Flash_PCROP1ER レジスタは次のようにプログラムする必要があります。
  - PCROP1_STRT = 0xC5F0
  - PCROP1_END = 0xE000

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

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

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

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

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

オプションビット PCROP_RDP をクリアする際に、RDP をレベル 1 からレベル 0 に変更している場合、PCROP 領域を維持するために完全全体消去が部分全体消去に代わります（読出し保護レベルの変更を参照してください）。この場合、PCROP1_STRT および PCROP1_END も消去されません。

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

表 24. PCROP 保護(1)

PCROPx レジスタ値 （x = 1）	PCROP 保護領域
PCROPx_offset_strt > PCROPx_offset_end	PCROP 領域なし
PCROPx_offset_strt < PCROPx_offset_end	PCROPx_offset_strt と PCROPx_offset_end の間の領域が保護されます。 以下の値を書込み可能です。 – PCROPx_offset_strt に下の値 – PCROPx_offset_end に上の値

1. PCROP 領域の最小サイズは 2 ダブルワード（PCROPx_offset_strt と PCROPx_offset_end）です。

4.5.3 書込み保護（WRP）

Flashメモリのユーザエリアを、不要な書込み操作から保護できます。以下のいずれかを指定可能です。

- 2 つの書込み保護（WRP）領域を、ページ（2 KB）単位で定義できます。

各領域は、物理的Flashのベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは次の WRP アドレスレジスタで定義されます。Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）、Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）。

WRP「y」領域（x = 1、y = A、B）は、次のアドレス範囲で定義されます。Flash メモリのベース･アドレス+ [WRPxy_STRT x 0x800]（このアドレスを含む）からFlash メモリのベース･アドレス + [(WRPxy_END+1) x 0x800]（このアドレスを含まない）

たとえば、アドレス 0x0801 2800（このアドレスを含む）からアドレス 0x0801 87FF（このアドレスを含む）までを WRP で保護するには、次のようにします。

- Flash のブートが選択されている場合、Flash_WRP1AR レジスタは次のようにプログラムする必要があります。
  - WRP1A_STRT = 0x25
  - WRP1A_END = 0x30

  Flash_WRP1BR の WRP1B_STRT と WRP1B_END が代わりに使用できます（Flash メモリの領域 B）。

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

Flashメモリの書込み保護されているエリアに消去／プログラム操作の実施を試みると、Flash_SR レジスタで書込み保護エラーフラグ（WRPERR）がセットされます。このフラグは、次の領域への書込みアクセスのすべてでもセットされます。

- OTP エリア
- ICP など書込みできないFlashメモリの部分
- PCROP エリア

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

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

表 25. WRP 保護

WRP レジスタの値（x = 1、y = A/B）	WRP 保護領域
WRPxy_STRT = WRPxy_END	ページ WRPxy は保護されます。
WRPxy_STRT > WRPxy_END	WRP 領域なし。
WRPxy_STRT < WRPxy_END	WRPxy_STRT から WRPxy_END までのページは保護されます。

4.5.4 セキュリティ保護可能な領域

セキュリティ保護可能な領域によって、ブート時に 1 回のみ実行可能であり、新たなリセットが発生しない限りは絶対に実行できないコード領域が定義されます。

セキュリティ保護可能な領域の主な目的は、望ましくないアクセスからFlashメモリの特定の領域を保護することです。これにより、セキュアキー保管やセーフブートなどのソフトウェアのセキュリティサービスを実装できます。セキュリティ保護可能な領域は、メインFlashメモリに配置されています。信頼できるコードを実行するための専用領域です。セキュリティ保護可能な領域が設定されない場合には、残りのメインFlashメモリと同じように動作します。セキュリティ保護可能な領域が設定（Flash_CR レジスタの SEC_PROT1 ビットがセット）された場合には、セキュリティ保護可能な領域のプログラムまたは消去を試みると書込み保護エラーが発生し（WRPERR フラグがセットされます）、その領域からの読出しを試みると読出しエラーが発生します（RDERR フラグがセットされます）。

セキュリティ保護可能なメモリ領域のサイズは、FLASH_SEC レジスタの SEC_SIZE1[8:0] ビットフィールドによって定義されます。変更は、RDP がレベル 0 の場合にのみ可能です。内容は、RDP をレベル 1 からレベル 0 に変更すると消去されます。PCROP ページとオーバーラップしている場合も同様です。

セキュリティ保護可能なメモリ領域は、次のアドレス範囲で定義されます。バンクのベース･アドレス（このアドレスを含む）からバンクのベース･アドレス + [0x800 x SEC_SIZE1]（このアドレスを含まない）まで。

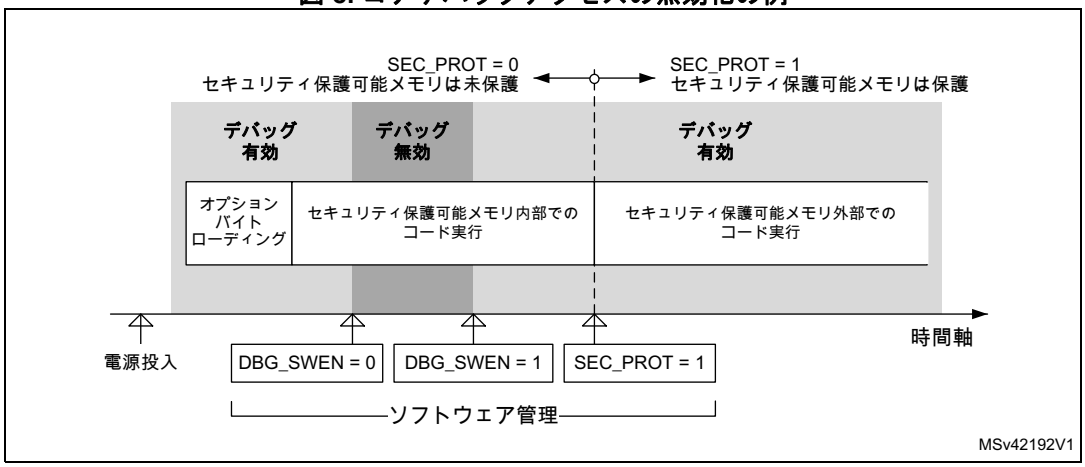
4.5.5 コア･デバッグ･アクセスの無効化

コアへのデバッグアクセスは、セキュリティ保護可能な領域で機密コードを実行したり機密データを処理するために、一時的に無効化できます。

RDP レベル 2 ではデバッガはハードウェアによって無効化されますが、それ以外の RDP レベルでは、デバッガは Flash_ACR レジスタの DBG_SWEN ビットを用いてソフトウェアによって無効化できます。

図 11 に、DBG_SWEN と SEC_PROT のビットの管理例を示します。

図 8. コアデバッグアクセスの無効化の例



4.5.6 Flashメモリからの強制ブート

セキュリティを強化し、信頼のチェーンを構築するため、Flash_SEC1R/Flash_SEC2R レジスタの BOOT_LOCK オプションビットでは、他のブートオプションにかかわらず、システムをメインFlash メモリから強制ブートすることができます。また、BOOT_LOCK ビットはいつでもセットすることができます。ただし、リセットは次の場合にのみ可能です。

- RDP をレベル 0 にセットした場合
- レベル 0 を要求し、完全全体消去を実行しているときに、RDP をレベル 1 にセットした場合

4.6 Flash 割込み

表 26. Flash 割込みリクエスト

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

1. EOP がセットされるのは EOPIE がセットされている場合のみです。
2. OPERR がセットされるのは ERRIE がセットされている場合のみです。

4.7 Flash レジスタ

4.7.1 Flash アクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x00

リセット値：0x0004 0601

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DBG_SWEN**：デバッグソフトウェア有効化

ソフトウェアは、このビットを使用してデバッガを有効／無効にすることができます。

0：デバッガは無効です。

1：デバッガは有効です。

ビット 17:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **SLEEP_PD**：SLEEP モードまたは低電力 SLEEP モード中のFlashパワーダウンモード

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

0：SLEEP モードまたは低電力 SLEEP モード中に Flash をアイドル･モードにします。

1：SLEEP モードまたは低電力 SLEEP モード中に Flash をパワーダウン･モードにします。

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

ビット 13 **RUN_PD**：RUN モードまたは低電力 RUN モード中のFlashパワーダウンモード

このビットは Flash_PDKEYR で書込み保護されています。

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

0：Flash をアイドル･モードにします。

1：Flash をパワーダウン･モードにします。

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

ビット 12 **DCRST**：データ･キャッシュのリセット

0：データ･キャッシュはリセットされません。

1：データ･キャッシュはリセットされます。

このビットに書き込めるのは、データキャッシュが無効のときだけです。

ビット 11 **ICRST**：命令キャッシュのリセット

0：命令キャッシュはリセットされません。

1：命令キャッシュはリセットされます。

このビットに書き込めるのは、命令キャッシュが無効のときだけです。

ビット 10 **DCEN**：データ･キャッシュ有効
0：データ･キャッシュは無効です。
1：データ･キャッシュは有効です。

ビット 9 **ICEN**：命令キャッシュ有効
0：命令キャッシュは無効です。
1：命令キャッシュは有効です。

ビット 8 **PRFTEN**：プリフェッチ有効化
0：プリフェッチは無効です。
1：プリフェッチは有効です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **LATENCY[3:0]**：遅延
これらのビットは、Flashアクセス時間に対する SYSCLK（システムクロック）周期の割合を表します。
0000：ウェイトステート 0 個
0001：ウェイトステート 1 個
0010：ウェイトステート 2 個
0011：ウェイトステート 3 個
0100：ウェイトステート 4 個
...1111：ウェイトステート 15 個

4.7.2 Flashパワーダウンキーレジスタ（Flash_PDKEYR）

アドレス･オフセット：0x04

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **PDKEYR[31:0]**：RUN モードにおけるパワーダウン Flash キー
Flash_ACR の RUN_PD ビットをアンロックするには、次の値を連続して書き込む必要があります。
PDKEY1：0x0415 2637
PDKEY2：0xFAFB FCFD

4.7.3 Flash キー レジスタ（FLASH_KEYR）

アドレス･オフセット：0x08

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **KEYR[31:0]**：Flash キー

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

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

4.7.4 Flashオプションキーレジスタ（Flash_OPTKEYR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **OPTKEYR[31:0]**：オプション･バイト･キー

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

KEY1：0x0819 2A3B

KEY2：0x4C5D 6E7F

4.7.5 Flashステータスレジスタ（Flash_SR）

アドレス･オフセット：0x10

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

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

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BSY**：ビジー

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

ビット 15 **OPTVERR**：オプション有効性エラー

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

1 を書き込むとクリアされます。

ビット 14 **RDERR**：PCROP 読出しエラー

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

1 を書き込むとクリアされます。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **FASTERR**：高速プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 8 **MISSERR**：高速プログラミング･データ･ミス･エラー

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

1 を書き込むとクリアされます。

ビット 7 **PGSERR**：プログラミング･シーケンス･エラー

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

1 を書き込むとクリアされます。

ビット 6 **SIZERR**：サイズ･エラー

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

1 を書き込むとクリアされます。

ビット 5 **PGAERR**：プログラミング整列エラー

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

1 を書き込むとクリアされます。

ビット 4 **WRPERR**：書込み保護エラー

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

1 を書き込むとクリアされます。

ビット 3 **PROGERR**：プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OPERR**：操作エラー

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

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

1 を書き込むとクリアされます。

ビット 0 **EOP**：操作終了

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

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

1 を書き込むとクリアされます。

4.7.6 Flash制御レジスタ（Flash_CR）

アドレス･オフセット：0x14

リセット値：0xC000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPT LOCK	Res.	SEC_ PROT1	OBL_ LAUNCH	RD ERRIE	ERR IE	EOP IE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPT STRT	START
rs	rs		rs	rc_w1	rw	rw	rw						rw	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PNB[7:0]								MER1	PER	PG
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：FLASH_CR ロック

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

ビット 30 **OPTLOCK**：オプションロック

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

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **SEC_PROT1**：セキュリティ保護可能な領域の保護ビット
このビットは、セキュリティ保護可能な領域へのアクセスをロックするために使用します。セキュリティ保護可能なメモリ領域を抜けるときにソフトウェアによってセットされ、1 回に限り書込み可能です。

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

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

ビット 26 **RDERRIE**：PCROP 読出しエラー割込み有効化

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

ビット 25 **ERRIE**：エラー割込み有効化

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

ビット 24 **EOPIE**：操作終了割込みイネーブル

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

0：EOP 割込みは無効です。

1：EOP 割込みは有効です。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSTPG**：高速プログラミング

0：高速プログラミングは無効です。

1：高速プログラミングは有効です。

ビット 17 **OPTSTRT**：オプション変更開始

このビットがセットされると、オプション操作がトリガされます。

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

ビット 16 **START**：スタート

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

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

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:3 **PNB[7:0]**：ページ番号選択

これらのビットは、消去するページを選択します。

00000000：ページ 0

00000001：ページ 1

.....

11111111：ページ 255

ビット 2 **MER1**：全体消去

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

ビット 1 **PER**：ページ消去

0：ページ消去は無効です。

1：ページ消去は有効です。

ビット 0 **PG**：プログラミング

0：Flash プログラミングは無効です。

1：Flash プログラミングは有効です。

4.7.7 Flash ECC レジスタ（Flash_ECCR）

アドレス･オフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

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

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

ビット 31 **ECCD**：ECC 検出

2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます（ECCC/ECCD が、あらかじめクリアされている場合のみ）。このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

ビット 30 **ECCC**：ECC 訂正

1 つの ECC エラーが検出されて訂正されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が以前にクリアされている場合のみ）。ECCCIE がセットされると、割込みが生成されます。

1 を書き込むとクリアされます。

ビット 29:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ECCCIE**：ECC 訂正割込み有効化

0：ECCC 割込み無効

1：ECCC 割込みは有効です。

このビットは、Flash_ECCR レジスタの ECCC ビットがセットされると割込み生成を有効にします。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SYSF_ECC**：システムFlash ECC 失敗

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

ビット 21:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **ADDR_ECC[18:0]**：ECC 失敗アドレス

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

4.7.8 Flashオプションレジスタ（Flash_OPTR）

アドレス･オフセット：0x20

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

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRHEN	NRST_MODE [1:0]		n BOOT0	nSW BOOT0	CCM SRAM_ RST	SRAM _PE	nBOOT1	PB4_PU PEN	Res.	Res.	WWDG_ SW	IWGD_ STDBY	IWDG_ StOP	IWDG_ SW
	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_S HDW	nRST_S TDBY	nRST_S TOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **IRHEN**：内部リセット･ホルダー･イネーブル･ビット

0：内部リセットは、NRST ピン上の単純なパルスとして伝播します。

1：内部リセットは、ロー･レベルと見なされるまで NRST ピンをローになるように駆動します。

ビット 29:28 **NRST_MODE[1:0]**：

00：予約済みです。

01：リセット入力専用：NRST ピンが Low レベルになるとシステム･リセットを生成します。内部リセットが NRST ピンに伝播することはありません。

10：GPIO：標準 GPIO パッドの機能を使用します。内部リセットのみ可能です。

11：双方向リセット：NRST ピンが、リセット入出力モードで設定されます（レガシーモード）。

ビット 27 **nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24 **SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル

0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。

1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

ビット 23 **nBOOT1**：ブート設定

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

ビット 22 **PB4_PUPEN**：PB4 プルアップ有効化

0：USB Power Delivery のデッド･バッテリ機能は有効、NJTRST プルアップは無効です。

1：USB Power Delivery のデッド･バッテリ機能は無効、NJTRST プルアップは有効です。

注： カテゴリ 4 デバイスの場合のみ（他のカテゴリでは予約済み）

ビット：21:20 予約済みであり、リセット値に保持する必要があります。

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

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

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

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

ビット 15 予約済みであり、リセット値に保持する必要があります。

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

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

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

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

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

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

4.7.9 Flash PCROP1 開始アドレスレジスタ（Flash_PCROP1SR）

アドレス･オフセット：0x24

リセット値：0xFFFF XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

アクセス：ノーウェイトステート（進行中のFlashメモリ操作がない場合）、ワードアクセス

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PCROP1_STRT[15:0]**：PCROP 領域開始オフセット
PCROP1_STRT は、PCROP 領域の最初のダブルワードを含んでいます。

4.7.10 Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）

アドレス･オフセット：0x28

リセット値：0xX000 XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

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

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域
このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。
0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。
1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PCROP1_END[15:0]**：PCROP 領域終了オフセット
PCROP1_END は、PCROP 領域の最後のダブルワードを含んでいます。

4.7.11 Flash WRP 領域 A アドレス･レジスタ（FLASH_WRP1AR）

アドレス･オフセット：0x2C

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **WRP1A_END[7:0]**：WRP 最初の領域「A」終了オフセット
WRP1A_END は、WRP の最初の領域の最終ページを含んでいます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

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

4.7.12 Flash WRP 領域 B アドレス･レジスタ（FLASH_WRP1BR）

アドレス･オフセット：0x30

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **WRP1B_END[7:0]**：WRP 2 番目の領域「B」終了オフセット
WRP1B_END は、WRP の 2 番目の領域の最終ページを含んでいます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **WRP1B_STRT[7:0]**：WRP 2 番目の領域「B」開始オフセット
WRP1B_STRT は、WRP の 2 番目の領域の最終ページを含んでいます。

4.7.13 Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）

アドレス･オフセット：0x70

リセット値：0xFFFX FXXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ Flash 領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **SEC_SIZE1[8:0]**：セキュリティ保護可能な領域に使用されているページ数がセットされます。
セキュリティ保護可能な領域は 0x0800 0000 から始まり、そのサイズは SEC_SIZE1 * ページ･サイズとなります。このフィールドはレベル 0 でのみ変更可能です。
レベル 1 で変更を試みるとサイレントに失敗し、レジスタ値は変更されません。

4.7.14 Flash レジスタマップ

表 27. Flash インタフェース - レジスタ・マップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_SWEN	Res.	Res.	Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY [3:0]			
	リセット値														1				0	0	0	0	1	1	0					0	0	0	1
0x04	FLASH_ PDKEYR	PDKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FLASH_KEYR	KEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FLASH_OPT KEYR	OPTKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FLASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY	OPTVERR	RDERR	Res.	Res.	Res.	Res.	FASTERR	MISSERR	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値																0	0	0					0	0	0	0	0	0	0		0	0
0x14	FLASH_CR	LOCK	OPTLOCK	Res.	SEC_PROT1	OBL_LAUNCH	RDERRIE	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPTSTRT	STRT	Res.	Res.	Res.	Res.	Res.	PNB[7:0]							MER1	PER	PG
	リセット値	1	1		1	0	0	0	0						0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x18	FLASH_ECCR	ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCCIE	Res.	SYSF_ECC	Res.	Res.	Res.	ADDR_ECC[18:0]																		
	リセット値	0	0						0		0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	FLASH_OPTR	Res.	Res.	Res.	Res.	nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	PB4_PUPEN	Res.	Res.	WWDG_SW	IWDG_STBY	IWDG_STOP	IWDG_SW	Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_ LEV[2:0]			RDP[7:0]							
	リセット値					X	X	X	X	X	X			X	X	X	X		X	X	X		X	X	X	X	X	X	X	X	X	X	X
0x24	FLASH_ PCROP1SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_STRT [15:0]															
	リセット値																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x28	FLASH_ PCROP1ER	PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_END [15:0]															
	リセット値	x																X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x2C	FLASH_ WRP1AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X
0x30	FLASH_ WRP1BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT [7:0]							
	リセット値									X	X	X	X	X	X	X	X									X	X	X	X	X	X	X	X

表 27. Flash インタフェース - レジスタ･マップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x70	FLASH_ SEC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [8:0]								
	リセット値																x								x	x	x	x	x	x	x	x	x

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

5 内蔵 Flash メモリ（Flash）カテゴリ 2 デバイスの場合

5.1 概要

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

また、命令プリフェッチおよびキャッシュラインでコードの実行を加速します。

5.2 Flashの主な機能

- 最大 128 KB の Flash メモリ（シングル･バンク）
- 64 ビットデータ幅に対応した Flash メモリ読出し操作
- ページ消去と全体消去

Flashメモリインタフェースの機能：

- Flashメモリ読出し操作
- Flashメモリプログラム／消去操作
- オプション（RDP）で有効化される読出し保護
- オプションで選択される 2 つの書込み保護領域
- オプションで定義される独自仕様コード読出し保護領域
- オプションで定義されるセキュリティ保護可能なメモリ領域
- ICode でプリフェッチ
- 命令キャッシュ：ICode で 4 x 64 ビットのキャッシュライン 32 本（1 KB RAM）
- データキャッシュ：DCode で 4 x 64 ビットのキャッシュライン 8 本（256B RAM）
- エラーコード訂正 ECC：64 ビットダブルワードごとに 8 ビット
  - 8 + 64 = 72 ビット、検出 2 ビット、訂正 1 ビット
- オプションバイトローダ
- 低電力モード

5.3 Flash の機能説明

5.3.1 Flash メモリの構成

Flashメモリは、以下のような主な特徴を備えています。

- 最大容量 128 KB（読出し幅 64 ビット）
- メインメモリの 1 つのシングルバンクに 128 KB を構成
- ページサイズ 2 KB
- 72 ビット幅のデータ読出し（64 ビット + ECC 8 ビット）
- ページ消去と全体消去
- 各ページには 256 KB の行が 8 行含まれます。

- 情報ブロックには次のものが含まれます。
  - システムメモリブートモードでデバイスがブートするシステムメモリ。この領域は STMicroelectronics によって予約済みで、次のいずれかのインタフェースを介したFlashメモリの再プログラムに使用するブートローダを含みます。USART、SPI、I2C、USB。これはデバイスの製造時に STMicroelectronics によってプログラムされており、誤った書込み／消去操作から保護されています。詳細については、www.st.com から入手可能な AN2606 を参照してください。
  - OTP（1 度だけプログラムが可能な）、ユーザデータ用の 1 KB（128 ダブルワード）。OTP データは消去不可で、1 度だけ書込み可能です。1 つのビットだけが 0 の場合、0x0000 0000 0000 0000 の値であってもダブルワード全体を書き込むことはできません。
  - ユーザ設定のオプションバイト。

メモリ構成は、表 28 に示されるメイン領域と情報ブロックに基づきます。

表 28. Flash モジュール - 64/128 KB の構成（読出し幅 64 ビット）

Flash 領域	Flashメモリアドレス	サイズ（バイト）	名前
メインメモリ 64/128 KB(1)	0x0800 0000 - 0x0800 07FF	2 K	ページ 0
	0x0800 0800 - 0x0800 0FFF	2 K	ページ 1
	0x0800 1000 - 0x0800 17FF	2 K	ページ 2
	0x0800 1800 - 0x0800 1FFF	2 K	ページ 3
	-	-	-
	-	-	-
	-	-	-
	-	-	-
	0x0801 F800 - 0x0801 FFFF	2 K	ページ 63
情報ブロック	0x1FFF 0000 - 0x1FFF 6FFF	28 K	システムメモリ
	0x1FFF 7000 - 0x1FFF 73FF	1 K	OTP エリア
	0x1FFF 7800 - 0x1FFF 782F	48	オプションバイト

1. 64 KB デバイスの場合：ページ 0 ~ ページ 31

5.3.2 エラーコード訂正（ECC）

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

- 1 つのエラー検出および訂正
- 2 つのエラー検出

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

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

ECC エラーが検出されると、失敗したダブルワードのアドレスが、Flash_ECCR レジスタの ADDR_ECC[20:0] に保存されます。ADDR_ECC[2:0] は常にクリアされます。

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

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

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

5.3.3 読出しアクセスの遅延

データをFlashメモリから正しく読み出すには、CPU クロック（HCLK）の周波数およびデバイスの内部電圧範囲 V_{CORE} に従って、Flash アクセス制御レジスタ（FLASH_ACR）でウェイトステート（LATENCY）の数を正しくプログラムする必要があります。セクション 6.1.5：ダイナミック電圧スケーリングの管理 を参照してください。表 29 には、ウェイトステートと CPU クロック周波数の対応を示します。

表 29. CPU クロック（HCLK）周波数によるウェイトステート数

ウェイトステート（WS）（LATENCY）	HCLK (MHz)		
	V_{CORE} レンジ 1 ブースト·モード	V_{CORE} レンジ 1 通常モード	V_{CORE} レンジ 2
0 WS（1 CPU サイクル）	≤ 34	≤ 30	≤ 12
1 WS（2 CPU サイクル）	≤ 68	≤ 60	≤ 24
2 WS（3 CPU サイクル）	≤ 102	≤ 90	≤ 26
3 WS（4 CPU サイクル）	≤ 136	≤ 120	-
4 WS（5 CPU サイクル）	≤ 170	≤ 150	-

リセット後、CPU クロック周波数は 16 MHz であり、Flash_ACR レジスタでは 1 ウェイトステート（WS）が設定されます。

CPU 周波数を変更するときは、Flash メモリにアクセスするために必要なウェイトステート数を適切な値にするために、以下のソフトウェア·シーケンスを適用する必要があります。

CPU 周波数の増加：

1. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
2. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が反映されていることを確認します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロックプリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が減少する場合には、最初にそのステップを実行してからもう一方を実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。

CPU 周波数の減少：

1. RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更します。
2. 必要であれば、RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。
3. 以下を行った場合の CPU 周波数変化の変化量を解析します。
   - RCC_CFGR レジスタの SW ビットによって定義されるクロックソースの変更
   - RCC_CFGR レジスタの HPRE ビットによって定義される CPU クロック･プリスケーラ

上記 2 ステップのいずれかによって CPU 周波数が増加する場合には、最初にもう一方のステップを実行してからそのステップを実行します。そうではない場合、RCC_CFGR レジスタに SW ビットを書き込んで CPU クロックソースを変更した後に、（必要であれば）RCC_CFGR レジスタに HPRE ビットを書き込んで CPU クロックプリスケーラを変更します。

4. RCC_CFGR レジスタでクロックソースステータス（SWS ビット）や AHB プリスケーラの値（HPRE ビット）を読み出して新しい CPU クロックソースや新しい CPU クロックプリスケーラの値がそれぞれ考慮されていることを確認します。
5. Flash アクセス制御レジスタ（FLASH_ACR）の LATENCY ビットに新しいウェイトステート数をプログラムします。
6. Flash_ACR レジスタを読み出してFlashメモリへのアクセスに新しいウェイトステート数が使用されていることを確認します。

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

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

プロセッサのフルパフォーマンスを引き出すため、アクセラレータは命令プリフェッチキューと、64 ビットのFlashメモリからのプログラム実行速度を上昇させるブランチキャッシュを実行します。CoreMark ベンチマークによると、ART アクセラレータによって達成されるパフォーマンスは、最大 CPU 周波数 170 MHz におけるFlashメモリからの 0 ウェイトステートプログラム実行と同等となります。

命令プリフェッチ

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

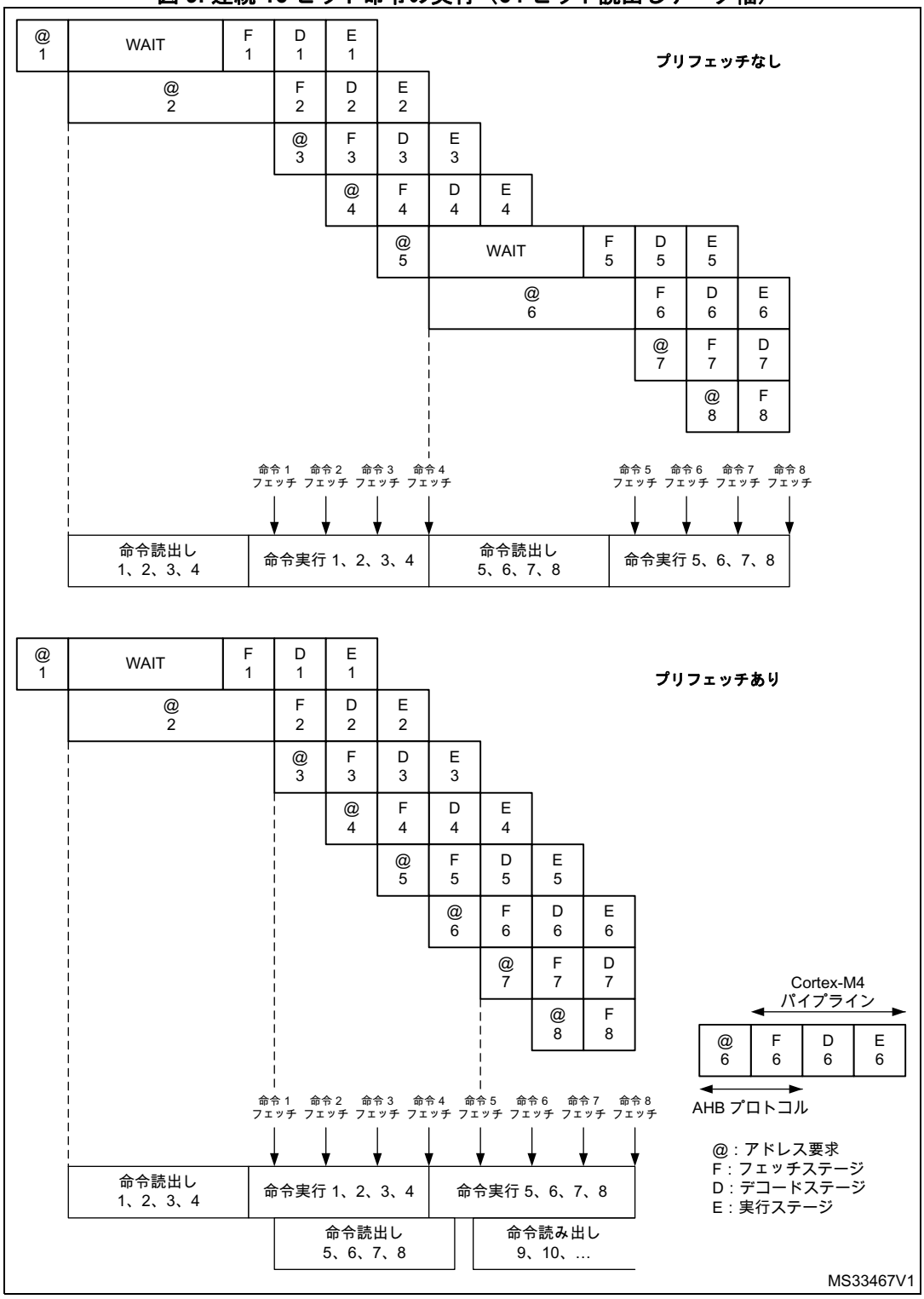
各Flashメモリ読出し操作では、起動されるプログラムに応じて、32 ビットの命令 2 個または 16 ビットの命令 4 個の 64 ビットが提供されます。この 64 ビットの現在の命令ラインは、現在のバッファに保存されますので、連続コードの場合は、その前の読出し命令ラインの実行に 2 個以上の CPU サイクルが必要となります。

ICode バスでのプリフェッチを使用すると、CPU によって現在の命令ラインが要求されている間にFlashメモリから次の連続命令ラインを読み出すことができます。

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

図 9 は、Flashメモリへのアクセスに 3 WS が必要なときに、プリフェッチがある場合とない場合の連続 16 ビット命令の実行を示しています。

図 9. 連続 16 ビット命令の実行（64 ビット読出しデータ幅）



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

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

命令キャッシュメモリ（I キャッシュ）

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

命令キャッシュメモリはシステムリセット後に有効になります。

データキャッシュメモリ（D キャッシュ）

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

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

データキャッシュメモリはシステムリセット後に有効になります。

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

オプションバイトブロックのデータは、キャッシュできません。

5.3.5 Flashのプログラムおよび消去操作

STM32G4 シリーズ 内蔵Flashメモリは、インサーキットプログラミングまたはアプリケーション内プログラミングを使用してプログラム可能です。

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

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

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

Flashメモリの消去およびプログラムは電圧スケーリングレンジ 1 でのみ可能です。PWR_CR1 の VOS[1:0] ビットは 01b に設定してください。

Flashメモリへのプログラム／消去操作中にFlashメモリを読み出そうとすると、バスがストールされます。読出し操作は、プログラム／消去操作が完了すると正しく処理されます。

Flashメモリのロック解除

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

1. Flash キー レジスタ（FLASH_KEYR）に KEY1 = 0x45670123 を書き込みます。
2. Flash_KEYR レジスタに KEY2 = 0xCDEF89AB を書き込みます。

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

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

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

5.3.6 Flashメインメモリの消去シーケンス

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

ページ消去

ページを消去するには、次の手順に従います。

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

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

消去するページが、（WRP または PCROP によって）書込み保護された領域に属している場合、WRPERR がセットされてページ消去リクエストが中止されます。

全体消去

全体消去を実行するには、次の手順に従います。

1. Flash_SR レジスタの BSY ビットを確認し、進行中のFlashメモリ操作がないことを確認します。
2. 以前のプログラミングによるエラー･プログラミング･フラグをすべて確認してクリアします。しない場合、PGSERR がセットされます。
3. Flash 制御レジスタ（FLASH_CR）の MER1 ビットをセットします。
4. FLACH_CR レジスタの STRT ビットをセットします。
5. Flashステータスレジスタ（Flash_SR）の BSY ビットがクリアされるまで待ちます。

注： **内部オシレータ HSI16（16 MHz）は、RCC_CR レジスタの HSION であらかじめ有効にしている場合を除き、STRT ビットをセットすると自動的に有効になり、クリアすると自動的に無効になります。**

Flash メモリに（WRP または PCROP によって）書込み保護された領域が含まれる場合、WRPERR がセットされて全体消去リクエストが中止されます。

5.3.7 Flash メイン･メモリのプログラミング･シーケンス

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

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

ダブルワード（2 x 32 ビットデータ）のプログラムのみ可能です。

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

標準プログラミング

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

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

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

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

ECC は、プログラムするダブルワードから計算されます。

高速プログラミング

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

メインメモリのみ高速プログラミングモードでプログラム可能です。

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

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

注： **読出し操作が進行している際に、Flash が高速プログラミング･モードでの書込みを試みた場合、システム通知なしでプログラミングが中止されます（エラー･フラグはセットされません）。**

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

32 ダブルワードは連続で書き込む必要があります。高電圧は、すべてのプログラミングに対してFlashで維持されます。連続した 2 つのダブルワードの間の書込みリクエストの最大時間は、最初のダブルワードに対して、Flashがプログラムされている時間（2 x 25 us 前後）です。2 番目のダブルワードがこのプログラムされている時間を越える場合、高速プログラミングは割り込まれ、MISSERR がセットされます。

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

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

プログラミングエラー

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

- **PROGERR**：プログラミングエラー

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

- **SIZERR**：サイズ･プログラミング･エラー

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

- **PGAERR**：配置プログラミング･エラー

  次のいずれかの条件が発生した場合、PGAERR がセットされます。

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

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

- **PGSERR**：プログラミング･シーケンス･エラー

  次のいずれかの条件が発生した場合、PGSERR がセットされます。

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

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

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

  – 全体消去シーケンス：MER1 がセットされたときに PG、FSTPG、および PER がクリアされない場合

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

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

- **WRPERR**：書込み保護エラー

  次のいずれかの条件が発生した場合、WRPERR がセットされます。

  – 書込み保護領域（WRP）、PCROP 領域、またはセキュリティ保護可能メモリ領域でプログラミングまたは消去を試みた場合

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

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

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

- **MISSERR**：高速プログラミングデータミスエラー

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

- **FASTERR**：高速プログラミングエラー

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

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

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

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

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

WRPERR（保護エラーフラグ）

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

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

プログラミングとキャッシュ

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

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

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

5.4 Flashオプションバイト

5.4.1 オプションバイトの説明

オプションバイトは、アプリケーション要件によってエンドユーザが設定します。設定の例として、ウォッチドッグをハードウェアまたはソフトウェアモードに選択できます（セクション 5.4.2：オプションバイトのプログラミングを参照してください）。

ダブルワードは、オプションバイトで次に示すように分割されます。

表 30. オプションバイトのフォーマット

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

情報ブロック内のこれらのバイトの構成は、表 31：オプションバイトの構成 に示すとおりです。

オプションバイトは、表 31：オプションバイトの構成 にリストされたメモリ位置またはオプションバイトレジスタから読み出すことができます。

- Flashオプションレジスタ（Flash_OPTR）
- Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）
- Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）
- Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）
- Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）

表 31. オプションバイトの構成

アドレス	[63:56]	[55:48]	[47:40]	[39:32]	[31:24]	[23:16]	[15:8]	[7:0]
1FFF7800	USER OPT			RDP	USER OPT			RDP
1FFF7808	未使用		未使用および PCROP1_STRT[13:0]		未使用		未使用および PCROP1_STRT[13:0]	
1FFF7810	PCROP_RDP および 未使用		未使用および PCROP1_END[13:0]		PCROP_RDP および 未使用		未使用および PCROP1_END[13:0]	
1FFF7818	未使用	WRP1A_END [5:0]	未使用	WRP1A_STRT [5:0]	未使用	WRP1A_END [5:0]	未使用	WRP1A_STRT [5:0]
1FFF7820	未使用	WRP2A_END [5:0]	未使用	WRP2A_STRT [5:0]	未使用	WRP2A_END [5:0]	未使用	WRP2A_STRT [5:0]
1FFF7828	未使用	BOOT_LOCK	未使用	SEC_SIZE1	未使用	BOOT_LOCK	未使用	SEC_SIZE1

ユーザ･オプション･バイトおよび読出し保護オプション･バイト

Flashメモリアドレス：0x1FFF 7800

ST 出荷時の値：0xFFEF F8AA

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRH_EN	NRST_MODE[1:0]		nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	Res.	Res.	Res.	WWDG_SW	IWGD_STDBY	IWDG_STOP	IWDG_SW
	r	r	r	r	r	r	r	r				r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	r	r	r		r	r	r	r	r	r	r	r	r	r	r

ビット 31　予約済みであり、リセット値に保持する必要があります。

ビット 30　**IRH_IN**：PG10の内部リセットホルダ

0：IRH は無効です。

1：IRH は有効です。

ビット 29:28　**NRST_MODE**：PG10 パッド･モード

00：入出力をリセット

01：入力のみをリセット

10：GPIO

11：入出力をリセット

ビット 27　**nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26　**nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25　**CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24　**SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル

0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。

1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

ビット 23　**nBOOT1**：ブート設定

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

ビット 22:20　予約済みであり、リセット値に保持する必要があります。

ビット 19　**WWDG_SW**：ウィンドウ型ウォッチドッグの選択

0：ハードウェアによるウィンドウ型ウォッチドッグです。

1：ソフトウェアによるウィンドウ型ウォッチドッグです。

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

0：STANDBY モードでの独立型ウォッチドッグカウンタを停止します。

1：STANDBY モードでの独立型ウォッチドッグカウンタが動作します。

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

0：STOP モードでの独立型ウォッチドッグカウンタを停止します。

1：STOP モードでの独立型ウォッチドッグカウンタが動作します。

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

ビット 15 予約済みであり、リセット値に保持する必要があります。

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

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

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

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

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

PCROP1 開始アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7808

リセット値：0xFFFF FFFF（ST 出荷時の値）

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

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **PCROP1_STRT[13:0]**：PCROP 領域開始オフセット
PCROP1_STRT は、バンク 1 の PCROP 領域の最初のダブルワードを含んでいます。

PCROP1 終了アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7810

リセット値：0x00FF 0000（ST 出荷時の値）

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

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

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域
このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。
0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。
1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **PCROP1_END[13:0]**：バンク 1 PCROP 領域終了オフセット
PCROP1_END は、PCROP 領域の最後のダブルワードを含んでいます。

WRP1 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7818

リセット値：0xFF00 FFFF（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [5:0]					
										r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [5:0]					
										r	r	r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **WRP1A_END[5:0]**：WRP 最初の領域「A」終了オフセット
WRP1A_END は、WRP の最初の領域の最終ページを含んでいます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WRP1A_STRT[5:0]**：WRP 最初の領域「A」開始オフセット
WRP1A_STRT は、WRP の最初の領域の最初のページを含んでいます。

WRP2 領域 A アドレスオプションバイト

Flashメモリアドレス：0x1FFF 7820

リセット値：0xFF00 FFFF（ST 出荷時の値）

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **WRP2A_END[5:0]**：WRP 2 番目の領域「B」終了オフセット
WRP1B_END は、WRP の 2 番目の領域の最終ページを含んでいます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WRP2A_STRT[5:0]**：WRP 2 番目の領域開始オフセット
WRP1B_STRT は、WRP の 2 番目の領域の最終ページを含んでいます。

セキュリティ保護可能メモリ領域オプションバイト

Flashメモリアドレス：0x1FFF7828

リセット値：0xFF00FF00（ST 出荷時の値）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [6:0]						
									r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ Flash 領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **SEC_SIZE1[6:0]**：セキュリティ保護可能な領域のサイズ
セキュリティ保護可能な Flash メモリのページ数を含んでいます。

5.4.2 オプションバイトのプログラミング

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

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

ソフトウェアによって OPTLOCK ビットをセットすると、ユーザ･オプションを意図せぬ消去やプログラム操作から保護できます。

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

ユーザオプションの変更

オプション･バイトは、メイン･メモリのユーザ･アドレスとは異なる方法でプログラムされます。ユーザ･オプション値を変更するには、次の手順に従います。

1. Flashステータスレジスタ（Flash_SR）の BSY ビットを確認し、進行中の Flash メモリ操作がないことを確認します。
2. 上記のクリアシーケンスで OPTLOCK オプションロックビットをクリアします。
3. オプション値を次のオプションレジスタに書き込みます。Flashオプションレジスタ（Flash_OPTR）、Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）、Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）、Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）、Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）。
4. Flash制御レジスタ（Flash_CR）でオプションスタートビット OPTSTRT をセットします。
5. BSY ビットがクリアされるのを待ちます。

注： **まず両方のユーザオプションバイトページを消去し、すべてのオプションバイトをFlashオプションレジスタに含まれている値でプログラムすると、一方のオプションの値の変更が自動的に実行されます。**

オプションバイトローディング

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

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

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

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

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

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

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

システムリセットの立ち上がりでは、内部オプションレジスタがソフトウェアで読出しおよび書込み可能なオプションレジスタ（Flash_OPTR、Flash_PCROP1SR、Flash_PCROP1ER、Flash_WRP1AR、Flash_WRP1BR）にコピーされます。これらのレジスタはオプション変更にも使用されます。これらのレジスタがユーザによって変更されない場合、システムのオプション状態が反映されます。詳細は、セクション ：ユーザオプションの変更を参照してください。

5.5 Flashメモリの保護

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

5.5.1 読出し保護（RDP）

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

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

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

Flashメモリは、RDP オプションバイトとその補数に 表 32 に示す値のペアが含まれる場合に保護されます。

表 32. Flashメモリの読出し保護ステータス

RDP バイト値	RDP 補数	読出し保護レベル
0xAA	0x55	レベル 0（ST 出荷時の設定）
0xAA または 0xCC を除く任意の値	0x55 および 0x33 以外の任意の値（相補である必要はない）	レベル 1
0xCC	0x33	レベル 2

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

レベル 0：保護なし

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

レベル 1：読出し保護

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

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

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

レベル 2：デバッグなし

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

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

注： **リセット後、デバッグ機能は無効化されます。**

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

読出し保護レベルの変更

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

レベル 1 からレベル 0 に移動するために RDP が値 0xAA に再プログラムされると、PCROP_RDP が Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）でセットされている場合、Flashメインメモリの全体消去が実行されます。バックアップレジスタ（RTC の RTC_BKPxR）および CCM SRAM も消去されます。PCROP 保護を除くユーザオプションは、Flash_OPTR、Flash_WRPxyR（x = 1 および y = A または B）からコピーされた以前の値にセットされます。PCROP は無効です。OTP エリアは全体消去に影響されませんので変化しません。

ビット PCROP_RDP が Flash_PCROP1ER でクリアされると、完全全体消去は、PCROP によって保護されたページを除いて連続したページを消去する部分全体消去に代わります。これは、PCROP コードを維持するために実行されます。Flash メモリが消去された場合のみ、オプションが以前の値で再プログラムされます。これは、Flash_PCROPxSR レジスタおよび Flash_PCROPxER レジスタ（x = 1）でも同様です。

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

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

図 10. 読出し保護（RDP）レベルの変更

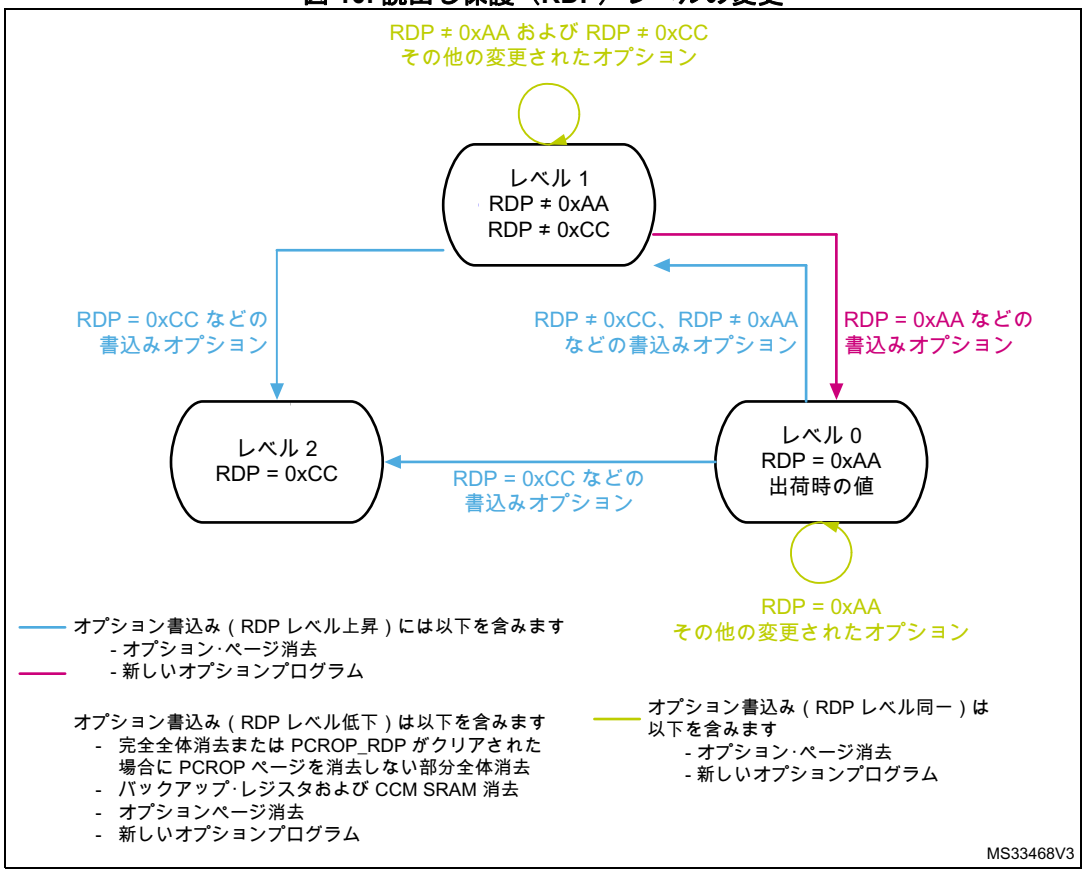


表 33. アクセス状態 対 保護レベルと実行モード

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
Flashメインメモリ	1	可能	可能	可能	不可	不可	不可 (3)
	2	可能	可能	可能	N/A	N/A	N/A
システムメモリ (2)	1	可能	不可	不可	可能	不可	不可
	2	可能	不可	不可	N/A	N/A	N/A
オプションバイト	1	可能	可能(3)	可能	はい	可能 (3)	可能
	2	可能	不可	不可	N/A	N/A	N/A
OTP	1	可能	可能(4)	N/A	不可	不可	N/A
	2	可能	可能 (4)	N/A	N/A	N/A	N/A
バックアップレジスタ	1	可能	可能	N/A	不可	不可	不可(5)
	2	可能	可能	N/A	N/A	N/A	N/A

表 33. アクセス状態 対 保護レベルと実行モード （続き）

領域	保護レベル	ユーザ実行（Flashからのブート）			デバッグ／RAM からのブート／ローダからのブート(1)		
		読出し	書込み	消去	読出し	書込み	消去
CCM SRAM	1	可能	可能	N/A	不可	不可	不可(6)
	2	可能	可能	N/A	N/A	N/A	N/A

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

5.5.2 独自仕様コード読出し保護（PCROP）

Flashメモリの一部を、第三者の読出しおよび書込みから保護することができます。保護領域は実行専用です。STM32 CPU のみが命令コードとしてアクセスでき、その他すべてのアクセス（DMA、デバッグ、CPU のデータ読出し、書込み、消去）は厳しく禁じられています。PCROP 領域の単位はダブルワード（64 ビット）です。追加のオプションビット（PCROP_RDP）で、RDP 保護がレベル 1 からレベル 0 に変更された場合、PCROP 領域を消去するしないを選択できます。（読出し保護レベルの変更を参照）。

各 PCROP 領域は、物理的なFlashアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは PCROP アドレスレジスタのFlash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）、Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）で定義されます。

- PCROPx （x = 1）領域は、次のアドレス範囲で定義されます。Flash メモリのベース･アドレス + [PCROPx_STRT x 0x8]（このアドレスを含む）からFlash メモリのベース･アドレス + [(PCROPx_END + 1) x 0x8]（このアドレスを含まない）PCROP 領域の最小サイズは 2 ダブルワード（128 ビット）です。

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

- Flashでのブートが選択された場合、Flash_PCROP1SR および Flash_PCROP1ER レジスタは次のようにプログラムする必要があります。
  - PCROP1_STRT = 0xC5F0
  - PCROP1_END = 0xE000

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

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

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

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

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

オプションビット PCROP_RDP をクリアする際に、RDP をレベル 1 からレベル 0 に変更している場合、PCROP 領域を維持するために完全全体消去が部分全体消去に代わります（読出し保護レベルの変更を参照してください）。この場合、PCROP1_STRT および PCROP1_END も消去されません。

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

表 34. PCROP 保護(1)

PCROPx レジスタ値 （x = 1）	PCROP 保護領域
PCROPx_offset_strt > PCROPx_offset_end	PCROP 領域なし
PCROPx_offset_strt < PCROPx_offset_end	PCROPx_offset_strt と PCROPx_offset_end の間の領域が保護されます。 以下の値を書込み可能です。 – PCROPx_offset_strt に下の値 – PCROPx_offset_end に上の値

1. PCROP 領域の最小サイズは 2 ダブルワード（PCROPx_offset_strt と PCROPx_offset_end）です。

5.5.3 書込み保護（WRP）

Flashメモリのユーザエリアを、不要な書込み操作から保護できます。以下のいずれかを指定可能です。

- 2 つの書込み保護（WRP）領域を、ページ（2 KB）単位で定義できます。

各領域は、物理的Flashのベースアドレスに関連する開始ページオフセットと終了ページオフセットにより定義されます。これらのオフセットは次の WRP アドレスレジスタで定義されます。Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）、Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）。

WRP「y」領域（x = 1、y = A、B）は、次のアドレス範囲で定義されます。Flash メモリのベース･アドレス+ [WRPxy_STRT x 0x800]（このアドレスを含む）からFlash メモリのベース･アドレス + [(WRPxy_END+1) x 0x800]（このアドレスを含まない）

たとえば、アドレス 0x0801 2800（このアドレスを含む）からアドレス 0x0801 87FF（このアドレスを含む）までを WRP で保護するには、次のようにします。

- Flash のブートが選択されている場合、Flash_WRP1AR レジスタは次のようにプログラムする必要があります。
  - WRP1A_STRT = 0x25
  - WRP1A_END = 0x30

  Flash_WRP1BR の WRP1B_STRT と WRP1B_END が代わりに使用できます（Flash メモリの領域 B）。

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

Flashメモリの書込み保護されているエリアに消去／プログラム操作の実施を試みると、Flash_SR レジスタで書込み保護エラーフラグ（WRPERR）がセットされます。このフラグは、次の領域への書込みアクセスのすべてでもセットされます。

- OTP エリア
- ICP など書込みできないFlashメモリの部分
- PCROP エリア

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

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

表 35. WRP 保護

WRP レジスタの値（x = 1、y = A/B）	WRP 保護領域
WRPxy_STRT = WRPxy_END	ページ WRPxy は保護されます。
WRPxy_STRT > WRPxy_END	WRP 領域なし。
WRPxy_STRT < WRPxy_END	WRPxy_STRT から WRPxy_END までのページは保護されます。

5.5.4 セキュリティ保護可能な領域

セキュリティ保護可能な領域によって、ブート時に 1 回のみ実行可能であり、新たなリセットが発生しない限りは絶対に実行できないコード領域が定義されます。

セキュリティ保護可能な領域の主な目的は、望ましくないアクセスからFlashメモリの特定の領域を保護することです。これにより、セキュアキー保管やセーフブートなどのソフトウェアのセキュリティサービスを実装できます。セキュリティ保護可能な領域は、メインFlashメモリに配置されています。信頼できるコードを実行するための専用領域です。セキュリティ保護可能な領域が設定されない場合には、残りのメインFlashメモリと同じように動作します。セキュリティ保護可能な領域が設定（Flash_CR レジスタの SEC_PROT1 ビットがセット）された場合には、セキュリティ保護可能な領域のプログラムまたは消去を試みると書込み保護エラーが発生し（WRPERR フラグがセットされます）、その領域からの読出しを試みると読出しエラーが発生します（RDERR フラグがセットされます）。

セキュリティ保護可能なメモリ領域のサイズは、FLASH_SEC レジスタの SEC_SIZE1[6:0] ビットフィールドによって定義されます。変更は、RDP がレベル 0 の場合にのみ可能です。内容は、RDP をレベル 1 からレベル 0 に変更すると消去されます。PCROP ページとオーバーラップしている場合も同様です。

セキュリティ保護可能なメモリ領域は、次のアドレス範囲で定義されます。バンクのベース･アドレス（このアドレスを含む）からバンクのベース･アドレス + [0x800 x SEC_SIZE1]（このアドレスを含まない）まで。

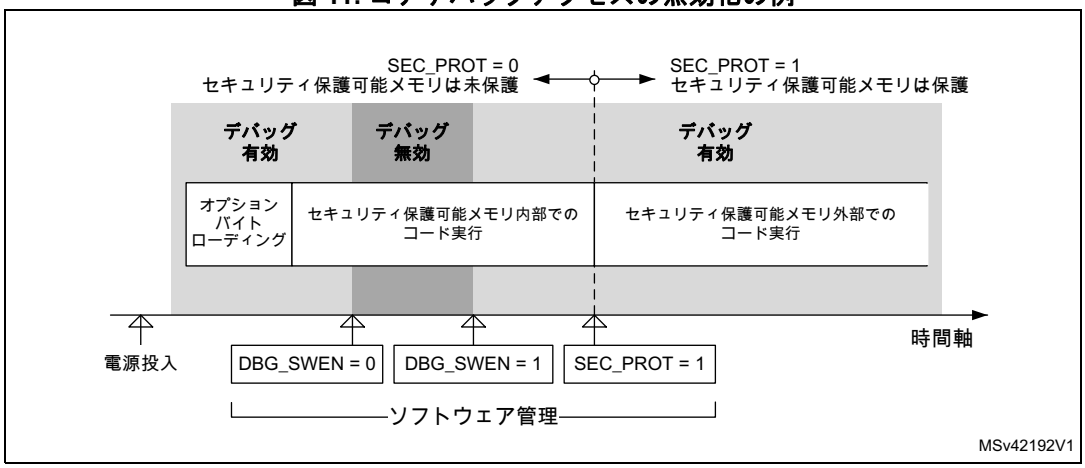
5.5.5 コア･デバッグ･アクセスの無効化

コアへのデバッグアクセスは、セキュリティ保護可能な領域で機密コードを実行したり機密データを処理するために、一時的に無効化できます。

RDP レベル 2 ではデバッガはハードウェアによって無効化されますが、それ以外の RDP レベルでは、デバッガは Flash_ACR レジスタの DBG_SWEN ビットを用いてソフトウェアによって無効化できます。

図 11 に、DBG_SWEN と SEC_PROT のビットの管理例を示します。

図 11. コアデバッグアクセスの無効化の例



5.5.6 Flashメモリからの強制ブート

セキュリティを強化し、信頼のチェーンを構築するため、Flash_SEC1R/Flash_SEC2R レジスタの BOOT_LOCK オプションビットでは、他のブートオプションにかかわらず、システムをメインFlash メモリから強制ブートすることができます。また、BOOT_LOCK ビットはいつでもセットすることができます。ただし、リセットは次の場合にのみ可能です。

- RDP をレベル 0 にセットした場合
- レベル 0 を要求し、完全全体消去を実行しているときに、RDP をレベル 1 にセットした場合

5.6 Flash 割込み

表 36. Flash 割込みリクエスト

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

1. EOP がセットされるのは EOPIE がセットされている場合のみです。
2. OPERR がセットされるのは ERRIE がセットされている場合のみです。

5.7 Flash レジスタ

5.7.1 Flash アクセス制御レジスタ（FLASH_ACR）

アドレスオフセット：0x00

リセット値：0x0004 0601

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY[3:0]			
	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DBG_SWEN**：デバッグソフトウェア有効化

ソフトウェアは、このビットを使用してデバッガを有効／無効にすることができます。

0：デバッガは無効です。

1：デバッガは有効です。

ビット 17:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **SLEEP_PD**：SLEEP モードまたは低電力 SLEEP モード中のFlashパワーダウンモード

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

0：SLEEP モードまたは低電力 SLEEP モード中に Flash をアイドル･モードにします。

1：SLEEP モードまたは低電力 SLEEP モード中に Flash をパワーダウン･モードにします。

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

ビット 13 **RUN_PD**：RUN モードまたは低電力 RUN モード中のFlashパワーダウンモード

このビットは Flash_PDKEYR で書込み保護されています。

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

0：Flash をアイドル･モードにします。

1：Flash をパワーダウン･モードにします。

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

ビット 12 **DCRST**：データ･キャッシュのリセット

0：データ･キャッシュはリセットされません。

1：データ･キャッシュはリセットされます。

このビットに書き込めるのは、データキャッシュが無効のときだけです。

ビット 11 **ICRST**：命令キャッシュのリセット

0：命令キャッシュはリセットされません。

1：命令キャッシュはリセットされます。

このビットに書き込めるのは、命令キャッシュが無効のときだけです。

ビット 10 **DCEN**：データ･キャッシュ有効
0：データ･キャッシュは無効です。
1：データ･キャッシュは有効です。

ビット 9 **ICEN**：命令キャッシュ有効
0：命令キャッシュは無効です。
1：命令キャッシュは有効です。

ビット 8 **PRFTEN**：プリフェッチ有効化
0：プリフェッチは無効です。
1：プリフェッチは有効です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **LATENCY[3:0]**：遅延
これらのビットは、Flashアクセス時間に対する SYSCLK（システムクロック）周期の割合を表します。
0000：ウェイトステート 0 個
0001：ウェイトステート 1 個
0010：ウェイトステート 2 個
0011：ウェイトステート 3 個
0100：ウェイトステート 4 個
...1111：ウェイトステート 15 個

5.7.2 Flashパワーダウンキーレジスタ（Flash_PDKEYR）

アドレス･オフセット：0x04

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **PDKEYR[31:0]**：RUN モードにおけるパワーダウン Flash キー
Flash_ACR の RUN_PD ビットをアンロックするには、次の値を連続して書き込む必要があります。
PDKEY1：0x0415 2637
PDKEY2：0xFAFB FCFD

5.7.3 Flash キー レジスタ（FLASH_KEYR）

アドレス･オフセット：0x08

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **KEYR[31:0]**：Flash キー

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

KEY1：0x4567 0123

KEY2：0xCDEF 89AB

5.7.4 Flashオプションキーレジスタ（Flash_OPTKEYR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード･アクセス

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

ビット 31:0 **OPTKEYR[31:0]**：オプション･バイト･キー

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

KEY1：0x0819 2A3B

KEY2：0x4C5D 6E7F

5.7.5 Flashステータスレジスタ（Flash_SR）

アドレス･オフセット：0x10

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTV ERR	RD ERR	Res.	Res.	Res.	Res.	FAST ERR	MISS ERR	PGS ERR	SIZ ERR	PGA ERR	WRP ERR	PROGE RR	Res.	OP ERR	EOP
rc_w1	rc_w1					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BSY**：ビジー

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

ビット 15 **OPTVERR**：オプション有効性エラー

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

1 を書き込むとクリアされます。

ビット 14 **RDERR**：PCROP 読出しエラー

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

1 を書き込むとクリアされます。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **FASTERR**：高速プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 8 **MISSERR**：高速プログラミング･データ･ミス･エラー

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

1 を書き込むとクリアされます。

ビット 7 **PGSERR**：プログラミング･シーケンス･エラー

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

1 を書き込むとクリアされます。

ビット 6 **SIZERR**：サイズ･エラー

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

1 を書き込むとクリアされます。

ビット 5 **PGAERR**：プログラミング配置エラー

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

1 を書き込むとクリアされます。

ビット 4 **WRPERR**：書込み保護エラー

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

1 を書き込むとクリアされます。

ビット 3 **PROGERR**：プログラミングエラー

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

1 を書き込むとクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **OPERR**：操作エラー

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

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

1 を書き込むとクリアされます。

ビット 0 **EOP**：操作終了

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

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

1 を書き込むとクリアされます。

5.7.6 Flash制御レジスタ（Flash_CR）

アドレス･オフセット：0x14

リセット値：0xC000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	OPT LOCK	Res.	SEC_ PROT1	OBL_ LAUNCH	RD ERRIE	ERR IE	EOP IE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPT STRT	STRT
rs	rs		rs	rc_w1	rw	rw	rw						rw	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	PNB[6:0]							MER1	PER	PG
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：FLASH_CR ロック

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

ビット 30 **OPTLOCK**：オプションロック

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

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **SEC_PROT1**：セキュリティ保護可能な領域の保護ビット
このビットは、セキュリティ保護可能な領域へのアクセスをロックするために使用します。セキュリティ保護可能なメモリ領域を抜けるときにソフトウェアによってセットされ、1 回に限り書込み可能です。

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

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

ビット 26 **RDERRIE**：PCROP 読出しエラー割込み有効化

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

ビット 25 **ERRIE**：エラー割込み有効化

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

ビット 24 **EOPIE**：操作終了割込みイネーブル

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

0：EOP 割込みは無効です。

1：EOP 割込みは有効です。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSTPG**：高速プログラミング

0：高速プログラミングは無効です。

1：高速プログラミングは有効です。

ビット 17 **OPTSTRT**：オプション変更開始

このビットがセットされると、オプション操作がトリガされます。

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

ビット 16 **START**：スタート

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

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

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:3 **PNB[6:0]**：ページ番号選択

これらのビットは、消去するページを選択します。

00000000 : ページ 0

00000001 : ページ 1

.....

11111111 : ページ 255

ビット 2 **MER1**：全体消去

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

ビット 1 **PER**：ページ消去

0：ページ消去は無効です。

1：ページ消去は有効です。

ビット 0 **PG**：プログラミング

0：Flash プログラミングは無効です。

1：Flash プログラミングは有効です。

5.7.7 Flash ECC レジスタ（Flash_ECCR）

アドレス･オフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

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

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

ビット 31 **ECCD**：ECC 検出

2 つの ECC エラーが検出されたときに、ハードウェアによってセットされます（ECCC/ECCD が、あらかじめクリアされている場合のみ）。このビットがセットされた場合、NMI が生成されます。

1 を書き込むとクリアされます。

ビット 30 **ECCC**：ECC 訂正

1 つの ECC エラーが検出されて訂正されたときに、ハードウェアによってセットされます（ECCC/ECCC2/ECCD/ECCD2 が以前にクリアされている場合のみ）。ECCCIE がセットされると、割込みが生成されます。

1 を書き込むとクリアされます。

ビット 29:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ECCCIE**：ECC 訂正割込み有効化

0：ECCC 割込み無効

1：ECCC 割込みは有効です。

このビットは、Flash_ECCR レジスタの ECCC ビットがセットされると割込み生成を有効にします。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **SYSF_ECC**：システムFlash ECC 失敗

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

ビット 21:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **ADDR_ECC**：ECC 失敗アドレス

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

5.7.8 Flashオプションレジスタ（Flash_OPTR）

アドレス･オフセット：0x20

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

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IRHEN	NRST_MODE [1:0]		n BOOT0	nSW BOOT0	CCM SRAM_ RST	SRAM _PE	nBOOT1	Res.	Res.	Res.	WWDG_ SW	IWGD_ STDBY	IWDG_ STOP	IWDG_ SW
	rw	rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	nRST_S HDW	nRST_S TDBY	nRST_S TOP	Res.	BOR_LEV[2:0]			RDP[7:0]							
	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **IRHEN**：内部リセット･ホルダー･イネーブル･ビット

0：内部リセットは、NRST ピン上の単純なパルスとして伝播します。

1：内部リセットは、ロー･レベルと見なされるまで NRST ピンをローになるように駆動します。

ビット 29:28 **NRST_MODE[1:0]**：

00：予約済みです。

01：リセット入力専用：NRST ピンが Low レベルになるとシステム･リセットを生成します。内部リセットが NRST ピンに伝播することはありません。

10：GPIO：標準 GPIO パッドの機能を使用します。内部リセットのみ可能です。

11：双方向リセット：NRST ピンが、リセット入出力モードで設定されます（レガシーモード）。

ビット 27 **nBOOT0**：nBOOT0 オプションビット

0：nBOOT0 = 0

1：nBOOT0 = 1

ビット 26 **nSWBOOT0**：ソフトウェア BOOT0

0：オプション･ビット nBOOT0 から取得した BOOT0 です。

1：PB8/BOOT0 ピンから取得した BOOT0 です。

ビット 25 **CCMSRAM_RST**：システムリセット時の CCM SRAM 消去

0：システム･リセット時に CCM SRAM が消去されます。

1：システム･リセット時に CCM SRAM が消去されません。

ビット 24 **SRAM_PE**：SRAM1 と CCM SRAM のパリティチェックイネーブル

0：SRAM1 と CCM SRAM のパリティ･チェックが有効です。

1：SRAM1 と CCM SRAM のパリティ･チェックが無効です。

ビット 23 **nBOOT1**：ブート設定

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

ビット：22:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **WWDG_SW**：ウィンドウ型ウォッチドッグの選択

0：ハードウェアによるウィンドウ型ウォッチドッグです。

1：ソフトウェアによるウィンドウ型ウォッチドッグです。

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

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

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

ビット 15 予約済みであり、リセット値に保持する必要があります。

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

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

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

ビット 11 予約済みであり、リセット値に保持する必要があります。

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

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

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

5.7.9 Flash PCROP1 開始アドレス･レジスタ（FLASH_PCROP1SR）

アドレス･オフセット：0x24

リセット値：0xFFFF XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

アクセス：ノーウェイトステート（進行中のFlashメモリ操作がない場合）、ワードアクセス

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_STRT[14:0]**：PCROP 領域開始オフセット
PCROP1_STRT は、PCROP 領域の最初のダブルワードを含んでいます。

5.7.10 Flash PCROP1 終了アドレス･レジスタ（FLASH_PCROP1ER）

アドレス･オフセット：0x28

リセット値：0xX000 XXXX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

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

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

ビット 31 **PCROP_RDP**：RDP レベルが下がると保存される PCROP 領域
このビットはセット専用です。RDP レベル の 1 から 0 への変更による完全全体消去の後にリセットされます。
0：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されません。
1：RDP レベルが 1 から 0 に下がった場合に PCROP 領域が消去されます（完全全体消去）。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PCROP1_END[14:0]**：PCROP 領域終了オフセット
PCROP1_END は、PCROP 領域の最後のダブルワードを含んでいます。

5.7.11 Flash WRP 領域 A アドレスレジスタ（Flash_WRP1AR）

アドレス･オフセット：0x2C

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [5:0]					
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **WRP1A_END[5:0]**：WRP 最初の領域「A」終了オフセット
WRP1A_END は、WRP の最初の領域の最終ページを含んでいます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WRP1A_STRT[5:0]**：WRP 最初の領域「A」開始オフセット
WRP1A_STRT は、WRP の最初の領域の最初のページを含んでいます。

5.7.12 Flash WRP 領域 B アドレスレジスタ（Flash_WRP1BR）

アドレス･オフセット：0x30

リセット値：0x00XX 00XX

レジスタ･ビットには、OBL 時に Flash メモリからの値が読み込まれます。

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

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END [5:0]					
										rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT [5:0]					
										rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:16 **WRP1B_END[5:0]**：WRP 2 番目の領域「B」終了オフセット
WRP1B_END は、WRP の 2 番目の領域の最終ページを含んでいます。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **WRP1B_STRT[5:0]**：WRP 2 番目の領域「B」開始オフセット
WRP1B_STRT は、WRP の 2 番目の領域の最終ページを含んでいます。

5.7.13 Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）

アドレス･オフセット：0x70

リセット値：0xFFFX FFXX

アクセス：ノーウェイトステート（進行中の Flash メモリ操作がない場合）、ワード、ハーフワード、バイト･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [6:0]						
									rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BOOT_LOCK**：ユーザ Flash 領域からのブートを強制するために使用
0：パッド/オプション･ビットの設定に基づいてブートします。
1：強制的にメイン Flash メモリからブートします。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **SEC_SIZE1[6:0]**：セキュリティ保護可能な領域に使用されているページ数がセットされます。
セキュリティ保護可能な領域は 0x0800 0000 から始まり、そのサイズは SEC_SIZE1 * ページ･サイズとなります。このフィールドはレベル 0 でのみ変更可能です。
レベル 1 で変更を試みるとサイレントに失敗し、レジスタ値は変更されません。

5.7.14 Flash レジスタマップ

表 37. Flash インタフェース - レジスタ・マップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FLASH_ACR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_SWEN	Res.	Res.	Res.	SLEEP_PD	RUN_PD	DCRST	ICRST	DCEN	ICEN	PRFTEN	Res.	Res.	Res.	Res.	LATENCY [3:0]			
	リセット値														1				0	0	0	0	1	1	0					0	0	0	1
0x04	FLASH_ PDKEYR	PDKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FLASH_KEYR	KEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FLASH_OPT KEYR	OPTKEYR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FLASH_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BSY	OPTVERR	RDERR	Res.	Res.	Res.	Res.	FASTERR	MISERR	PGSERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.	OPERR	EOP
	リセット値																0	0	0					0	0	0	0	0	0	0		0	0
0x14	FLASH_CR	LOCK	OPTLOCK	Res.	SEC_PROT1	OBL_LAUNCH	RDERRIE	ERRIE	EOPIE	Res.	Res.	Res.	Res.	Res.	FSTPG	OPTSTRT	STRT	Res.	Res.	Res.	Res.	Res.	Res.	PNB[6:0]							MER1	PER	PG
	リセット値	1	1		1	0	0	0	0						0	0	0							0	0	0	0	0	0	0	0	0	0
0x18	FLASH_ECCR	ECCD	ECCC	Res.	Res.	Res.	Res.	Res.	ECCCIE	Res.	SYSF_ECC	Res.	Res.	Res.	ADDR_ECC[18:0]																		
	リセット値	0	0						0		0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	FLASH_OPTR	Res.	Res.	Res.	Res.	nBOOT0	nSWBOOT0	CCMSRAM_RST	SRAM_PE	nBOOT1	Res.	Res.	Res.	WWDG_SW	IWDG_STBY	IWDG_STOP	IWDG_SW	Res.	nRST_SHDW	nRST_STDBY	nRST_STOP	Res.	BOR_ LEV[2:0]			RDP[7:0]							
	リセット値					X	X	X	X	X				X	X	X	X		X	X	X		X	X	X	X	X	X	X	X	X	X	X
0x24	FLASH_ PCROP1SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_STRT [14:0]														
	リセット値																		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x28	FLASH_ PCROP1ER	PCROP_RDP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PCROP1_END [14:0]														
	リセット値	x																	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
0x2C	FLASH_ WRP1AR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_END [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1A_STRT [6:0]						
	リセット値										X	X	X	X	X	X	X										X	X	X	X	X	X	X
0x30	FLASH_ WRP1BR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_END [6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WRP1B_STRT [6:0]						
	リセット値										X	X	X	X	X	X	X										X	X	X	X	X	X	X

表 37. Flash インタフェース - レジスタ･マップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x70	FLASH_ SEC1R	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BOOT_LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEC_SIZE1 [6:0]						
	リセット値																X										X	X	X	X	X	X	X

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

6 電源制御（PWR）

6.1 電源

STM32G4 シリーズ デバイスには、1.71～3.6 V の動作電源電圧（V_{DD}）が必要です。アナログペリフェラルには、独立した電源ドメインV_{DDA}を介して電源が供給されます。

- V_{DD} = 1.71～3.6 V

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

- V_{DDA} = 1.62 V（ADC / COMP）/ 1.71 V（DAC 1MSPS / DAC 15MSPS）/ 2 V（OPAMP）/ 2.4 V（VREFBUF）

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

  パワーアップとパワーダウンの間は、次の電源シーケンスが必要です。

  – V_{DD} が 1 V 未満の場合、V_{DDA} 電源は V_{DD} + 300 mV 未満にとどまる必要があります。

  – V_{DD} が 1 V を上回る場合、すべての電源は独立となります。

  パワーダウン･フェーズの間、マイクロコントローラに供給されるエネルギーが 1 mJ 未満に留まっている場合に限り、一時的に V_{DD} が他の電源よりも低い電圧となっても構いません。これによって、パワーダウン過渡フェーズの間に、外部デカップリングコンデンサを異なる時定数で放電させることが可能となります。

- V_{BAT} = 1.55～3.6 V

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

- VREF-、VREF+

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

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

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

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

  内部電圧リファレンスバッファは次の 3 つの出力電圧をサポートしており、VREFBUF_CSR レジスタの VRS ビットで設定します。

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

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

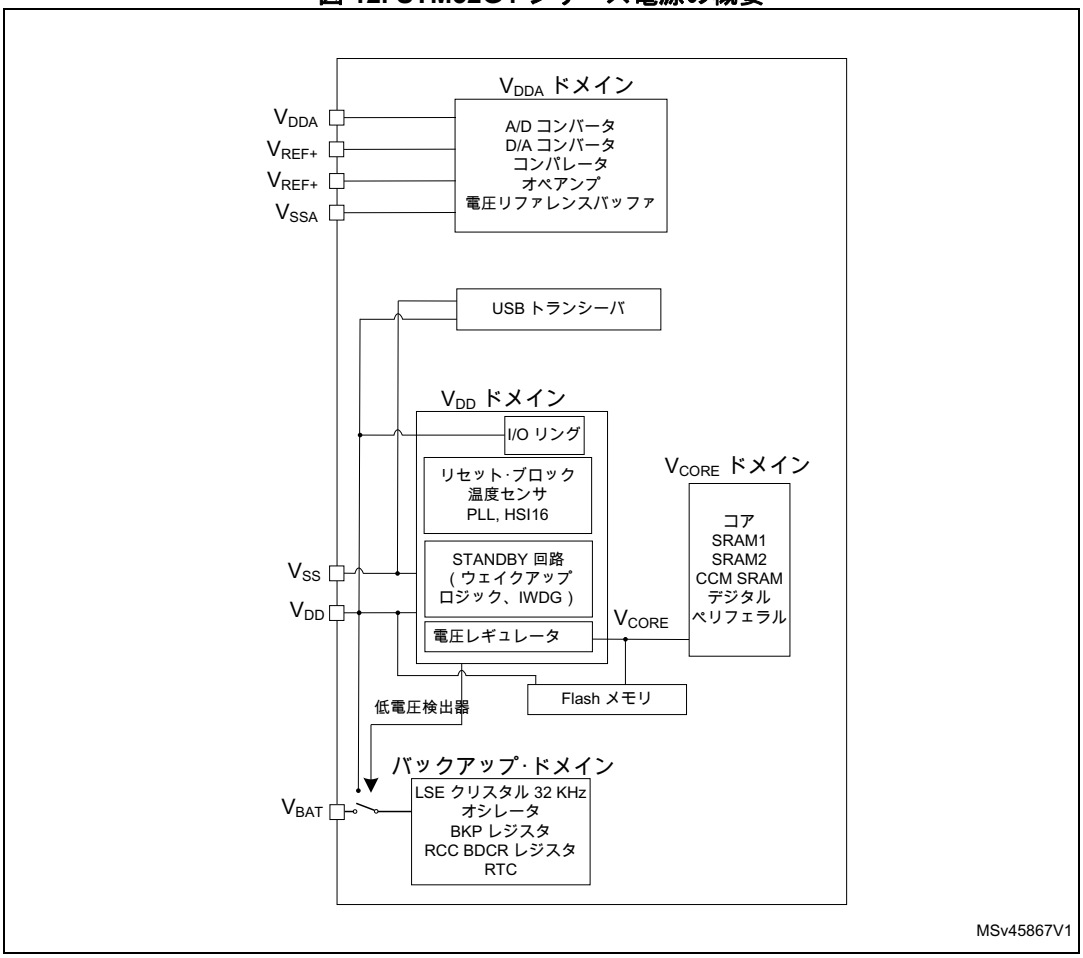
  – V_{REF+} 約 2.9 V。これには、3.135 V 以上の V_{DDA} が必要です。

  V_{REF+} ピンは、すべてのパッケージで使用できるわけではありません。このピンを使用できないパッケージでは、V_{DDA} にボンディングされています。V_{REF+} がパッケージ内で V_{DDA} とダブルボンディングされている場合、内部電圧リファレンス･バッファ（VREFBUF）は使用できず、無効にしておく必要があります（パッケージのピン配置の説明については、関連するデバイス･データシートを参照してください）。

  V_{REF-} は内部的に V_{SSA} にダブルボンディングされています。

内部のデジタル電源 V_{CORE} を供給するために、リニア電圧レギュレータが搭載されています。V_{CORE} は、デジタルペリフェラル、SRAM1、SRAM2 および CCM SRAM の電源です。Flash メモリは V_{CORE} と V_{DD} によって給電されます。

図 12. STM32G4 シリーズ電源の概要



6.1.1 独立したアナログペリフェラルの電源

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

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

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

V_{DDA} 電源は、ペリフェラル電圧モニタで監視され、閾値と比較できます。詳細については、セクション 6.2.3 : ペリフェラル電圧モニタ（PVM）を参照してください。

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

ADC および DAC の基準電圧

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

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

内部バッファ付き電圧リファレンス（VREFBUF）は、セクション 23.4.1：VREFBUF 制御およびステータスレジスタ（VREFBUF_CSR）の ENVR ビットをセットすると有効になります。内部バッファ付き電圧リファレンス（VREFBUF）は、VRS[1:0] ビットの設定に応じて、2.048 V、2.5 V または 2.9 V にセットされます。内部バッファ付き電圧リファレンスは、V_{REF+} ピンから外部コンポーネントに電圧を供給することもできます。詳細については、製品データシートおよびセクション 23：電圧基準バッファ（VREFBUF）を参照してください。

6.1.2 USB トランシーバ電源

USB トランシーバの電源は、V_{DD} 電源ピンから供給されます。USB 用の V_{DD} の範囲は 3.0 V から 3.6 V です。

6.1.3 バッテリバックアップドメイン

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

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

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

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

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

- PC13、PC14、および PC15 は、GPIO ピンとして使用可能
- PC13、PC14、および PC15 は、RTC または LSE で設定可能（1511 ページのセクション 35.3：RTC の機能説明を参照）
- タンパピンとして RTC で設定する場合、PA0/RTC_TAMP2 および PE6/RTC_TAMP3

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

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

- PC13、PC14、および PC15 は、RTC または LSE でのみ制御可能（1511 ページのセクション 35.3：RTC の機能説明を参照）
- タンパピンとして RTC で設定する場合、PA0/RTC_TAMP2 および PE6/RTC_TAMP3

バックアップドメインアクセス

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

1. セクション 7.4.17:APB1 ペリフェラル･クロック有効化レジスタ 1（RCC_APB1ENR1）の PWREN ビットをセットして、電源インタフェースクロックを有効にします。
2. 電源制御レジスタ 1（PWR_CR1）の DBP ビットをセットして、バックアップドメインへのアクセスを有効にします。
3. RTC ドメイン制御レジスタ（RCC_BDCR）の RTC クロックソースを選択します。
4. RTC ドメイン制御レジスタ（RCC_BDCR）の RTCEN [15] ビットをセットすることで、RTC クロックを有効にします。

VBAT バッテリ充電

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

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

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

6.1.4 電圧レギュレータ

STANDBY 回路およびバックアップドメイン以外のすべてのデジタル回路に電圧を供給する 2 つのリニア電圧レギュレータが組み込まれています。メインレギュレータの出力電圧（V_{CORE}）は、ソフトウェアで 2 つの異なる電源範囲（レンジ 1 およびレンジ 2）にプログラムして、システムの最大動作周波数に応じて最適化できます（セクション 7.2.8：クロックソースの周波数と電圧スケーリングおよびセクション 3.3.3：読出しアクセスの遅延を参照してください）。

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

- RUN モード、SLEEP モードおよび STOP 0 モードでは、両レギュレータが有効になり、メインレギュレータ（MR）は V_{CORE} ドメイン（コア、メモリ、デジタルペリフェラル）にフル電力を供給します。
- 低消費電力 RUN モードおよび低消費電力 SLEEP モードでは、メインレギュレータはオフ状態で、低消費電力レギュレータ（LPR）が V_{CORE} ドメインに低消費電力を供給して、レジスタ、SRAM1、SRAM2 および CCM SRAM の内容を保持します。
- STOP 1 モードでは、メインレギュレータはオフになり、低消費電力レギュレータ（LPR）はレジスタと SRAM1、SRAM2 および CCM SRAM の内容を保持したまま、V_{CORE} ドメインに低消費電力を供給します。

- SRAM2 の内容を保持した STANDBY モード（RRS ビットが PWR_CR3 レジスタでセットされている）では、メインレギュレータ（MR）はオフになり、低消費電力レギュレータ（LPR）は SRAM2 にのみ電源を供給します。コア、デジタルペリフェラル（STANDBY 回路およびバックアップドメイン以外）、SRAM1 および CCM SRAM の電源がオフになります。
- STANDBY モードでは、両レギュレータの電源がオフになります。STANDBY 回路とバックアップドメインを除き、レジスタ、SRAM1、SRAM2 および CCM SRAM の内容は失われます。
- SHUTDOWN モードでは、両レギュレータの電源がオフになります。SHUTDOWN モードが終了すると、パワーオンリセットが生成されます。結果として、バックアップドメインを除き、レジスタ、SRAM1、SRAM2 および CCM SRAM の内容は失われます。

6.1.5 ダイナミック電圧スケーリングの管理

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

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

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

- レンジ 1：ハイパフォーマンスレンジ

レンジ 1 では、メインレギュレータは PWR_CR5 レジスタの R1MODE ビットに従って 2 つのモードで動作します。

- メインレギュレータレンジ 1 ノーマルモード：標準電圧 1.2 V で出力します。システムクロック周波数は最大 150 MHz まで可能です。読出しアクセスのFlashアクセス時間は最小で、書込みおよび消去操作が可能です。
- メインレギュレータレンジ 1 ブーストモード：標準電圧 1.28 V で出力します。システムクロック周波数は最大 170 MHz まで可能です。読出しアクセスのFlashアクセス時間は最小で、書込みおよび消去操作が可能です。消費電力を最適化するには、システムクロック周波数が 150 MHz を超える場合にレンジ 1 ブーストモードを選択することをお勧めします。表 38 を参照してください。

表 38. レンジ 1 ブーストモード設定

システム周波数	SYSCLK ≤ 150 MHz	SYSCLK ≤ 170 MHz
R1MODE ビット設定	1	0

- レンジ 2：低消費電力レンジ

メインレギュレータは標準電圧 1.0 V で出力します。システムクロック周波数は最大 26 MHz まで可能です。読出しアクセスの Flash アクセス時間はレンジ 1 と比べて増加します。書込みおよび消去操作を行うことはできません。

電圧スケーリングは、セクション 6.4.1：電源制御レジスタ 1（PWR_CR1）レジスタの VOS ビットで選択します。

レンジ 1（ノーマル／ブースト）からレンジ 2 に移行するには、次の手順を実行します。

1. レンジ 1 ブーストモードからレンジ 2 に切り替える場合、低いシステム周波数に切り替える前に最低でも 1 us の間、AHB プリスケーラを使用してシステムクロックを 2 分周し、その後で AHB プリスケーラを再設定する必要があります。
2. システム周波数を 26 MHz 未満の値に下げます。
3. レンジ 2 の新しい周波数ターゲット（Flash_ACR の LATENCY ビット）に応じてウェイトステートの数を調整します。
4. PWR_CR1 レジスタの VOS ビットに「10」をプログラムします。

レンジ 2 からレンジ 1（ノーマル／ブーストモード）に移行するには、次の手順を実行します。

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

- システム周波数が 26 MHz を超え、150 MHz 以下の場合
  - PWR_CR5 レジスタの R1MODE ビットをセットして、レンジ 1 ノーマルモードを選択します。
  - 新しいシステム周波数に合った PLL への設定および切替えを行います。
- システム周波数 SYSCLK が 150 MHz 超える場合
  - 高いシステム周波数に切り替える前に、AHB プリスケーラを使用してシステムクロックを 2 分周する必要があります。
  - PWR_CR5 レジスタの R1MODE ビットをクリアして、レンジ 1 ブーストモードを選択します。
  - 新しいシステム周波数に合った PLL への設定および切替えを行います。
  - 最低でも 1 us の間待ってから、必要な HCLK クロック周波数を得るために AHB プリスケーラを再設定します。

レンジ 1 ノーマルモードからレンジ 1 ブーストモードに移行するには、次の手順を実行します。

1. 高いシステム周波数に切り替える前に、AHB プリスケーラを使用してシステムクロックを 2 分周する必要があります。
2. PWR_CR5 レジスタの R1MODE ビットをクリアします。
3. レンジ 1 ブーストモードの新しい周波数ターゲットに応じてウェイトステートの数を調整します。
4. 新しいシステム周波数への設定および切替えを行います。
5. 最低でも 1 us の間待ってから、必要な HCLK クロック周波数を得るために AHB プリスケーラを再設定します。

レンジ 1 ブースト･モードからレンジ 1 ノーマル･モードに移行するには、次の手順を実行します。

1. PWR_CR5 レジスタの R1MODE ビットをセットします。
2. レンジ 1 デフォルトモードの新しい周波数ターゲットに応じてウェイトステートの数を調整します。
3. 新しいシステム周波数への設定および切替えを行います。

6.2 電源供給スーパバイザ

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

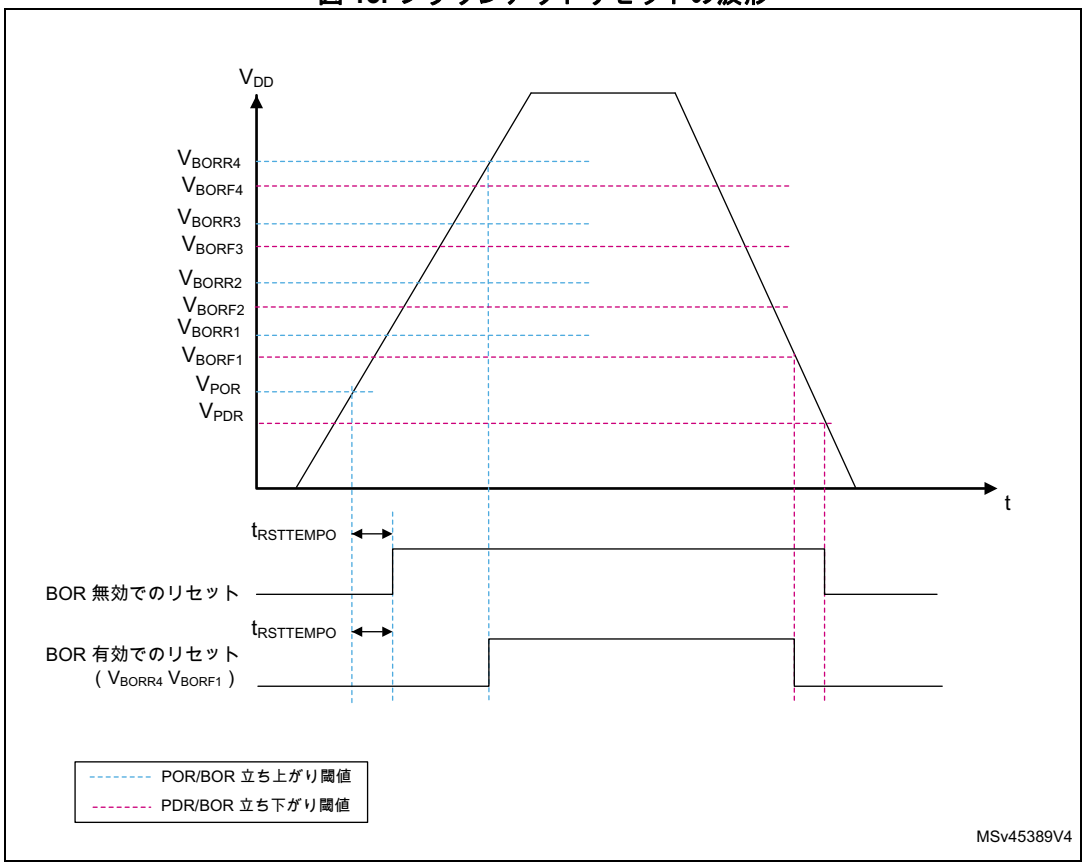
このデバイスは、パワーオン･リセット（POR）/パワーダウン･リセット（PDR）に加え、ブラウンアウト･リセット（BOR）回路を内蔵しています。BOR は、SHUTDOWN モードを除くすべての電力モードでアクティブで無効にできません。

BOR の閾値はオプションバイトで 5 種類から選択できます。

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

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

図 13. ブラウンアウトリセットの波形



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

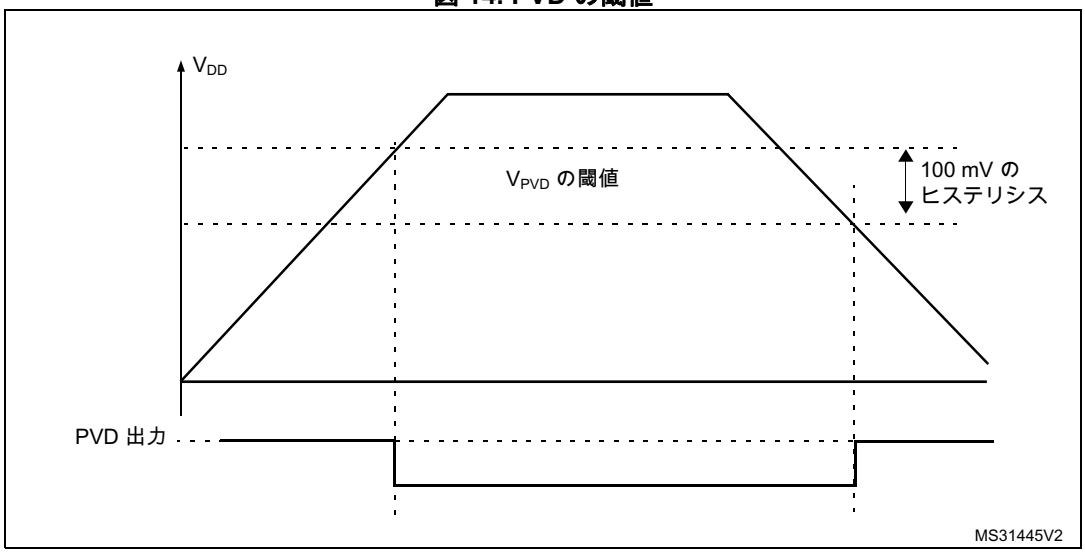
6.2.2 プログラム可能な電圧検出器（PVD）

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

PVD は PVDE ビットをセットすることで有効になります。

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

図 14. PVD の閾値



6.2.3 ペリフェラル電圧モニタ（PVM）

V_{DD} のみすべてのシステム関連機能に必要な唯一の電源であるため、デフォルトで監視されます。V_{DDA} は V_{DD} から独立しており、2 つのペリフェラル電圧モニタ（PVM）で監視できます。

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

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

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

表 39. PVM の機能

PVM	電源	PVM 閾値	EXTI ライン
PVM1	V_{DDA}	V_{PVM1}（約 1.65 V）	40
PVM2	V_{DDA}	V_{PVM2}（約 1.8 V）	41

独立したアナログ電源 V_{DDA} はデフォルトでは存在するとはみなされず、論理的および電気的分離を適用して、この専用電源が供給するペリフェラルからの情報を無視します。

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

6.3 低消費電力モード

デフォルトでは、マイクロコントローラは、システムリセットまたは電源リセット後は RUN モードです。
外部イベント待ちなど、CPU を動作させつづける必要がない場合の省電力化のために、いくつかの低消費電力モードが用意されています。消費電力の節減、スタートアップ時間の短縮、使用可能とするウェイクアップ·ソースなどの要因間のトレードオフを最適化するためのモード選択はユーザに委ねられています。

このデバイスは、次の 7 つの低消費電力モードを備えています。

- SLEEP モード：CPU クロックがオフとなり、NVIC や SysTick などの FPU 搭載 Cortex®-M4 コアペリフェラルを含むすべてのペリフェラルが実行でき、割込みやイベントが発生したときに CPU をウェイクアップさせます。セクション 6.3.4：SLEEP モードを参照してください。
- 低消費電力 RUN モード：このモードは、CPU クロック周波数が 2 MHz 未満に下がったときに設定できます。コードは SRAM またはFlashメモリから実行されます。レギュレータは低消費電力モードで、レギュレータの動作電流を最小限にします。セクション 6.3.2：低消費電力 RUN モード（LP RUN）を参照してください。
- 低消費電力 SLEEP モード：このモードは、低消費電力 RUN モードから移行します。FPU 搭載 Cortex®-M4 はオフとなります。セクション 6.3.5：低消費電力 SLEEP モード（LP SLEEP）を参照してください。
- STOP 0 および STOP 1 モード：SRAM とすべてのレジスタの内容は保持されます。V_{CORE} ドメインのすべてのクロックが停止し、PLL、HSI16、および HSE が無効になります。LSI および LSE を動作したままの状態にできます。

  RTC および TAMP をアクティブなままにできます（RTC ありの STOP モード、RTC なしの STOP モード）。

  ウェイクアップ機能付きペリフェラルは、STOP モード中に HSI16 RC を有効にして、ウェイクアップ条件を検出できます。

  STOP 0 モードでは、メインレギュレータがオンのままになり、最速のウェイクアップ時間を提供しますが、より高い消費電力になります。アクティブなペリフェラルおよびウェイクアップソースは、STOP 1 モードと同じです。

  STOP 0 または STOP 1 モードが終了すると、システムクロックは HSI16 クロックになります。デバイスが低消費電力 RUN モードでウェイクアップするように設定されている場合、2 MHz 以下の周波数を提供するには、STOP モードに入る前に RCC_CFGR レジスタの HPRE ビットを設定する必要があります。

  STOP 0 モードの詳細については、セクション 6.3.6：STOP 0 モードを参照してください。

- STANDBY モード：V_{CORE} ドメインの電源がオフになります。ただし、SRAM の内容は保持できます。
  - PWR_CR3 レジスタの RRS ビットがセットされたときの SRAM2 を保持した STANDBY モード。この場合、SRAM2 は低消費電力レギュレータによって電源供給されます。
  - PWR_CR3 レジスタの RRS ビットがクリアされたときの STANDBY モード。この場合、メインレギュレータおよび低消費電力レギュレータの電源がオフになります。

  V_{CORE} ドメインのすべてのクロックが停止し、PLL、HSI16、および HSE オシレータが無効になります。LSI および LSE を動作したままの状態にできます。

  RTC をアクティブなままにできます（RTC ありの STANDBY モード、RTC なしの STANDBY モード）。

  STANDBY モードが終了すると、システムクロックは HSI16 オシレータクロックになります。

  セクション 6.3.8：STANDBY モード を参照してください。
- SHUTDOWN モード：V_{CORE} ドメインの電源がオフになります。V_{CORE} ドメインのすべてのクロックが停止し、PLL、HSI16、LSI、および HSE が無効になります。LSE を動作したままの状態にできます。SHUTDOWN モードが終了すると、システムクロックは HSI16 オシレータクロックになります。このモードでは、電源電圧モニタが無効になり、電圧低下時の製品の動作は保証されません。セクション 6.3.9：SHUTDOWN モード を参照してください。

さらに、次の手段のいずれかによって RUN モードの消費電力を節減できます。

- システムクロックの低速化
- 使用しない APB および AHB ペリフェラルへのクロックのゲーティング

図 15. 低消費電力モードの可能な遷移

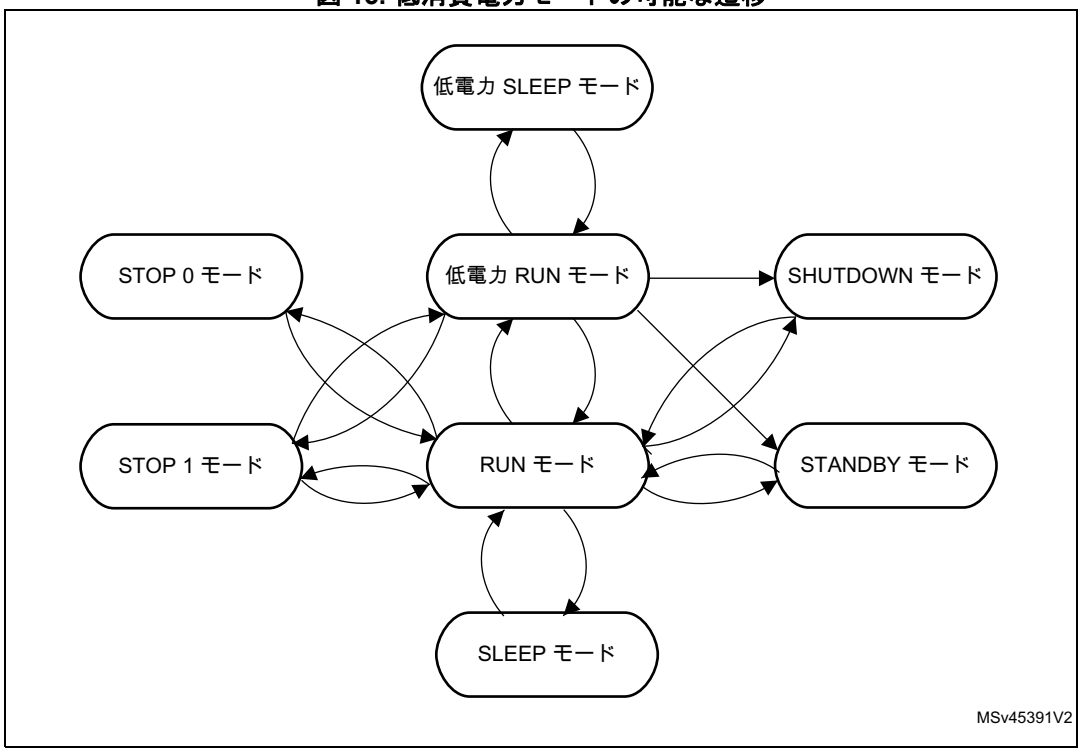


表 40. 低消費電力モードの概要

モード名	エントリ	ウェイクアップソース(1)	ウェイクアップシステムクロック	クロックへの影響	電圧レギュレータ	
					MR	LPR
SLEEP（Sleep-now または Sleep-on-exit）	WFI または ISRからの復帰	割込み	SLEEP モードへの移行前と同様	CPU クロックオフ 他のクロックおよびアナログクロックソースへの影響なし	オン	オン
	WFE	ウェイクアップ・イベント				
低消費電力 RUN	LPR ビットのセット	LPR ビットのクリア	低消費電力 RUN クロックと同様	なし	オフ	
低消費電力 SLEEP	LPR ビットのセット + WFI または ISR からのリターン	割込み	低消費電力 SLEEP モードへの移行前と同様	CPU クロックオフ 他のクロックおよびアナログクロックソースへの影響なし		
	LPR ビットのセット + WFE	ウェイクアップイベント				
STOP 0	LPMS=”000” + SLEEPDEEP ビット + WFI または ISRからのリターンまたは WFE	任意の EXTI ライン（EXTI レジスタで設定） 特定のペリフェラルイベント	HSI16	LSI および LSE を除くすべてのクロックをオフ	オン	
STOP 1	LPMS=”001” + SLEEPDEEP ビット + WFI または ISRからのリターンまたは WFE				オフ	
SRAM2 での STANDBY	LPMS=”011” + RRS ビットのセット + SLEEPDEEP ビット + WFI または ISRからの復帰または WFE	WKUP ピンのエッジ、RTC イベント、TAMP イベント、NRST ピンでの外部リセット、IWDG リセット				
STANDBY	LPMS=”011” + RRS ビットのクリア + SLEEPDEEP ビット + WFI または ISRからのリターンまたは WFE					オフ
SHUTDOWN	LPMS=”1--” + SLEEPDEEP ビット + WFI または ISRからのリターンまたは WFE	WKUP ピンのエッジ、RTC イベント、TAMP イベント、NRST ピンでの外部リセット		LSE を除くすべてのクロックオフ		

1. 表 41：動作モードに応じた機能を参照してください。

表 41. 動作モードに応じた機能(1)

ペリフェラル	RUN	SLEEP	低消費電力 RUN	低消費電力 SLEEP	STOP 0/1		STANDBY		SHUTDOW		VBAT
					-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
CPU	Y	-	Y	-	-	-	-	-	-	-	-
Flashメモリ	O(2)	O (2)	O (2)	O (2)	-	-	-	-	-	-	-
SRAM1	Y	Y(3)	Y	Y (3)	Y	-	-	-	-	-	-
SRAM2	Y	Y (3)	Y	Y (3)	Y	-	O(4)	-	-	-	-
CCM SRAM	Y	Y (3)	Y	Y (3)	Y	-	-	-	-	-	-
FSMC	O	O	O	O	-	-	-	-	-	-	-
QUADSPI	O	O	O	O	-	-	-	-	-	-	-
バックアップレジスタ	Y	Y	Y	Y	Y	-	Y	-	Y	-	Y
ブラウンアウトリセット（BOR）	Y	Y	Y	Y	Y	Y	Y	Y	-	-	-
プログラム可能な電圧検出器（PVD）	O	O	O	O	O	O	-	-	-	-	-
ペリフェラル電圧モニタ（PVM）	O	O	O	O	O	O	-	-	-	-	-
DMA	O	O	O	O	-	-	-	-	-	-	-
オシレータ HSI16	O	O	O	O	(5)	-	-	-	-	-	-
オシレータ HSI48	O	O	-	-	-	-	-	-	-	-	-
高速外部（HSE）	O	O	O	O	-	-	-	-	-	-	-
低速内部（LSI）	O	O	O	O	O	-	O	-	-	-	-
低速外部（LSE）	O	O	O	O	O	-	O	-	O	-	O
クロック・セキュリティ・システム（CSS）	O	O	O	O	-	-	-	-	-	-	-
LSE のクロック・セキュリティ・システム	O	O	O	O	O	O	O	O	-	-	-
RTC／自動ウェイクアップ	O	O	O	O	O	O	O	O	O	O	O
RTC タンパピンの数	3	3	3	3	3	O	3	O	3	O	3
USB	O (8)	O (8)	-	-	-	O	-	-	-	-	-
USARTx（x = 1、2、3、4、5）	O	O	O	O	O (6)	O (6)	-	-	-	-	-
低消費電力 UART（LPUART1）	O	O	O	O	O (6)	O (6)	-	-	-	-	-
I2Cx（x = 1、2、3、4）	O	O	O	O	O (7)	O (7)	-	-	-	-	-

表 41. 動作モードに応じた機能(1)（続き）

ペリフェラル	RUN	SLEEP	低消費電力 RUN	低消費電力 SLEEP	STOP 0/1		STANDBY		SHUTDOW		VBAT
					-	ウェイクアップ機能	-	ウェイクアップ機能	-	ウェイクアップ機能	
SPIx（x = 1、2、3、4）	O	O	O	O	-	-	-	-	-	-	-
FDCANx（1、2、3）	O	O	O	O	-	-	-	-	-	-	-
SAI1	O	O	O	O	-	-	-	-	-	-	-
ADCx（x = 1、2、3、4、5）	O	O	O	O	-	-	-	-	-	-	-
DACx（x=1、2、3、4）	O	O	O	O	O	-	-	-	-	-	-
VREFBUF	O	O	O	O	O	-	-	-	-	-	-
OPAMPx（x = 1、2、3、4、5、6）	O	O	O	O	O	-	-	-	-	-	-
COMPx（x = 1、2、3、4、5、6、7）	O	O	O	O	O	O	-	-	-	-	-
温度センサ	O	O	O	O	-	-	-	-	-	-	-
タイマ（TIMx）	O	O	O	O	-	-	-	-	-	-	-
高分解能タイマ 1（HRTIM1）	O	O	O	O	-	-	-	-	-	-	-
低消費電力タイマ 1（LPTIM1）	O	O	O	O	O	O	-	-	-	-	-
独立型ウォッチドッグ（IWDG）	O	O	O	O	O	O	O	O	-	-	-
ウィンドウ型ウォッチドッグ（WWDG）	O	O	O	O	-	-	-	-	-	-	-
SysTick タイマ	O	O	O	O	-	-	-	-	-	-	-
乱数発生器（RNG）	O(8)	O (8)	-	-	-	-	-	-	-	-	-
AES ハードウェアアクセラレータ	O	O	O	O	-	-	-	-	-	-	-
CRC 計算ユニット	O	O	O	O	-	-	-	-	-	-	-
GPIO	O	O	O	O	O	O	(9)	ピン：5本 (10)	(11)	ピン：5本 (10)	-
フィルタ演算アクセラレータ（FMAC）	O	O	O	O	-	-	-	-	-	-	-
CORDIC コプロセッサ（CORDIC）	O	O	O	O	-	-	-	-	-	-	-

1. 凡例 : Y = はい（有効）O = 任意(デフォルトでは無効だが、ソフトウェアで有効化できる)。- = 使用不可、ウェイクアップはグレイでハイライト
2. Flashはパワーダウンモードで設定できます。デフォルトは、パワーダウンモードではありません。
3. SRAM クロックのゲートオン／オフ可能
4. PWR_CR3 レジスタの RRS ビットがセットされたとき、SRAM2 の内容は保持されます。

5. STOP モードからのウェイクアップ機能を持つ一部のペリフェラルは、HSI16 の有効化をリクエストできます。この場合、HSI16 はペリフェラルによってウェイクアップされ、HSI16 をリクエストしたペリフェラルだけにクロックを供給します。ペリフェラルが HSI16 を必要としなくなると、HSI16 は自動的にオフされます。
6. UART および LPUART の受信は STOP モードで機能し、START でのウェイクアップ割込み、アドレス一致または受信フレームイベントを生成します。
7. I2C アドレス検出は STOP モードで機能し、アドレス一致時にウェイクアップ割込みを生成します。
8. 電圧スケーリングレンジ 1 のみです。
9. I/O は、STANDBY モードで内部プルアップ、プルダウンまたはフローティングで設定できます。
10. STANDBY/SHUTDOWN モードからのウェイクアップ機能が付いた I/O は、PA0、PC13、PE6、PA2、PC5 です。
11. I/O は、SHUTDOWN モードで内部プルアップ、プルダウンまたはフローティングで設定できますが、SHUTDOWN モードを終了すると設定が失われます。

デバッグモード

デフォルトでは、デバッグ機能が使用されているときにアプリケーションがマイクロコントローラを STOP 0、STOP 1、STANDBY または SHUTDOWN モードにすると、デバッグ接続は失われます。これは、FPU 搭載 Cortex®-M4 コアにクロックが供給されなくなるためです。

ただし、DBGMCU_CR レジスタの設定ビットをセットすることによって、低消費電力モードを多用しているときでも、ソフトウェアのデバッグを行うことができます。詳細については、セクション 47.16.1：低電力モードのデバッグサポートを参照してください。

6.3.1 RUN モード

システムクロックの低速化

RUN モードでは、プリスケーラレジスタをプログラミングすることによって、システムクロック（SYSCLK、HCLK、PCLK）の速度を下げることができます。SLEEP モードに移行する前にペリフェラルの速度を下げるため、これらのプリスケーラを使用することもできます。

詳細については、セクション 7.4.3：クロック設定レジスタ（RCC_CFGR）を参照してください。

ペリフェラルクロックゲーティング

RUN モードでは、消費電力を低減するため、個々のペリフェラルとメモリへの HCLK および PCLK をいつでも停止することができます。

SLEEP モードで消費電力をさらに低減するため、WFI または WFE 命令を実行する前に、ペリフェラルクロックを停止することができます。

ペリフェラルクロックゲーティングは、RCC_AHBxENR および RCC_APBxENR レジスタで制御されます。

RCC_AHBxSMENR レジスタと RCC_APBxSMENR レジスタの対応ビットをリセットすることで、SLEEP モード時のペリフェラルクロックを自動的に停止させることができます。

6.3.2 低消費電力 RUN モード（LP RUN）

システムが RUN モードの場合に消費電力を削減するために、レギュレータを低消費電力モードで設定することができます。このモードでは、CPU 周波数は 2 MHz を超えてはいけません。

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

低消費電力 RUN モードにおける I/O の状態

低消費電力 RUN モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

低消費電力 RUN モードへの移行

低消費電力 RUN モードに移行するには、次の手順に従います。

1. オプション：SRAM にジャンプし、Flash アクセス制御レジスタ（FLASH_ACR）の RUN_PD ビットをセットして、Flashをパワーダウンさせます。
2. CPU クロック周波数を 2 MHz 未満に下げます。
3. PWR_CR1 レジスタの LPR ビットをセットして、レギュレータを強制的に低消費電力モードにします。

低消費電力 RUN モードへの移行方法については、表 42：低消費電力 RUN を参照してください。

低消費電力 RUN モードの終了

低消費電力 RUN モードを終了するには、次の手順に従います。

1. PWR_CR1 レジスタの LPR ビットをクリアして、レギュレータを強制的にメインモードにします。
2. PWR_SR2 レジスタの REGLPF ビットがクリアされるまで待ちます。
3. CPU クロック周波数を上げます。

低消費電力 RUN モードの終了方法については、表 42：低消費電力 RUN を参照してください。

表 42. 低消費電力 RUN

低消費電力 RUN モード	説明
モードへの移行	CPU クロック周波数を 2 MHz 未満に下げます。 LPR = 1
モードの終了	LPR = 0 REGLPF = 0 になるまで待ちます。 CPU クロック周波数を上げます。
ウェイクアップ遅延時間	低消費電力モードからのレギュレータのウェイクアップ時間です。

6.3.3 低消費電力モード

低消費電力モードへの移行

低消費電力モードへの移行は、MCU が WFI（Wait For Interrupt）または WFE（Wait For Event）命令を実行することにより、または FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPONEXIT ビットが割込みサービスルーチン（ISR）からの復帰でセットされるときに実行されます。
WFI または WFE による低消費電力モードへの移行は、ペンディング状態の割込みやイベントがない場合にのみ実行されます。

低消費電力モードの終了

MCU は、低消費電力モードへの移行方法に応じて、低消費電力の SLEEP および STOP モードを終了します。

- WFI 命令または割込みサービスルーチン（ISR）からの復帰によって低消費電力モードに移行した場合、NVIC によって認識されたペリフェラル割込みであればどれでもデバイスをウェイクアップすることができます。

- WFE 命令によって低消費電力モードに移行した場合、MCU はイベントの発生直後に低消費電力モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。
  - NVIC IRQ 割込み。

    - FPU 搭載 Cortex®-M4 システム制御レジスタの SEVONPEND = 0 の場合。ペリフェラル制御レジスタおよび NVIC にて割込みを有効にすることで行います。MCU が WFE からリスタートするときには、ペリフェラル割込みペンディングビットと（NVIC 割込みクリアペンディングレジスタの）NVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    十分な優先度の NVIC 割込みのみが MCU をウェイクアップし、割込みを生成します。

    - FPU 搭載 Cortex®-M4 システム制御レジスタの SEVONPEND = 1 の場合。

    ペリフェラル制御レジスタ（および 任意でNVIC の割込み）にて割込みを有効にすることで行います。MCU が WFE からリスタートするときには、（NVIC 割込みクリアペンディングレジスタの）ペリフェラル割込みペンディングビットと、有効であればNVIC ペリフェラル IRQ チャネルペンディングビットをクリアする必要があります。

    すべての NVIC 割込み（無効化されているものも含む）によってマイクロコントローラをウェイクアップします。十分な優先度の有効化された NVIC 割込みのみがマイクロコントローラをウェイクアップし、割込みを生成します。
  - イベント

    EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、EXTI ペリフェラル割込みペンディングビットや NVIC IRQ チャネルペンディングをクリアする必要はありません。

    ペリフェラルの割込みフラグをクリアする必要があるかもしれません。

MCU は、外部リセット（NRST ピン）、IWDG リセット、有効な WKUPx ピンのうちの 1 つの立ち上がりエッジ、または RTC イベントのいずれかが発生すると、低消費電力の STANDBY モードおよび SHUTDOWN モードを終了します（ 図 529：RTC ブロック図を参照）。
STANDBY または SHUTDOWN モードからのウェイクアップ後、プログラム実行はリセット後と同様にリスタートされます（ブートピン信号のサンプリング、オプションバイトローディング、リセットベクタのフェッチなど）。

6.3.4 SLEEP モード

SLEEP モードにおける I/O の状態

SLEEP モードで、すべての I/O ピンは RUN モードと同じ状態を保持します。

SLEEP モードへの移行

SLEEP モードへの移行は、FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、セクション ：低消費電力モードへの移行に従って実行されます。

SLEEP モードへの移行方法の詳細については、表 43：SLEEP を参照してください。

SLEEP モードの終了

SLEEP モードは、セクションに従って終了します：低消費電力モードの終了。

SLEEP モードの終了方法の詳細については、表 43：SLEEP を参照してください。

表 43. SLEEP

Sleep-now モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 – ペンディング状態の割込み（WFI）やイベント（WFE）なし FPU 搭載 Cortex®-M4 システム制御レジスタを参照。
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 – ペンディング状態の割込みなし FPU 搭載 Cortex®-M4 システム制御レジスタを参照。
モードの終了	WFI または ISR からの復帰を使用して移行した場合 割込み：表 97：STM32G4 シリーズ ベクタテーブルを参照。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： ウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割込みイベント（NVIC で無効化されている場合も含む）：表 97：STM32G4 シリーズ ベクタテーブル またはウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。
ウェイクアップ遅延時間	なし

6.3.5 低消費電力 SLEEP モード（LP SLEEP）

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

低消費電力 SLEEP モードにおける I/O の状態

低消費電力 SLEEP モードで、I/O ピンはすべて RUN モードと同じ状態を保持します。

低消費電力 SLEEP モードへの移行

低消費電力 RUN モードから低消費電力 SLEEP モードへの移行は、FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがクリアされている場合に、セクション ：低消費電力モードへの移行に従って実行されます。

低消費電力 SLEEP モードへの移行方法の詳細については、表 44：低消費電力 SLEEP を参照してください。

低消費電力 SLEEP モードの終了

低消費電力 SLEEP モードは、セクション ：低消費電力モードの終了に従って終了します。割込みまたはウェイクアップイベントの発行によって低消費電力 SLEEP モードを終了する場合、MCU は低消費電力 RUN モードになります。

低消費電力 SLEEP モードの終了方法の詳細については、表 44：低消費電力 SLEEP を参照してください。

表 44. 低消費電力 SLEEP

低消費電力 sleep-now モード：	説明
モードへの移行	低消費電力 SLEEP モードは、低消費電力 RUN モードから移行します。 次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – SLEEPDEEP = 0 – ペンディング状態の割込み（WFI）やイベント（WFE）なし FPU 搭載 Cortex®-M4 システム制御レジスタを参照。
	低消費電力 SLEEP モードは、低消費電力 RUN モードから移行します。 次の条件下での割込みサービスルーチン（ISR）からの復帰時 – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 – ペンディング状態の割込みなし FPU 搭載 Cortex®-M4 システム制御レジスタを参照。
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割込み：表 97：STM32G4 シリーズ ベクタテーブルを参照。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： ウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割込み（NVIC で無効化されている場合も含む）：表 97：STM32G4 シリーズ ベクタテーブルを参照。 ウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。 低消費電力 SLEEP モードの終了後、MCU は低消費電力 RUN モードになります。
ウェイクアップ遅延時間	なし

6.3.6 STOP 0 モード

STOP 0 モードは、ペリフェラルクロックゲーティングと組み合わせた FPU 搭載 Cortex®-M4 のディープスリープ（deepsleep）モードに準拠しています。電圧レギュレータは、メインレギュレータモードで設定されます。STOP 0 モードでは、V_{CORE} ドメインのすべてのクロックが停止し、PLL、HSI16、および HSE オシレータが無効になります。ウェイクアップ機能を搭載したいくつかのペリフェラル（I2Cx（x = 1、2、3,4）、U(S)ARTx（x = 1、2〜5）、LPUARTは、フレームを受信するために HSI16 をオンにでき、ウェイクアップフレームではない場合はフレーム受信後に HSI16 をオフにできます。この場合、HSI16 クロックはフレームをリクエストしているペリフェラルに対してのみ伝達されます。

SRAM1、SRAM2、CCM SRAM およびレジスタの内容は保持されます。

BOR は常に STOP 0 モードで使用できます。V_{BOR0} を超える閾値が使用される場合、消費電力は増えます。

STOP 0 モードにおける I/O の状態

STOP 0 モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

STOP 0 モードへの移行

STOP 0 モードへの移行は、FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、セクション ：低消費電力モードへの移行に従って実行されます。

STOP 0 モードへの移行方法の詳細については、表 45：STOP 0 モードを参照してください。

Flashメモリがプログラミング中の場合、メモリアクセスが終了してから、STOP 0 モードに移行します。

APB ドメインにアクセス中の場合、APB アクセスが終了してから、STOP 0 モードに移行します。

STOP 0 モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）：IWDG は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。セクション 42.3：IWDG の機能説明を参照してください。
- リアルタイムクロック（RTC）：この設定はRTC ドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI）：この設定は制御／ステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE）：この設定はRTC ドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

次のとおり、いくつかのペリフェラルは、LSI や LSE で有効化されクロック供給されている場合や HSI16 クロックをリクエストしている場合、STOP 0 モードで使用できます。LPTIM1、I2Cx（x = 1、2、3、4）U(S)ARTx（x = 1、2～5）、LPUART

DACx（x = 1、2、3、4）、OPAMP、およびコンパレータは、STOP 0 モードで使用でき、PVM および PVD も同様です。これらが必要ではない場合、消費電力を低減するためにソフトウェアで無効にする必要があります。

ADCx（x = 1、2、3、4、5）、温度センサ、および VREFBUF バッファは、STOP 0 モードに移行する前に無効にしない限り、STOP 0 モード中に電力を消費します。

STOP 0 モードの終了

STOP 0 モードは、低消費電力モードへの移行に従って終了します。

STOP 0 モードの終了方法の詳細については、表 45：STOP 0 モードを参照してください。

割込みまたはウェイクアップイベントの発行によって STOP 0 モードを終了する場合、システムクロックとして HSI16 オシレータが選択されます。デバイスが低消費電力 RUN モードでウェイクアップするように設定されている場合、2 MHz 以下の周波数を提供するには、STOP 0モードに入る前に RCC_CFGR レジスタの HPRE ビットを設定する必要があります。

電圧レギュレータが低消費電力モードで動作している場合、HSI16 で STOP 0 モードからウェイクアップする際、更なる起動時間が必要になります。STOP 0 モードの間も内蔵レギュレータをオン状態に保つことによって、消費電力は増加しますが、起動時間は短縮されます。

STOP 0 モードを終了すると、電源制御レジスタ 1（PWR_CR1）で LPR ビットがセットされている場合、マイクロコントローラは RUN モード（PWR_CR1 の VOS ビットに応じてレンジ 1 またはレンジ 2）または低消費電力 RUN モードになります。

表 45. STOP 0 モード

STOP 0 モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割込み（WFI）やイベント（WFE）なし – LPMS = “000”（PWR_CR1）
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割込みなし – LPMS = “000”（PWR_CR1）
	注： STOP 0 モードに移行するには、すべての EXTI ラインのペンディングビット（ペンディングレジスタ 1（EXTI_PR1）内）、ウェイクアップ割込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP 0 モード移行手順が無視され、プログラムが実行され続けます。
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割込みモードに設定されている任意の EXTI ライン（対応する EXTI 割込みベクタが NVIC で有効になっている必要があります）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルになることがあります。表 97：STM32G4 シリーズ ベクタテーブル を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： イベントモードに設定されている任意の EXTI ライン。セクション 15.3.2：ウェイクアップイベント管理 を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割込みモードに設定されている任意の EXTI ライン（対応する EXTI 割込みベクタが NVIC で無効になっている場合も含む）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルになることがあります。表 97：STM32G4 シリーズ ベクタテーブルを参照してください。 ウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。
ウェイクアップ遅延時間	最長ウェイクアップ時間：HSI16 ウェイクアップ時間と STOP 0 モードからの Flash ウェイクアップ時間

6.3.7 STOP 1 モード

STOP 1 モードは、メインレギュレータがオフになっており、低消費電力レギュレータがオンになっている場合を除き、STOP 0 モードと同じです。STOP 1 モードは、RUN モードおよび低消費電力 RUN モードから移行できます。

STOP 1 モードの移行および終了方法の詳細については、表 46：STOP 1 モードを参照してください。

表 46. STOP 1 モード

STOP 1 モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割込み（WFI）やイベント（WFE）なし – LPMS = “001”（PWR_CR1）
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割込みなし – LPMS = “001”（PWR_CR1）
	注： STOP 1 モードに移行するには、すべての EXTI ラインのペンディングビット（セクション 15.5.6：ペンディングレジスタ 1（EXTI_PR1）内）、ウェイクアップ割込みを生成するペリフェラルフラグをクリアする必要があります。そうしないと、STOP 1 モード移行手順が無視され、プログラムが実行され続けます。
モードの終了	WFI または ISR からの復帰を使用して移行した場合： 割込みモードに設定されている任意の EXTI ライン（対応する EXTI 割込みベクタが NVIC で有効になっている必要があります）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルになることがあります。表 97：STM32G4 シリーズ ベクタテーブル を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 0 の場合： イベントモードに設定されている任意の EXTI ライン。セクション 15.3.2：ウェイクアップイベント管理 を参照してください。 WFE 命令を使用して移行し、SEVONPEND = 1 の場合： 割込みモードに設定されている任意の EXTI ライン（対応する EXTI 割込みベクタが NVIC で無効になっている場合も含む）。割込みソースは、ウェイクアップ機能を備えた外部割込みまたはペリフェラルになることがあります。表 97：STM32G4 シリーズ ベクタテーブルを参照してください。 ウェイクアップイベント：セクション 15.3.2：ウェイクアップイベント管理を参照してください。
ウェイクアップ遅延時間	最長ウェイクアップ時間：HSI16 ウェイクアップ時間と低消費電力モードからのレギュレータウェイクアップ時間 + STOP 1 モードからの Flash ウェイクアップ時間

6.3.8 STANDBY モード

STANDBY モードでは、BOR で消費電力を最も少なくできます。このモードは、電圧レギュレータを無効にした状態（SRAM2 の内容が保持されている場合を除く）の FPU 搭載 Cortex®-M4 のディープスリープ（deepsleep）モードに基づきます。PLL、HSI16、および HSE オシレータもオフになります。

バックアップドメインと STANDBY 回路のレジスタを除いて、SRAM1 とレジスタの内容は失われます（図 12 を参照）。PWR_CR3 レジスタの RRS ビットがセットされた場合、SRAM2 の内容を保持できます。この場合、低消費電力レギュレータがオンになり、SRAM2 に対してのみ電源供給されます。

BOR は常に STANDBY モードで使用できます。V_{BOR0} を超える閾値が使用される場合、消費電力は増えます。

STANDBY モードにおける I/O の状態

STANDBY モードでは、I/O をプルアップ（PWR_PUCRx レジスタ（x = A、B、C、D、E、F、G）を参照）、またはプルダウン（PWR_PDCRx レジスタ（x = A、B、C、D、E、F、G）を参照）に設定したり、アナログ状態のままに設定したりできます。

PC13 の RTC 出力は STANDBY モードで機能します。LSE に使用される PC14 と PC15 も機能します。5 つのウェイクアップピン（WKUPx、x=1、2...5）および 3 つの RTC タンパが使用できます。

STANDBY モードへの移行

STANDBY モードへの移行は、FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、セクション ：低消費電力モードへの移行に従って実行されます。

STANDBY モードへの移行方法の詳細については、表 47：STANDBY モードを参照してください。

STANDBY モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ（IWDG）：IWDG は、キーレジスタへの書込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。参照：セクション 42.3：IWDG の機能説明 、セクション 42：独立型ウォッチドッグ（IWDG）
- リアルタイムクロック（RTC）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。
- 内部 RC オシレータ（LSI）：この設定は、制御／ステータスレジスタ（RCC_CSR）の LSION ビットで行います。
- 外部 32.768 kHz オシレータ（LSE）：この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

STANDBY モードの終了

STANDBY モードは、セクション ：低消費電力モードへの移行に従って終了します。電源制御レジスタ 3（PWR_CR3）の SBF ステータスフラグは MCU が STANDBY モードにあったことを示します。電源制御レジスタ 3（PWR_CR3）を除くすべてのレジスタは、STANDBY モードからのウェイクアップ後にリセットされます。

STANDBY モードの終了方法の詳細については、表 47：STANDBY モードを参照してください。

表 47. STANDBY モード

STANDBY モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割込み（WFI）やイベント（WFE）なし – LPMS = “011”（PWR_CR1） – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXIT = 1 – ペンディング状態の割込みなし – LPMS = “011”（PWR_CR1）および – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア – 選択されたウェイクアップソース（RTC アラーム A、RTC アラーム B、RTC ウェイクアップ、タンパ、タイムスタンプフラグ）に対応する RTC フラグをクリア
モードの終了	WKUPx ピンのエッジ、RTC イベント、NRST ピンによる外部リセット、IWDG リセット、BOR リセット
ウェイクアップ遅延時間	リセットフェーズ

6.3.9 SHUTDOWN モード

SHUTDOWN モードでは、消費電力を最も少なくできます。このモードは、電圧レギュレータを無効にした状態のディープスリープ（deepsleep）モードに基づきます。結果として、V_{CORE} ドメインの電源がオフになります。PLL、HSI16、LSI および HSE オシレータもオフになります。

バックアップドメインのレジスタを除いて、SRAM1、SRAM2、CCM SRAM およびレジスタの内容は失われます。BOR は SHUTDOWN モードで使用できません。このモードで電源電圧監視はできないため、バックアップドメインへの切り替えはサポートされません。

SHUTDOWN モードにおける I/O の状態

SHUTDOWN モードでは、I/O をプルアップ（PWR_PUCRx レジスタ（x = A、B、C、D、E、F、G）を参照）、またはプルダウン（PWR_PDCRx レジスタ（x = A、B、C、D、E、F、G）を参照）に設定したり、アナログ状態のままに設定したりできます。ただし、パワーオンリセットにより SHUTDOWN モードが終了すると、この設定は失われます。

PC13 の RTC 出力は SHUTDOWN モードで機能します。LSE に使用される PC14 と PC15 も機能します。5 つのウェイクアップピン（WKUPx、x=1、2...5）および 3 つの RTC タンパが使用できます。

SHUTDOWN モードへの移行

SHUTDOWN モードへの移行は、FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットがセットされている場合に、低消費電力モードへの移行に従って実行されます。

SHUTDOWN モードへの移行方法の詳細については、表 48 : SHUTDOWN モードを参照してください。

SHUTDOWN モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- リアルタイムクロック（RTC）: この設定は、バックアップドメイン制御レジスタ（RCC_BDCR）の RTCEN ビットで行います。注意 : VDD がパワーダウンすると、RTC の内容は失われます。
- 外部 32.768 kHz オシレータ（LSE）: この設定は、バックアップ・ドメイン制御レジスタ（RCC_BDCR）の LSEON ビットで行います。

SHUTDOWN モードの終了

SHUTDOWN モードは、セクション ：低消費電力モードの終了に従って終了します。SHUTDOWN モードを終了すると、パワーオンリセットが発生します。SHUTDOWN モードからウェイクアップ後、すべてのレジスタ（バックアップドメインのものを除く）がリセットされます。

SHUTDOWN モードの終了方法の詳細については、表 48 : SHUTDOWN モードを参照してください。

表 48. SHUTDOWN モード

SHUTDOWN モード	説明
モードへの移行	次の条件下での WFI（Wait for Interrupt）または WFE（Wait for Event） – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – ペンディング状態の割込み（WFI）やイベント（WFE）なし – LPMS = “1XX”（PWR_CR1） – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア
	次の条件下での割込みサービスルーチン（ISR）からの復帰時 – FPU 搭載 Cortex®-M4 システム制御レジスタの SLEEPDEEP ビットをセット – SLEEPONEXT = 1 – ペンディング状態の割込みなし – LPMS = “1XX”（PWR_CR1）および – 電源ステータスレジスタ 1 （PWR_SR1）の WUFx ビットをクリア – 選択されたウェイクアップソース（RTC アラーム A、RTC アラーム B、RTC ウェイクアップ、タンパ、タイムスタンプフラグ）に対応する RTC フラグをクリア
モードの終了	WKUPx ピンのエッジ、RTC イベント、NRST ピンによる外部リセット
ウェイクアップ遅延時間	リセットフェーズ

6.3.10 低消費電力モードからの自動ウェイクアップ

RTC は、外部割込み（自動ウェイクアップモード）に頼ることなく、低消費電力モードから MCU をウェイクアップするために使用できます。RTC を、一定の時間間隔で STOP（0 または 1）モードや STANDBY モードからウェイクアップさせるためのプログラム可能なタイムベースとすることができます。この目的のため、RTC ドメイン制御レジスタ（RCC_BDCR）の RTCSEL[1:0] ビットをプログラムすることによって、次の 3 つの代替 RTC クロックソースのうちから 2 つを選択できます。

- 低消費電力 32.768 kHz 外部クリスタルオシレータ（LSE OSC）
  このクロックソースは、非常に少ない消費電力で高精度のタイムベースとなります。
- 低消費電力内部 RC オシレータ（LSI）
  このクロックソースには、32.768 kHz クリスタルのコストを節約できるという利点があります。この内部 RC オシレータは、最小限の消費電力を追加するように設計されています。

RTC アラームイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。

- EXTI ライン 17 を立ち上がりエッジを検知するように設定します。
- RTC が RTC アラームを生成するように設定します。

STANDBY モードからウェイクアップするには、EXTI ライン 17 の設定は必要ありません。

RTC ウェイクアップイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。

- EXTI ライン 20 を立ち上がりエッジを検知するように設定します。
- RTC が RTC アラームを生成するように設定します。

STANDBY モードからウェイクアップするには、EXTI ライン 20 の設定は必要ありません。

6.4 PWR レジスタ

ペリフェラル･レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

6.4.1 電源制御レジスタ 1（PWR_CR1）

アドレス･オフセット：0x00

リセット値：0x0000 0200

このレジスタは、STANDBY モードからのウェイクアップ後にリセットされます。

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

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

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **LPR**：低消費電力 RUN

このビットがセットされると、レギュレータはメインモード（MR）から低消費電力モード（LPR）に切り替わります。

ビット 13:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:9 **VOS[1:0]**：電圧スケーリング･レンジの選択

00：書込み不可（ハードウェアによって禁止されています）
01：レンジ 1
10：レンジ 2
11：書込み不可（ハードウェアによって禁止されています）

ビット 8 **DBP**：バックアップ･ドメイン書込み保護の無効化

リセット状態で、RTC およびバックアップレジスタは不要な書込みアクセスから保護されます。これらのレジスタへの書込みアクセスを可能にするには、このビットをセットする必要があります。
0：RTC およびバックアップ･レジスタへのアクセスは無効です。
1：RTC およびバックアップ･レジスタへのアクセスは有効です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **LPMS[2:0]**：低消費電力モード選択

これらのビットは、CPU がディープスリープ･モードへ移行するときに、移行する低消費電力モードを選択します。
000：STOP 0 モード
001：STOP 1 モード
010：予約済みです。
011：STANDBY モード
1xx：SHUTDOWN モード

注： STANDBY モードでは、PWR_CR3 の RRS ビットの設定に応じて、SRAM2 を保持できるかどうかが決まります。

6.4.2 電源制御レジスタ 2（PWR_CR2）

アドレス･オフセット：0x04

リセット値：0x0000 0000

このレジスタは、STANDBY モード終了時にリセットされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVME2	PVME1	Res.	Res.	PLS[2:0]			PVDE
								rw	rw			rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PVME2**：ペリフェラル電圧モニタ 2 有効化：V_{DDA} 対 DAC 1MSPS/DAC 15MSPS 最小電圧
0：PVM2（1.8 V を閾値として V_{DDA} を監視）を無効にします。
1：PVM2（1.8 V を閾値として V_{DDA} を監視）を有効にします。

ビット 6 **PVME1**：ペリフェラル電圧モニタ 1 有効化：V_{DDA} 対 ADC/COMP 最小電圧 1.62 V
0：PVM1（1.62V 閾値に対する V_{DDA} の監視）を無効にします。
1：PVM1（1.62V 閾値に対する V_{DDA} の監視）を有効にします。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:1 **PLS[2:0]**：プログラム可能な電圧検出器のレベル選択
これらのビットは、PVD 立ち下がり閾値を選択します。
000：V_{PVD0} PVD 閾値 0
001：V_{PVD1} PVD 閾値 1
010：V_{PVD2} PVD 閾値 2
011：V_{PVD3} PVD 閾値 3
100：V_{PVD4} PVD 閾値 4
101：V_{PVD5} PVD 閾値 5
110：V_{PVD6} PVD 閾値 6
111：外部入力アナログ電圧 PVD_IN（内部的に V_{REFINT} と比較されます）

注： これらのビットは、SYSCFG_CFGR2 レジスタの PVDL ビットがセットされているとき、書込み保護されます。保護は、システム･リセットによってのみリセットされます。

ビット 0 **PVDE**：プログラム可能な電圧検出器有効化
0：プログラム可能な電圧検出器を無効化します。
1：プログラム可能な電圧検出器を有効化します。

注： このビットは、SYSCFG_CFGR2 レジスタの PVDL ビットがセットされているとき、書込み保護されます。保護は、システム･リセットによってのみリセットされます。

6.4.3 電源制御レジスタ 3（PWR_CR3）

アドレス･オフセット：0x08

リセット値：0x0000 8000

このレジスタは、STANDBY モードの終了時、および RCC_APB1RSTR1 レジスタの PWRRST ビットではリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIWUL	UCPD1_DBDIS	UCPD1_STDBY	Res.	Res.	APC	Res.	RRS	Res.	Res.	Res.	EWUP5	EWUP4	EWUP3	EWUP2	EWUP1
rw	rw	rw			rw		rw				rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **EIWUL**：内部ウェイクアップ･ライン有効化

0：内部ウェイクアップ･ラインは無効です。

1：内部ウェイクアップ･ラインは有効です。

ビット 14 **UCPD1_DBDIS**：USB タイプ C およびパワーデリバリデッドバッテリの無効化

リセット終了後、USB タイプ C の「デッドバッテリ」動作が有効になり、CC1 ピンと CC2 ピンがプルダウンされた状態になる可能性があります。

どの場合でも、このビットを無効にすることをお勧めします。これにより、このプルダウンが発生しないようにするか、制御を UCPD1 に渡します（したがって、ビットを無効にする前に初期化が必要です）。

0：UCPD1_CC1 および UCPD1_CC2 ピンに対する USB タイプ C のデッドバッテリ･プルダウン動作を有効にします。

1：UCPD1_CC1 および UCPD1_CC2 ピンに対する USB タイプ C のデッドバッテリ･プルダウン動作を無効にします。

ビット 13 **UCPD1_STDBY :** UCPD1_STDBY USB タイプ C および電源供給の STANDBY モード

0：UCPD1 使用時、スタンバイ終了直後（およびいずれかの UCPD1 レジスタに書き込む前）に「0」を書き込みます。

1：UCPD1 使用時、スタンバイに入る直前に「1」を書き込みます。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **APC**：プルアップおよびプルダウン設定の適用

このビットがセットされると、PWR_PUCRx および PWR_PDCRx レジスタで定義された I/O プルアップおよびプルダウン設定が適用されます。このビットがクリアされると、PWR_PUCRx および PWR_PDCRx レジスタは I/O に適用されません。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **RRS**：STANDBY モードでの SRAM2 保持

0：SRAM2 が STANDBY モードで電源オフになります（SRAM2 の内容は失われます）。

1：SRAM2 が STANDBY モードで低消費電力レギュレータから給電されます（SRAM2 の内容は保持されます）。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **EWUP5**：ウェイクアップピン WKUP5 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP5 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP5 ビットでアクティブなエッジが設定されます。

ビット 3 **EWUP4**：ウェイクアップピン WKUP4 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP4 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP4 ビットでアクティブなエッジが設定されます。

ビット 2 **EWUP3**：ウェイクアップピン WKUP3 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP3 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP3 ビットでアクティブなエッジが設定されます。

ビット 1 **EWUP2**：ウェイクアップピン WKUP2 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP2 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP2 ビットでアクティブなエッジが設定されます。

ビット 0 **EWUP1**：ウェイクアップピン WKUP1 有効化

このビットをセットすると、立ち上がりまたは立ち下がりエッジが発生した際に、外部ウェイクアップピン WKUP1 が有効になり、STANDBY または SHUTDOWN イベントからウェイクアップします。PWR_CR4 レジスタの WP1 ビットでアクティブなエッジが設定されます。

6.4.4 電源制御レジスタ 4（PWR_CR4）

アドレス·オフセット：0x0C

リセット値：0x0000 0000

このレジスタは、STANDBY モードの終了時、および RCC_APB1RSTR1 レジスタの PWRRST ビットではリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	WP5	WP4	WP3	WP2	WP1
						rw	rw				rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **VBRS**：V_{BAT} バッテリ充電抵抗の選択

0：5 kΩ の抵抗で V_{BAT} を充電します。

1：1.5 kΩ の抵抗で V_{BAT} を充電します。

ビット 8 **VBE**：V_{BAT} バッテリ充電の有効化

0：V_{BAT} バッテリ充電は無効です。

1：V_{BAT} バッテリ充電は有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WP5**：ウェイクアップピン WKUP5 の極性

このビットは、外部ウェイクアップピン WKUP5 のイベントの検出に使用される極性を定義します。

0：High レベルで検出します（立ち上がりエッジ）。

1：Low レベルで検出します（立ち下がりエッジ）。

ビット 3 **WP4**：ウェイクアップピン WKUP4 の極性

このビットは、外部ウェイクアップピン WKUP4 のイベントの検出に使用される極性を定義します。

0：High レベルで検出します（立ち上がりエッジ）。

1：Low レベルで検出します（立ち下がりエッジ）。

ビット 2 **WP3**：ウェイクアップピン WKUP3 の極性

このビットは、外部ウェイクアップピン WKUP3 のイベントの検出に使用される極性を定義します。

0：High レベルで検出します（立ち上がりエッジ）。

1：Low レベルで検出します（立ち下がりエッジ）。

ビット 1 **WP2**：ウェイクアップピン WKUP2 の極性

このビットは、外部ウェイクアップピン WKUP2 のイベントの検出に使用される極性を定義します。

0：High レベルで検出します（立ち上がりエッジ）。

1：Low レベルで検出します（立ち下がりエッジ）。

ビット 0 **WP1**：ウェイクアップピン WKUP1 の極性

このビットは、外部ウェイクアップピン WKUP1 のイベントの検出に使用される極性を定義します。

0：High レベルで検出します（立ち上がりエッジ）。

1：Low レベルで検出します（立ち下がりエッジ）。

6.4.5 電源ステータスレジスタ 1（PWR_SR1）

アドレス･オフセット：0x10

リセット値：0x0000 0000

このレジスタは、STANDBY モードの終了時、および RCC_APB1RSTR1 レジスタの PWRRST ビットではリセットされません。

アクセス：このレジスタを読み出すには、通常の APB 読出しに対して、追加の APB サイクルが 2 つ必要です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUFI	Res.	Res.	Res.	Res.	Res.	Res.	SBF	Res.	Res.	Res.	WUF5	WUF4	WUF3	WUF2	WUF1
r							r				r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **WUFI**：内部ウェイクアップ･フラグ

このビットは、内部ウェイクアップラインにウェイクアップが検出されたときにセットされます。また、すべての内部ウェイクアップソースがクリアされたときにクリアされます。

ビット 14:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SBF**：STANDBY フラグ

このビットは、デバイスが STANDBY モードに移行したときにハードウェアによってセットされ、PWR_SCR レジスタの CSBF ビットをセットするか、パワーオンリセットによってクリアされます。システムリセットによってクリアされません。

0：デバイスが STANDBY モードに移行しませんでした。

1：デバイスが STANDBY モードに移行しました。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **WUF5**：ウェイクアップフラグ 5

このビットは、ウェイクアップピン WKUP5 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF5 ビットに「1」を書き込むことによってクリアされます。

ビット 3 **WUF4**：ウェイクアップフラグ 4

このビットは、ウェイクアップピン WKUP4 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF4 ビットに「1」を書き込むことによってクリアされます。

ビット 2 **WUF3**：ウェイクアップフラグ 3

このビットは、ウェイクアップピン WKUP3 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF3 ビットに「1」を書き込むことによってクリアされます。

ビット 1 **WUF2**：ウェイクアップフラグ 2

このビットは、ウェイクアップピン WKUP2 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF2 ビットに「1」を書き込むことによってクリアされます。

ビット 0 **WUF1**：ウェイクアップフラグ 1

このビットは、ウェイクアップピン WKUP1 でウェイクアップイベントが検出されたときにセットされます。PWR_SCR レジスタの CWUF1 ビットに「1」を書き込むことによってクリアされます。

6.4.6 電源ステータスレジスタ 2（PWR_SR2）

アドレス･オフセット：0x14

リセット値：0x0000 0000

このレジスタは、STANDBY/SHUTDOWN モード終了時に一部リセットされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PVMO2	PVMO1	Res.	Res.	PVDO	VOSF	REGLPF	REGLPS	FLASH_ RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r			r	r	r	r	r							

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PVMO2**：ペリフェラル電圧モニタの出力：V_{DDA} と 1.8 V

0：V_{DDA} 電圧が、PVM2 閾値（約 1.8 V）を上回っています。

1：V_{DDA} 電圧が、PVM2 閾値（約 1.8 V）を下回っています。

注： PVM2 が無効になったとき（PVME2 = 0）に PVMO2 がクリアされます。PVM2 を有効にすると、PVM2 ウェイクアップ時間後、PVM2 出力が有効になります。

ビット 14 **PVMO1**：ペリフェラル電圧モニタの出力：V_{DDA} と 1.62 V

0：V_{DDA} 電圧が、PVM1 閾値（約 1.62 V）を上回っています。

1：V_{DDA} 電圧が、PVM1 閾値（約 1.62 V）を下回っています。

注： PVM1 が無効になったとき（PVME1 = 0）に PVMO1 がクリアされます。PVM1 を有効にすると、PVM1 ウェイクアップ時間後、PVM1 出力が有効になります。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **PVDO**：プログラム可能な電圧検出器の出力

0：V_{DD} が、選択された PVD 閾値を上回っています。

1：V_{DD} が、選択された PVD 閾値を下回っています。

ビット 10 **VOSF**：電圧スケーリング･フラグ

電圧スケーリング変更後に内蔵レギュレータがレディ状態になるには、遅延が必要です。VOSF は、レギュレータが PWR_CR1 レジスタの VOS ビットで定義された電圧レベルに達したことを示します。

0：レギュレータは選択された電圧レンジでレディ状態です。

1：レギュレータ出力電圧が必要な電圧レベルに変化中です。

ビット 9 **REGLPF**：低消費電力レギュレータ･フラグ

このビットは、MCU が低消費電力 RUN モードである場合に、ハードウェアによってセットされます。MCU が低消費電力 RUN モードを終了すると、レギュレータがメインモードでレディ状態になるまで、このビットは 1 の状態を維持します。製品の周波数を上げる前に、このビットのポーリングが必須です。

このビットは、レギュレータがレディ状態になると、ハードウェアによってクリアされます。

0：レギュレータはメイン･モード（MR）でレディ状態です。

1：レギュレータは低消費電力モード（LPR）です。

ビット 8 **REGLPS**：低消費電力レギュレータ開始

このビットは、パワーオンリセット後または STANDBY/SHUTDOWN 後に低消費電力レギュレータがレディ状態であるかどうかの情報を提供します。REGLPS ビットがクリアされている状態で STANDBY モードに移行すると、STANDBY モードからのウェイクアップ時間が増加する場合があります。

0：低消費電力レギュレータは、レディ状態ではありません。

1：低消費電力レギュレータは、レディ状態です。

ビット 7 **Flash_RDY**：Flash レディ･フラグ

このビットは、Flashメモリがパワーダウンからのウェイクアップ後にアクセス可能状態になったことを示すために、ハードウェアによってセットされます。Flashメモリをパワーダウンモードにするには、FPD_LPRUN、FPD_LPSLP、または FPD_STP ビットのいずれかをセットします。

0：Flash メモリがパワーダウンしています。

1：Flash メモリはアクセス可能状態です。

注： システムを SRAM からブートする場合、ユーザアプリケーションはFlashメモリにジャンプする前に、Flash_RDY ビットがセットされるまで待機する必要があります。

ビット 6:0 予約済みであり、リセット値に保持する必要があります。

6.4.7 電源ステータスクリアレジスタ（PWR_SCR）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：このレジスタに書き込むには、通常の APB 書込みに対して、追加の APB サイクルが 3 つ必要です。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBF	Res.	Res.	Res.	CWUF5	CWUF4	CWUF3	CWUF2	CWUF1
							w				w	w	w	w	w

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CSBF**：STANDBY フラグのクリア
このビットをセットすると PWR_SR1 レジスタの SBF フラグがクリアされます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CWUF5**：ウェイクアップフラグ 5 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF5 フラグがクリアされます。

ビット 3 **CWUF4**：ウェイクアップフラグ 4 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF4 フラグがクリアされます。

ビット 2 **CWUF3**：ウェイクアップフラグ 3 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF3 フラグがクリアされます。

ビット 1 **CWUF2**：ウェイクアップフラグ 2 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF2 フラグがクリアされます。

ビット 0 **CWUF1**：ウェイクアップフラグ 1 のクリア
このビットをセットすると PWR_SR1 レジスタの WUF1 フラグがクリアされます。

6.4.8 電源ポート A プルアップ制御レジスタ（PWR_PUCRA）

アドレスオフセット：0x20

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **PU15**：ポート A プルアップビット 15

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[15] でプルアップが有効になります。対応する PD15 ビットもセットされている場合、プルアップは有効になりません。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **PUy**：ポート A プルアップ･ビット y（y = 13 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PA[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.9 電源ポート A プルダウン制御レジスタ（PWR_PDCRA）

アドレスオフセット：0x24

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **PD14**：ポート A プルダウンビット 14

このビットをセットすると、APC ビットが PWR_CR3 レジスタでセットされたときに PA[14] でプルダウンが有効になります。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:0 **PDy**：ポート A プルダウン･ビット y（y = 12 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PA[y] のプルダウンが有効になります。

6.4.10 電源ポート B プルアップ制御レジスタ（PWR_PUCRB）

アドレスオフセット：0x28

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート B プルアップ･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PB[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.11 電源ポート B プルダウン制御レジスタ（PWR_PDCRB）

アドレスオフセット：0x2C

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:5 **PDy**：ポート B プルダウン･ビット y（y = 15 ~ 5）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PB[y] のプルダウンが有効になります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PDy**：ポート B プルダウン･ビット y（y = 3 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PB[y] のプルダウンが有効になります。

6.4.12 電源ポート C プルアップ制御レジスタ（PWR_PUCRC）

アドレスオフセット：0x30

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート C プルアップ･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PC[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.13 電源ポート C プルダウン制御レジスタ（PWR_PDCRC）

アドレスオフセット：0x34

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート C プルダウン･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PC[y] のプルダウンが有効になります。

6.4.14 電源ポート D プルアップ制御レジスタ（PWR_PUCRD）

アドレスオフセット：0x38

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート D プルアップ･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PD[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.15 電源ポート D プルダウン制御レジスタ（PWR_PDCRD）

アドレスオフセット：0x3C

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート D プルダウン･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PD[y] のプルダウンが有効になります。

6.4.16 電源ポート E プルアップ制御レジスタ（PWR_PUCRE）

アドレスオフセット：0x40

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート E プルアップ･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PE[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.17 電源ポート E プルダウン制御レジスタ（PWR_PDCRE）

アドレスオフセット：0x44

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート E プルダウン･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PE[y] のプルダウンが有効になります。

6.4.18 電源ポート F プルアップ制御レジスタ（PWR_PUCRF）

アドレスオフセット：0x48

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PUy**：ポート F プルアップ･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PF[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.19 電源ポート F プルダウン制御レジスタ（PWR_PDCRF）

アドレスオフセット：0x4C

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PDy**：ポート F プルダウン･ビット y（y = 15 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PF[y] のプルダウンが有効になります。

6.4.20 電源ポート G プルアップ制御レジスタ（PWR_PUCRG）

アドレスオフセット：0x50

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **PUy**：ポート G プルアップ･ビット y（y = 10 ~ 0）

このビットをセットすると、PWR_CR3 レジスタ の APC ビットがセットされたときに PG[y] のプルアップが有効になります。対応する PDy ビットもセットされている場合、プルアップは有効になりません。

6.4.21 電源ポート G プルダウン制御レジスタ（PWR_PDCRG）

アドレスオフセット：0x54

リセット値：0x0000 0000

このレジスタは、RCC_APB1RSTR1 レジスタの PWRRST ビットで STANDBY モードを終了するとリセットされません。

アクセス：このレジスタへのアクセスには、通常の APB アクセスに対して、追加の APB サイクルが必要です（書込みは 3 つ、読出しは 2 つ）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:0 **PDy**：ポート G プルダウン･ビット y（y = 10 ~ 0）

このビットをセットすると、PWR_CR3 レジスタの APC ビットがセットされたときに PG[y] のプルダウンが有効になります。

6.4.22 電源制御レジスタ （PWR_CR5）

アドレス･オフセット：0x80

リセット値：0x0000 0100

このレジスタは、パワーオン･リセットのみによってリセットされます。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	R1MODE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **R1MODE**：メインレギュレータレンジ 1 モード

このビットはレンジ 1 のメインレギュレータに対してのみ有効であり、レンジ 2 での効果はありません。システム周波数が 150 MHz を超える場合には、このビットをリセットすることをお勧めします。表 38：レンジ 1 ブーストモード設定を参照してください。

0：レンジ 1 のメイン･レギュレータをブースト･モードに設定します。

1：レンジ 1 のメイン･レギュレータをノーマル･モードに設定します。

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

6.4.23 PWR レジスタマップとリセット値の表

表 49. PWR レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	PWR_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LPR	Res.	Res.	Res.	VOS[1:0]		DBP		Res.	Res.	Res.	FPD_STOP	LPMS[2:0]		
	リセット値																		0				0	1	0					1	0	0	0
0x04	PWR_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVME2	PVME1	Res.	Res.	PLS[2:0]			PVDE
	リセット値																									0	0			0	0	0	0
0x08	PWR_CR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EIWUL	UCPD1_DBDIS	UCPD1_STDBY	Res.	Res.	APC	Res.	RRS	Res.	Res.	Res.	EWUP5	EWUP4	EWUP3	EWUP2	EWUP1
	リセット値																	0	0	0			0		0				0	0	0	0	0
0x0C	PWR_CR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBRS	VBE	Res.	Res.	Res.	WP5	WP4	WP3	WP2	WP1
	リセット値																							0	0				0	0	0	0	0
0x10	PWR_SR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUFI	Res.	Res.	Res.	Res.	Res.	Res.	SBF	Res.	Res.	Res.	WUF5	WUF4	WUF3	WUF2	WUF1
	リセット値																	0							0				0	0	0	0	0
0x14	PWR_SR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PVMO2	PVMO1	Res.	Res.	PVDO	VOSF	REGLPF	REGLPS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																	0	0			0	0	0	0								
0x18	PWR_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSBF	Res.	Res.	Res.	CWUF5	CWUF4	CWUF3	CWUF2	CWUF1
	リセット値																								0				0	0	0	0	0
0x20	PWR_PUCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	Res.	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	PWR_PDCRA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD14	Res.	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																		0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	PWR_PUCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	PWR_PDCRB	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	Res.	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0
0x30	PWR_PUCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x34	PWR_PDCRC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	PWR_PUCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C	PWR_PDCRD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	D1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	PWR_PUCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 49. PWR レジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	PWR_PDCRE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	PWR_PUCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU15	PU14	PU13	PU12	PU11	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	PWR_PDCRF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	PWR_PUCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PU10	PU9	PU8	PU7	PU6	PU5	PU4	PU3	PU2	PU1	PU0
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x54	PWR_PDCRG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x80	PWR_CR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	R1MODE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																								1								

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

7 リセットおよびクロック制御（RCC）

7.1 リセット

リセットには、システムリセット、電源リセット、RTC ドメインリセットの 3 種類があります。

7.1.1 電源リセット

電源リセットは、次のいずれかのイベントが発生したときに生成されます。

1. パワーオンリセット（POR）またはブラウンアウトリセット（BOR）
2. STANDBY モードを終了するとき
3. SHUTDOWN モードを終了するとき

パワーオンリセットまたはパワーダウンリセット（POR/PDR）を含むブラウンアウトリセットは、RTC ドメインを除くすべてのレジスタをリセット値に設定します。

STANDBY モードを終了すると、V_{CORE} ドメインのすべてのレジスタがリセット値にセットされます。V_{CORE} ドメイン以外のレジスタ（RTC、WKUP、IWDG および STANDBY/SHUTDOWN モード制御）は影響を受けません。

SHUTDOWN モードを終了すると、ブラウンアウトリセットが生成され、RTC ドメイン以外のすべてのレジスタがリセットされます。

7.1.2 システムリセット

システムリセットは、クロック制御／ステータスレジスタ（RCC_CSR）のリセットフラグと RTC ドメインのレジスタを除き、すべてのレジスタをリセット値に設定します。

システムリセットは、次のイベントのいずれかの発生時に生成されます。

1. NRST ピンのローレベル（外部リセット）
2. ウィンドウ型ウォッチドッグイベント（WWDG リセット）
3. 独立型ウォッチドッグイベント（IWDG リセット）
4. ソフトウェアリセット（SW リセット）（ソフトウェアリセットを参照）
5. 低電力モードセキュリティリセット（低電力モードセキュリティリセットを参照）
6. オプションバイトローダリセット（オプションバイトローダリセットを参照）
7. ブラウンアウトリセット

リセットソースは、制御／ステータスレジスタ（RCC_CSR）のリセットフラグを確認することによって識別できます（セクション 7.4.28：制御／ステータスレジスタ（RCC_CSR）を参照）。

NRST ピン（外部リセット）

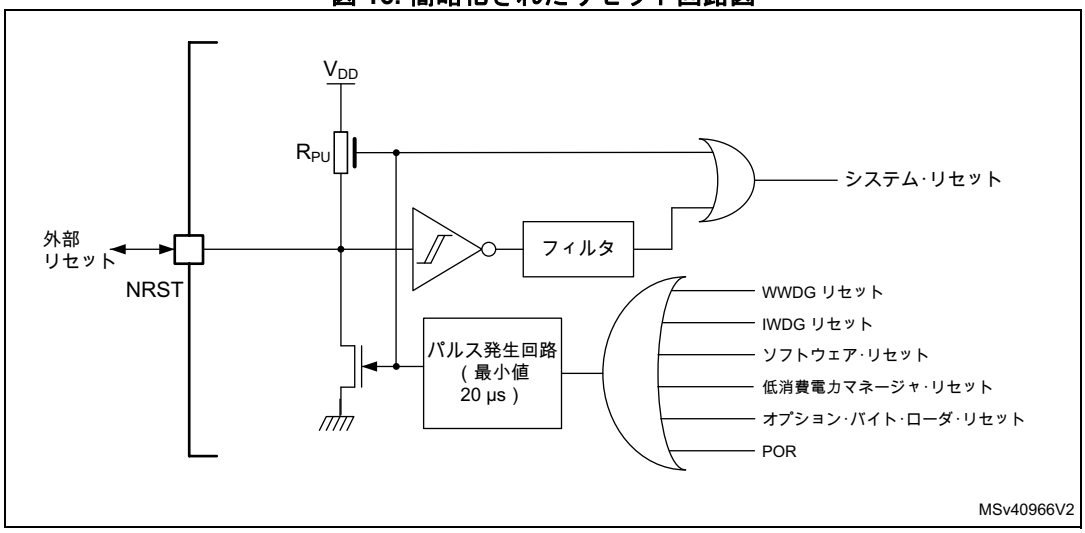
NRST ピンは、特定のオプションビット（NRST_MODE）を通して次の操作に設定できます。

- リセット入出力（デバイス供給時のデフォルト）
  ピン上のあらゆる有効なリセット信号がデバイスの内部ロジックに伝播され、このピンに対するパルス発生回路を介して、すべての内部リセットソースが外部的に駆動されます。GPIO 機能（PG10）は使用できません。パルス発生回路は、NRST ピンで出力する各内部リセットソースについて 20 µs の最小リセットパルス期間を保証します。内部リセットホルダオプションは、オプションバイトで有効化されている場合に、ピンの電圧が VIL 閾値と一致するまでピンがローに引き下げられていることを確認するために使用できます。この機能により、ラインで大きな静電容量負荷が発生したときに、外部コンポーネントによる内部リセットソースの検出が保証され

ます。内部リセットの場合、プルアップ抵抗による消費電力を削減するために、内部プルアップ RPU が無効化されます。各デバイスがパワーオンリセット（デバイスの電源オンまたは SHUTDOWN モードからのウェイクアップ）している間（オプションバイトがロードされるまで）、（オプションバイトの設定とは無関係に）このモードは常に有効です。

- リセット入力
  このモードでは、NRST ピンの有効なリセット信号はすべてデバイスの内部ロジックに伝播されますが、デバイスによって内部的に生成されたリセットはピンには表示されません。この設定では、GPIO 機能（PG10）は使用できません。
- GPIO
  このモードでは、このピンを PG10 で標準の GPIO として使用できます。ピンのリセット機能は使用できません。リセットはデバイスの内部リセットソースからのみ実行でき、ピンには伝播しません。

図 16. 簡略化されたリセット回路図



ソフトウェアリセット

デバイス上で強制的にソフトウェアリセットを実行するには、FPU 搭載 Cortex®-M4 のアプリケーション割込みおよびリセット制御レジスタの SYSRESETREQ ビットをセットする必要があります（STM32F3xx/F4xx/L4xx Cortex®-M4 プログラミングマニュアル（PM0214）を参照）。

低電力モードセキュリティリセット

誤って重要なアプリケーションが低電力モードに移行しないように、2 つの低電力モードセキュリティリセットが使用できます。オプションバイトで有効になっている場合、以下の条件下でリセットを生成します。

1. STANDBY モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_STDBY ビットをリセットすることで有効になります。この場合、STANDBY モードへの遷移シーケンスが正常に実行されるたびに、STANDBY モードに移行する代わりにデバイスがリセットされます。
2. STOP モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_ストップビットをリセットすることで有効になります。この場合、STOP モードへの遷移シーケンスが正常に実行されるたびに、STOP モードに移行する代わりにデバイスがリセットされます。
3. SHUTDOWN モードへの移行：この種類のリセットは、ユーザオプションバイトの nRST_SHDW ビットをリセットすることで有効になります。この場合、SHUTDOWN モードへの遷移シーケンスが正常に実行されるたびに、SHUTDOWN モードに移行する代わりにデバイスがリセットされます。

ユーザオプションバイトの詳細については、セクション 3.4.1：オプションバイトの説明を参照してください。

オプションバイトローダリセット

オプションバイトローダリセットは、Flash_CR レジスタの OBL_LAUNCH ビット（ビット 27）がセットされると生成されます。このビットは、ソフトウェアによるオプション･バイト･ローディングを起動するために使用されます。

7.1.3 RTC ドメインリセット

RTC ドメインには 2 つの固有のリセットがあります。

RTC ドメインのリセットは、次のいずれかのイベントが発生したときに生成されます。

1. RTC ドメイン制御レジスタ（RCC_BDCR）の BDRST ビットをセットすることでトリガされるソフトウェアリセット
2. V_{DD} または V_{BAT} パワーオン。ただし、両方の電源供給がともにオフ状態であった場合。

RTC ドメインリセットは、LSE オシレータ、RTC、バックアップレジスタ、および RCC RTC ドメイン制御レジスタにのみ影響します。

7.2 クロック

次の 3 つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます。

- HSI16（ハイスピード内部）16 MHz RC オシレータクロック
- HSE オシレータクロック、4〜48 MHz
- PLL クロック

リセットからの起動後は、HSI16 がシステム･クロック･ソースとして使用されます。

デバイスには、次の追加のクロックソースがあります。

- 32 kHz 低速内蔵 RC（LSI RC）。独立型ウォッチドッグを駆動し、オプションで、STOP および STANDBY モードからの自動ウェイクアップに使用される RTC を駆動します。
- 32.768 kHz 低速外部クリスタル（LSE クリスタル）。オプションで、リアルタイムクロック（RTCCLK）を駆動します。
- USB FS および RNG を駆動できる RC 48 MHz 内部クロックソース（HSI48）。

それぞれのクロックソースは、使用しないときに個別にオン／オフを切り替えて、電力消費を最適化可能です。

複数のプリスケーラを使用して AHB 周波数、APB1 および APB2 ドメインを設定できます。AHB、APB1 および APB2 ドメインの最高周波数は 170 MHz です。

すべてのペリフェラルクロックは、次を除いてそれらのバスクロックから生成されます（HCLK、PCLK1、または PCLK2）。

- USB デバイス FS および RNG に使用される 48 MHz クロック。このクロックは、次の 4 つのうちのいずれかのソースから生成されます（ソフトウェアによって選択される）:
  - PLL 「Q」クロック
  - HSI48 内部オシレータ

  使用できる場合、HSI48 48 MHz クロックを 適切に USB OTG FS にクロック接続できるクロックリカバリシステムと結合できます（クリスタルレスソリューション）。
- 次のうちのいずれかのソースから生成される ADC クロック（ソフトウェアによって選択される）:
  - システムクロック（SYSCLK）
  - PLL 「P」クロック
- 次の 4 つのうちのいずれかのソース（ソフトウェアで選択）から生成される U(S)ART クロック:
  - システムクロック（SYSCLK）
  - HSI16 クロック
  - LSE クロック
  - APB1 または APB2 クロック（U(S)ART に配置されている APB に応じて、PCLK1 または PCLK2）

  STOP モードからのウェイクアップは、クロックが HSI16 または LSE である場合のみサポートされます。
- 次の 3 つのうちのいずれかのソースから生成される I^2C クロック（ソフトウェアによって選択される）:
  - システムクロック（SYSCLK）
  - HSI16 クロック
  - APB1 クロック（PCLK1）

  STOP モードからのウェイクアップは、クロックが HSI16 である場合のみサポートされます。
- 次のうちのいずれかのソースから生成される SAI1 クロック（ソフトウェアによって選択される）:
  - I2S_CKIN に配置される外部クロック
  - システムクロック
  - PLL 「Q」クロック
  - HSI16 クロック
- 次のうちのいずれかのソースから生成される QUADSPI カーネルクロック（ソフトウェアによって選択される）:
  - システムクロック
  - PLL 「Q」クロック
  - HSI16 クロック
- 次の 5 つのうちのいずれかのソースから生成される低電力タイマ（LPTIM1）クロック（ソフトウェアによって選択される）:
  - LSI クロック
  - LSE クロック
  - HSI16 クロック
  - APB1 クロック（PCLK1）
  - LPTIMx_IN1 ピンに配置された外部クロック

  STOP モードの機能（ウェイクアップ含む）は、クロックが LSI または LSE である場合または外部クロックモードのときのみサポートされます。

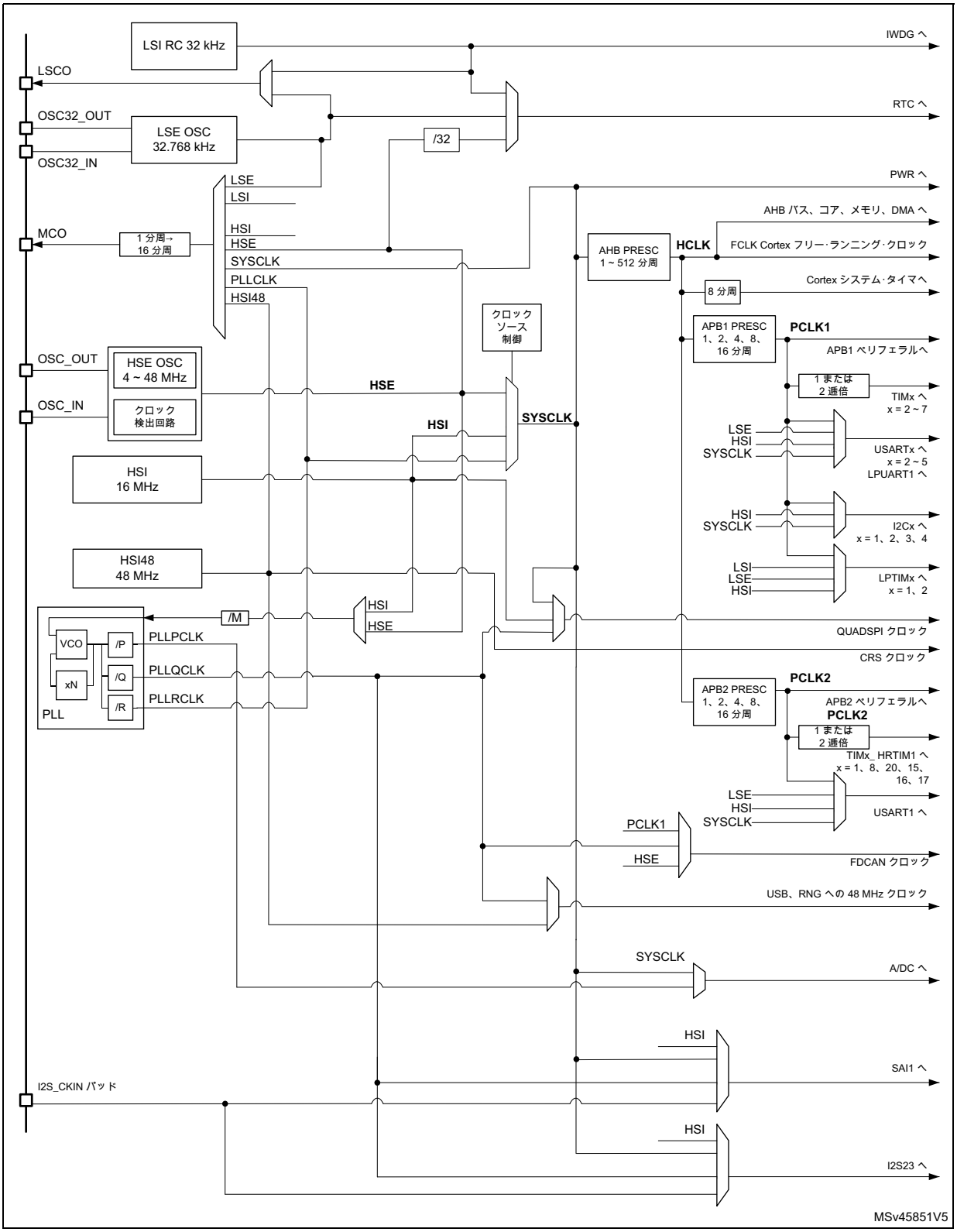
- 次の 3 つのうちのいずれかのソースから生成される RTC クロック（ソフトウェアによって選択される）
  - LSE クロック
  - LSI クロック
  - 32 分周された HSE クロック

  STOP モードの機能（ウェイクアップ含む）は、クロックが LSI または LSE である場合のみサポートされます。
- IWDG クロックは、常に LSI クロックから生成されます。
- UCPD1 クロック。HSI16 クロックから生成されます。
- FDCAN1 クロック。次の 2 つのうちのいずれかのソースから生成されます （ソフトウェアによって選択される）:
  - HSE クロック
  - PLL 「Q」クロック
  - PCLK クロック

RCC は Cortex® システム･タイマ（SysTick）の外部クロックに、8 分周した AHB クロック（HCLK）を供給します。SysTick は、この分周されたクロックか直接 Cortex® クロック（HCLK）で動作でき、SysTick 制御およびステータスレジスタで設定可能です。

FCLK は FPU 搭載 Cortex®-M4 のフリーランニングクロックとして動作します。詳細については、Cortex®-M4 プログラミングマニュアル（PM0214）を参照してください。

図 17. クロックツリー



1. 内部および外部クロックソース特性の詳細については、デバイスのデータシートの「電気的特性」のセクションを参照してください。

2. ADC クロックは、ADC バスインタフェースの AHB クロックから生成し、プログラム可能な係数（1、2、または 4）で分周することができます。プログラム可能な係数が「1」の場合、AHB プリスケーラは「1」でなければなりません。

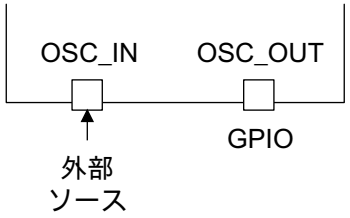
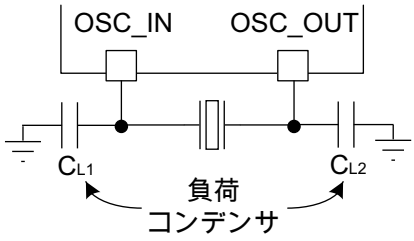
7.2.1 HSE クロック

高速外部クロック信号（HSE）は、次のどちらかのクロックソースから生成できます。

- HSE 外部クリスタル／セラミック発振子
- HSE ユーザ外部クロック

波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したオシレータに応じて調整する必要があります。

図 18. HSE/LSE クロックソース

クロックソース	ハードウェア構成
外部クロック	
クリスタル／ セラミック発振子	

外部クリスタル／セラミック発振子（HSE クリスタル）

4 から 48 MHz の外部オシレータには、メインクロックの周波数を非常に高い精度で生成できるという利点があります。

関連するハードウェア構成を図 18 に示します。詳細については、データシートの電気的特性のセクションを参照してください。

クロック制御レジスタ（RCC_CR）の HSERDY フラグは、HSE オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。クロック割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

HSE クリスタルは、クロック制御レジスタ（RCC_CR）の HSEON ビットを使用してオン／オフできます。

外部ソース（HSE バイパス）

このモードでは、外部クロックソースが必要です。最大 48 MHz までの周波数を使用できます。このモードを選択するには、クロック制御レジスタ（RCC_CR）HSEBYP および HSEON ビット をセットします。周波数（データシートを参照）に応じて 40～60% までのデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）で OSC_IN ピンを駆動する必要があり、その間、OSC_OUT ピンは GPIO として使用できます。図 18 を参照してください。

7.2.2 HSI16 クロック

HSI16 クロック信号は、内部 16 MHz RC オシレータから生成されます。

HSI16 RC オシレータには、低コスト（外付部品なし）でクロックソースを供給できるという利点があります。同時に、HSE クリスタルオシレータよりもスタートアップ時間を短縮できますが、較正を実施していても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

HSI16 クロックは、STOP モード（STOP 0、STOP 1）からのウェイクアップ後、システムクロックとして選択できます。セクション 7.3 : 低電力モード を参照してください。これは、HSE クリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。セクション 7.2.9 : クロックセキュリティシステム（CSS）を参照してください。

較正

RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、T_A = 25°C で 1 % の精度を確保するよう、製造時に ST 社においてデバイスごとの較正が行われています。

リセット後、工場較正値が 内部クロックソース較正レジスタ（RCC_ICSCR）の HSICAL[7:0] ビットにロードされます。

アプリケーションが電圧または温度の変動を受ける場合は、これが RC オシレータの速度に影響を与えることがあります。内部クロックソース較正レジスタ（RCC_ICSCR）の HSITRIM[6:0] を使用して、アプリケーションで HSI16 周波数をトリミングできます。

HSI16 の周波数変動の測定方法の詳細については、セクション 7.2.16 : TIM5/TIM15/TIM16/TIM17 を使用した内部／外部クロックの測定を参照してください。

クロック制御レジスタ（RCC_CR）の HSIRDY フラグは、HSI16 オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI16 出力クロックはリリースされません。

HSI16 オシレータ は クロック制御レジスタ（RCC_CR）の HSION ビットを使用してオン／オフの切り替えができます。

HSI16 の信号は、HSE クリスタルオシレータに障害が起きた場合のバックアップソース（補助クロック）としても使用できます。セクション 7.2.9 : 277 ページのクロックセキュリティシステム（CSS）を参照してください。

7.2.3 HSI48 クロック

HSI48 クロック信号は、内部 48 MHz RC オシレータから生成され、直接USB、または乱数生成器（RNG）に使用できます。

内部 48 MHz RC オシレータは、主に、特殊なクロックリカバリシステム（CRS）回路によって USB ペリフェラルに高精度のクロックを提供します。CRS では、動作中のオシレータの周波数を自動的かつ迅速に調整するために、LSE または外部信号を使用できます。これは、システムが STOP または STANDBY モードに移行した直後に無効化されます。CRS が使用されない場合、HSI48 RC オシレータは製造工程でのばらつきの影響を受けるデフォルトの周波数で実行されます。

クロックリカバリ RC レジスタ（RCC_CRRCR）の HSI48RDY フラグは、HSI48 RC オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 RC オシレータ出力クロックはリリースされません。

HSI48 は、クロックリカバリ RC レジスタ（RCC_CRRCR）の HSI48ON ビットを使用してオン／オフできます。

7.2.4 PLL

内部 PLL は、HSI16 または HSE 出力クロック周波数を逓倍するために使用できます。PLL の入力周波数は、デバイスのデータシートに規定された範囲内とする必要があります。選択したクロック・ソースは、プログラム可能な係数 PLLM によって 1 から 8 分周され、要求される入力範囲でクロック周波数を供給します。図 17 : クロックツリー および PLL 設定レジスタ（RCC_PLLCFGR）を参照してください。

PLL 設定（入力クロックおよび逓倍数の選択）は、PLL を有効にする前に実行する必要があります。PLL が有効になると、これらのパラメータは変更できません。

PLL 設定を変更するには、次の手順に従います。

1. クロック制御レジスタ（RCC_CR）で PLLON を 0 に設定して PLL を無効にします。
2. PLLRDY がクリアされるまで待ちます。PLL が完全に停止します。
3. 任意のパラメータを変更します。
4. PLLON を 1 にセットして PLL を再度有効にします。
5. PLL 設定レジスタ（RCC_PLLCFGR）で PLLPEN、PLLQEN、PLLREN を設定して対象の PLL 出力を有効にします。

クロック割込み有効レジスタ（RCC_CIER）で有効になっていれば、PLL がレディ状態になると割込みを生成することができます。

PLL 出力周波数は 170 MHz を超えてはなりません。

各 PLL 出力クロックのイネーブルビット（PLLPEN、PLLQEN、PLLREN）は、対応する PLL を停止せずにいつでも変更できます。PLLCLK がシステムクロックとして使用されている場合は、PLLREN をクリアできません。

7.2.5 LSE クロック

LSE クリスタルは、32.768 kHz の低速外部クリスタルまたはセラミック発振子です。時計／カレンダ、その他のタイミング機能のためのリアルタイムクロックペリフェラル（RTC）に、低電力ながら高精度のクロックソースを供給できるという利点があります。

LSE クリスタルは、RTC ドメイン制御レジスタ（RCC_BDCR）の LSEON ビットを使用してオン／オフの切り替えができます。安定性、短い起動時間、および低い電力消費のバランスを取るために、クリスタルオシレータの駆動能力は、RTC ドメイン制御レジスタ（RCC_BDCR）の LSEDRV[1:0] ビットによって、駆動中に変更できます。LSE 駆動は、LSE がオンの場合、より低い駆動能力（LSEDRV=00）まで低下させることができます。ただし、一旦LSEDRV が選択されると、LSEON=1 の場合、駆動能力を上げられません。

RTC ドメイン制御レジスタ（RCC_BDCR）の LSERDY フラグは、LSE クリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSE クリスタル出力クロック信号はリリースされません。クロック割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

外部ソース（LSE バイパス）

このモードでは、外部クロックソースが必要です。最大 1 MHz までの周波数を使用できます。このモードを選択するには、SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）の LSEBYP および LSEON ビットをセットします。50 % までのデューティサイクルを持つ外部クロック信号（矩形波、正弦波、または三角波）で OSC_32_IN ピンを駆動する必要があり、その間、OSC_32_OUT ピンを GPIO として使用することができます。図 18 を参照してください。

7.2.6 LSI クロック

LSI RC は、独立型ウォッチドッグ（IWDG）および RTC の STOP モードおよび STANDBY モードでの動作を可能にする低電力のクロックソースとして動作します。クロック周波数は 32 kHz です。詳細については、データシートの電気的特性のセクションを参照してください。

LSI RC は、制御／ステータスレジスタ（RCC_CSR）の LSION ビットを使用してオン／オフの切り替えができます。

制御／ステータスレジスタ（RCC_CSR）の LSIRDY フラグは、LSI オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。クロック割込み有効レジスタ（RCC_CIER）で有効になっていれば、割込みを生成することができます。

7.2.7 システムクロック（SYSCLK）の選択

次の 4 つのクロックソースを使用して、システムクロック（SYSCLK）を駆動できます：

- HSI16 オシレータ
- HSE オシレータ
- PLL

システム･クロックの最大周波数は 170 MHz です。システムリセット後は、HSI16 オシレータがシステムクロックとして選択されています。クロックソースが直接、または PLL を経由してシステムクロックとして使用されているときに、このクロック信号を停止することはできません。

あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができている場合（起動遅延時間を経てクロックが安定している状態、または、PLL がロックされている状態）にのみ行われます。準備ができていないクロックソースが選択された場合は、ク

ロックソースの準備ができたときに切り替えが行われます。内部クロックソース較正レジスタ（RCC_ICSCR）のステータスビットは、どのクロックの準備ができているか、およびどのクロックがシステムクロックとして使用されているかを示します。

低速のシステムクロックから高速のシステムクロックに、または高速から低速に切り替えるには、少なくとも 1 µs の間、中間速度クロックの遷移状態を用いることをお勧めします。

クロックソース切り替え条件

- HSE または HSI16 から、80 MHz を超える AHB 周波数（HCLK）の PLL への切替え
- 80 MHz を超える HCLK の PLL から、HSE または HSI16 への切替え

遷移状態

- AHB プリスケーラ HPRE[3:0] ビットをセットして、システム周波数を 2 分周します。
- システムクロックを PLL に切り替えます。
- 最低でも 1 µs の間待ってから、AHB プリスケーラのビットを必要な HCLK 周波数に再設定します。

7.2.8 クロックソースの周波数と電圧スケーリング

次の表に、製品電圧レンジに応じて異なるクロックソースの周波数を示します。

表 50. クロックソースの周波数

製品電圧レンジ	クロック周波数		
	HSI16	HSE	PLL
レンジ 1 ブースト·モード	16 MHz	48 MHz	170 MHz
レンジ 1 ノーマルモード	16 MHz	48 MHz	150 MHz
レンジ2	16 MHz	26 MHz	26 MHz

7.2.9 クロックセキュリティシステム（CSS）

クロックセキュリティシステムはソフトウェアで有効にできます。この場合、HSE オシレータのスタートアップ遅延時間の後にクロック検出回路が有効になり、オシレータが停止すると検出回路も無効になります。

HSE クロックに異常が検出された場合、HSE オシレータは自動的に無効になり、クロック異常イベントが高機能制御タイマ（TIM1/TIM8/TIM20 および TIM15/16/17）のブレーク入力と hrtim_sys_flt に送られます。また、ソフトウェアに異常を通知する割込み（クロックセキュリティシステム割込み（CSSI））が生成されるので、MCU は対応処理を行うことができます。CSSI は、FPU 搭載 Cortex®-M4 NMI（ノンマスカブル割込み）例外ベクタにリンクされています。

注： **CSS を有効にすると、HSE クロックに障害が発生した場合に CSS 割込みが発生し、NMI が自動的に生成されます。NMI は、CSS 割込みペンディング·ビットがクリアされない限り、無限に実行されます。そのため、NMI 割込みサービスルーチン（ISR）では、クロック割込みクリアレジスタ（RCC_CICR）の CSSC ビットをセットして CSS 割込みをクリアする必要があります。**

HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合（間接的とは、PLL 入力クロックとして使用され、その PLL クロックがシステムクロックとして使用されることを意味する）、障害が検出されると、システムクロックが HSI16 オシレータに切り替えられ、HSE オシレータは無効になります。障害発生時に HSE クロック（分周されているかいないかにかかわらず）がシステムクロックとして使用されている PLL のクロック入力であった場合には、PLL も無効になります。

7.2.10 LSE のクロックセキュリティシステム

LSE のクロックセキュリティシステムは、制御／ステータスレジスタ（RCC_CSR）の LSECSSON ビットをソフトウェアで書き込むことによって有効にできます。このビットは、ハードウェアリセットまたは RTC ソフトウェアリセット、または LSE で障害検出後にのみ無効にできます。LSECSSON は、LSE と LSI を有効（LSEON および LSION を有効化）かつレディ状態（ハードウェアで LSERDY および LSIRDY をセット）にし、RTCSEL で RTC クロックを選択した後で書き込む必要があります。

LSE の CSS は、VBAT を除くすべてのモードで動作します。システムリセット時（パワーオンリセットを除く）でも動作します。外部 32 kHz オシレータで障害が検出された場合、LSE クロックは RTC に供給されなくなりますが、レジスタに対してハードウェアのアクションは実行されません。HSI16 が PLL モードであった場合、このモードは無効になります。

STANDBY モードでは、ウェイクアップが生成されます。その他のモードでは、ソフトウェアをウェイクアップさせるために割込みを送信できます（クロック割込み有効レジスタ（RCC_CIER）、クロック割込みフラグ･レジスタ（RCC_CIFR）、クロック割込みクリアレジスタ（RCC_CICR）を参照）。

その場合、ソフトウェアは、LSECSSON ビットを無効化し、障害のある 32 kHz オシレータを停止（LSEON を無効化）して、RTC クロックソースを変更（クロックなし、または RTCSEL で LSI や HSEを選択）するか、必要なアクションを行いアプリケーションを保護する必要があります。

CSS の誤検出を回避するには、LSE オシレータの周波数を 30 kHz 以上にする必要があります。

7.2.11 ADC クロック

ADC クロックは、システム･クロックまたは PLL 「P」出力から生成されます。ADCx_CCR レジスタの設定により、1、2、4、6、8、10、12、16、32、64、128、256 のプリスケーラ値で分周できます。このクロックは AHB クロックとは非同期です。代わりに、ADC バスインタフェースの AHB クロックから ADC クロックを生成し、プログラム可能な係数（1、2、または 4）で分周できます。プログラム可能な係数は、ADCx_CCR の CKMODE ビットフィールドを使用して設定されます。

プログラムされた係数が「1」の場合、AHB プリスケーラは「1」にセットする必要があります。

ADC クロックの最大周波数については、デバイスのデータシートを参照してください。

7.2.12 RTC クロック

RTCCLK クロックソースは HSE/32、LSE、または LSI クロックのいずれかです。これは、RTC ドメイン制御レジスタ（RCC_BDCR）の RTCSEL[1:0] ビットをプログラムすることで選択できます。この選択は、RTC ドメインをリセットしない限り変更できません。RTC を正常に動作させるには、PCLK 周波数が常に RTCCLK 周波数以上になるようにシステムを設定する必要があります。

LSE クロックは RTC ドメインに属しますが、HSE と LSI クロックはそうではありません。そのため、次のようになります。

- LSE が RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合でも、V_{BAT} の供給が保たれている限り、RTC は動作を続けます。
- LSI が RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合、RTC の状態は保証されません。
- プリスケーラで分周された HSE クロックが RTC クロックとして使用されている場合：
  - V_{DD} 供給がオフになった場合や、内部電圧レギュレータがオフになった場合（V_{CORE} ドメインからの電力供給が停止された場合）、RTC の状態は保証されません。

RTC クロックが LSE や LSI の場合、RTC は、システムリセット中でもクロック供給された状態、かつ機能的な状態のままです。

7.2.13 タイマクロック

タイマクロック周波数は、ハードウェアによって自動で設定されます。これには、2 つのケースがあります。

1. APB プリスケーラが 1 の場合、タイマクロック周波数は APB ドメインのクロックと同じ周波数に設定されます。
2. そうでない場合は、APB ドメインの周波数の 2 倍に設定されます。

7.2.14 ウォッチドッグクロック

独立型ウォッチドッグ（IWDG）がハードウェアのオプションまたはソフトウェアアクセスによって開始された場合、LSI オシレータは強制的にオンになり、オフにすることはできません。LSI オシレータの起動時の過渡期間が終わると、このクロックが IWDG に供給されます。

7.2.15 クロック信号出力

- MCO

  マイクロコントローラ·クロック出力（MCO）機能では、外部 MCO ピンにクロックを出力できます。8 つのクロック信号のうちの 1 つを MCO クロックとして選択できます。
  - LSI
  - LSE
  - SYSCLK
  - HSI16
  - HSI48
  - HSE
  - PLLCLK

  選択はクロック設定レジスタ（RCC_CFGR）の MCOSEL[3:0] ビットによって制御されます。選択されたクロックは、クロック設定レジスタ（RCC_CFGR）の MCOPRE[2:0] フィールドで分周できます。
- LSCO

  別の出力（LSCO）では、外部 LSCO ピンに低速クロックを出力できます。
  - LSI
  - LSE

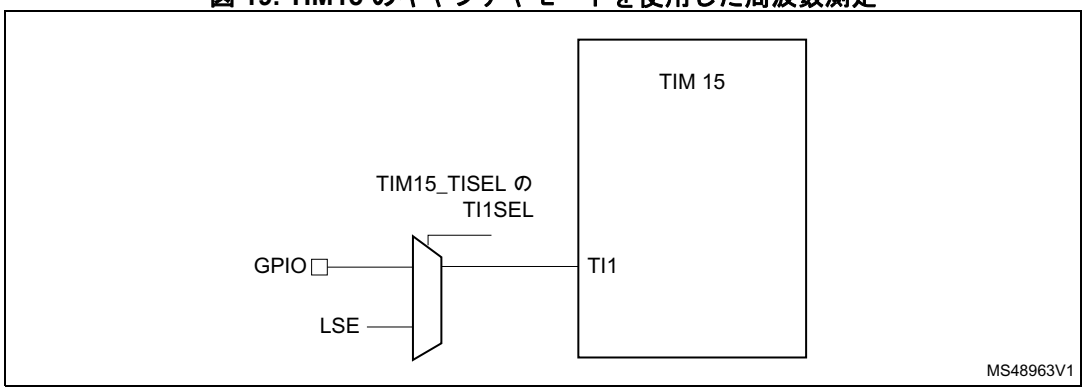
  この出力は、STOP モード（STOP 0 および STOP 1）および STANDBY モードでも維持されます。LSCOSEL によって選択され、RTC ドメイン制御レジスタ（RCC_BDCR）の LSCOEN で有効化されます。

MCO クロック出力には、MCO ピンで対応するオルタネート機能を選択する必要があります。LSCO ピンは、デフォルトの POR 状態に保つ必要があります。

7.2.16 TIM5/TIM15/TIM16/TIM17 を使用した内部／外部クロックの測定

図 19、図 20、図 21 、および 図 22に示すように、TIM5、TIM15、TIM16、または TIM17 のチャネル 1 の入力キャプチャを使用して、ボード上のすべてのクロックソースの周波数を間接的に測定することができます。

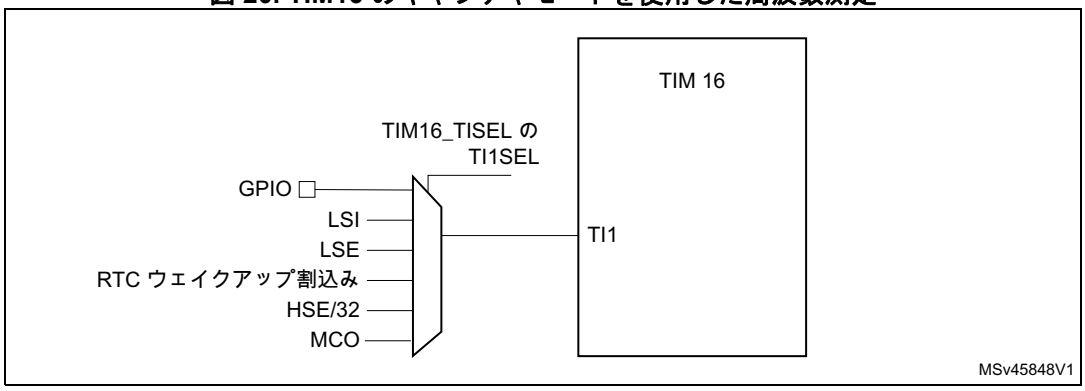
図 19. TIM15 のキャプチャモードを使用した周波数測定



タイマ 15 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。以下を使用できます。

- TIM15 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM15 のチャネル 1 は、LSE に接続されます。

図 20. TIM16 のキャプチャモードを使用した周波数測定

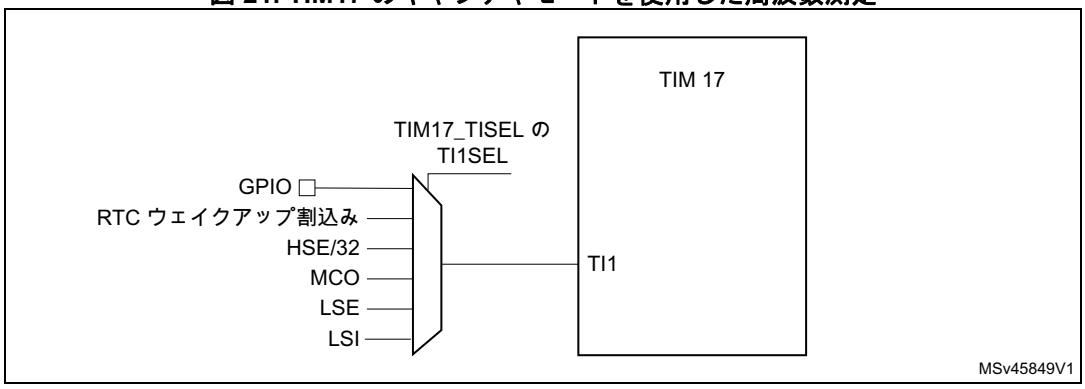


タイマ 16 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。

以下を使用できます。

- TIM16 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM16 のチャネル 1 は、LSI クロックに接続されます。
- TIM16 のチャネル 1 は、LSE クロックに接続されます。
- TIM16 のチャネル 1 は RTC ウェイクアップ割込み信号に接続されます。この場合、RTC 割込みを有効にする必要があります。
- TIM16 のチャネル 1 は、HSE/32 クロックに接続されます。
- TIM16 のチャネル 1 は、MCO に接続されます。

図 21. TIM17 のキャプチャモードを使用した周波数測定

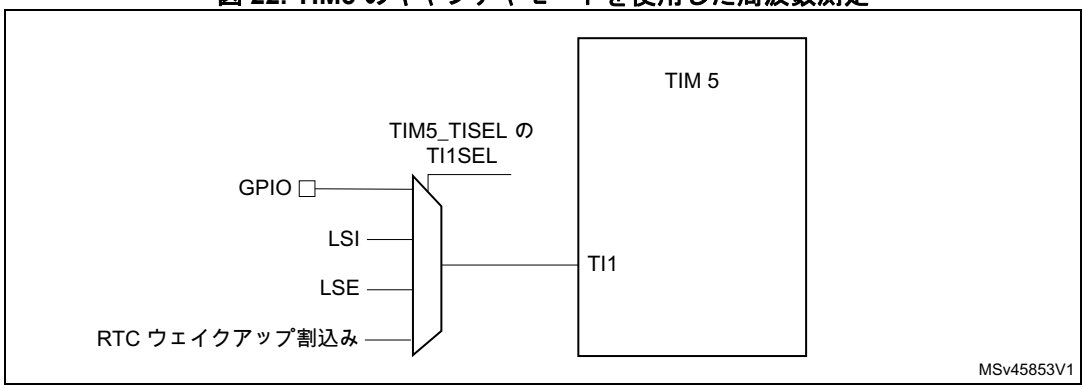


タイマ 17 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。

以下を使用できます。

- TIM17 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM17 のチャネル 1 は RTC ウェイクアップ割込みに接続されます。この場合、RTC 割込みを有効にする必要があります。
- TIM17 のチャネル 1 は、HSE/32 クロックに接続されます。
- TIM17 のチャネル 1 はマイクロコントローラクロック出力（MCO）に接続されます。この選択は、クロック設定レジスタ（RCC_CFGR）の MCOSEL[3:0] ビットで制御されます。
- TIM17 のチャネル 1 は、LSE クロックに接続されます。
- TIM17 のチャネル 1 は、LSI クロックに接続されます。

図 22. TIM5 のキャプチャモードを使用した周波数測定



タイマ 5 の入力キャプチャチャネルは GPIO ラインまたは MCU の内部クロックです。

以下を使用できます。

- TIM5 のチャネル 1 は、GPIO に接続されます。デバイスのデータシートのオルタネート機能配置表を参照してください。
- TIM5 のチャネル 1 は、LSI クロックに接続されます。
- TIM5 のチャネル 1 は、LSE クロックに接続されます。
- TIM5 のチャネル 1 は RTC ウェイクアップ割込み信号に接続されます。この場合、RTC 割込みを有効にする必要があります。

HSI16 の較正

TIM15 および TIM16 では、LSE をチャネル 1 の入力キャプチャに接続する主な目的は、HSI16 システムクロックを正確に測定できるようにすることです（これには、HSI16 がシステムクロックソースとして使用されている必要があります）。LSE 信号の連続したエッジ間の HSI16 クロックカウント数により、内部クロック周期が測定されます。LSE クリスタルの高い精度（通常は数十 ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造、工程、温度および／または電圧に関連した周波数偏差に対してクロックソースを調整して補正することができます。

HSI16 オシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります。

この基本概念は、相対測定（HSI16/LSE 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接にかかわっています。この比が大きければ大きいほど、測定精度は向上します。

LSE を使用できない場合、可能な限り正確な較正を達成するには、HSE/32 を選択することをお勧めします。

LSI の較正

LSI の較正は HSI16 と同じ方法で行いますが、リファレンスクロックを変更します。LSI クロックを TIM16 チャネル 1 の入力キャプチャに接続する必要があります。次に、HSE をシステム･クロック･ソースとして定義し、LSI 信号の連続したエッジ間のクロック･カウント数により、内部低速クロック周期が測定されます。

この基本概念は、相対測定（HSE/LSI 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接にかかわっています。この比が大きければ大きいほど、測定精度は向上します。

7.2.17 ペリフェラル･クロック有効化レジスタ（RCC_AHBxENR、RCC_APBxENRy）

各ペリフェラルクロックは、RCC_AHBxENR、RCC_APBxENRy レジスタの xxxxEN ビットによって有効にできます。

ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

イネーブルビットには、グリッチのないペリフェラルのクロックを生成する同期メカニズムが含まれています。イネーブルビットがセットされてからクロックがアクティブになるまでに、2 クロックサイクルの遅延があります。

注意： **ペリフェラルのクロックを有効にした直後、ソフトウェアは遅延を待ってから、ペリフェラルレジスタにアクセスする必要があります。**

7.3 低電力モード

- DMA クロックを含む AHB および APB ペリフェラルクロックは、ソフトウェアで無効にできます。
- SLEEP モードおよび低電力 SLEEP モードでは CPU クロックを停止します。メモリインタフェースクロック（Flash ならびに SRAM1、SRAM2 および CCM SRAM インタフェース）は、ソフトウェアによって SLEEP モード中に停止できます。AHB - APB ブリッジのクロックは、SLEEP モード中にこれらに接続されたペリフェラルのクロックがすべて無効になった場合に、ハードウェアによって無効化されます。
- STOP モード（STOP 0 および STOP 1）では、V_{CORE} ドメインのすべてのクロックを停止し、PLL、HSI16 および HSE オシレータを無効にします。

  すべての U(S)ART、LPUART と I^2C では、MCU が STOP モードの場合でも、HSI16 オシレータを有効にすることができます（HSI16 が当該ペリフェラルのクロックソースとして選択されている場合）。

  すべての U(S)ART および LPUART は、システムが STOP モードの場合（LSE が当該ペリフェラルのクロックソースとして選択されている場合）で LSE オシレータが有効な場合（LSEON）に、LSE オシレータによる駆動もできます。その場合、LSE は STOP モードで常にオンになります（LSE オシレータをオンにする機能はありません）。
- STANDBY モードおよび SHUTDOWN モードでは、V_{CORE} ドメインのすべてのクロックを停止し、PLL、HSI16 および HSE オシレータを無効にします。

CPU のディープスリープモードは、DBGMCU_CR レジスタに DBG_STOP または DBG_STANDBY ビットをセットすることで、デバッグのためにオーバライドすることができます。

STOP モード（Stop 0、Stop 1 またはSTANDBY）終了時のシステムクロックは HSI16 です。

Flashメモリプログラム操作が実行中の場合、Flashメモリインタフェースのアクセスが終了してから、STOPモード、STANDBY モードおよび SHUTDOWN モードに移行します。APB ドメインにアクセス中の場合、APB アクセスが終了してから、STOP モード、STANDBY モードおよび SHUTDOWN モードに移行します。

7.4 RCC レジスタ

7.4.1 クロック制御レジスタ（RCC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0063

HSEBYP は、リセットの影響を受けません。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	PLL RDY	PLLON	Res.	Res.	Res.	Res.	CSSON	HSEBYP	HSERDY	HSEON
						r	rw					rs	rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI RDY	HSI KERON	HSION	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
					r	rw	rw								

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **PLLRDY**：メイン PLL クロック･レディ･フラグ

ハードウェアによってセットされ、メイン PLL がロック状態であることを示します。

0：PLL が非ロック状態

1：PLL がロック状態

ビット 24 **PLLON**：メイン PLL イネーブル

メイン PLL を有効にするために、ソフトウェアによってセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに移行するときに、ハードウェアによってクリアされます。PLL クロックがシステムクロックとして使用されている場合、このビットをリセットできません。

0：PLL オフ

1：PLL オン

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CSSON**：クロック･セキュリティ･システム有効化

クロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。CSSON がセットされているとき、HSE オシレータがレディになるとクロック検出回路がハードウェアによって有効にされ、HSE クロックの障害が検出された場合ハードウェアによって無効にされます。このビットはセット専用で、リセットによってクリアされます。

0：クロック･セキュリティ･システム･オフ（クロック検出回路オフ）

1：クロック･セキュリティ･システム･オン（HSE オシレータが安定していればクロック検出回路オン、そうでなければオフ）

ビット 18 **HSEBYP**：HSE クリスタル･オシレータ･バイパス

外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。デバイスで使用するには、HSEON ビットをセットして外部クロックを有効にする必要があります。HSEBYP ビットは、HSE オシレータが無効のときのみ、書込みができます。

0：HSE クリスタル･オシレータはバイパスされません。

1：HSE クリスタル･オシレータが外部クロックによってバイパスされます。

ビット 17 **HSERDY**：HSE クロック･レディ･フラグ

HSE オシレータが安定していることを示すために、ハードウェアによってセットされます。

0：HSE オシレータはレディ状態ではありません。

1：HSE オシレータはレディ状態です。

注： HSEON ビットがクリアされると、HSERDY は、HSE クロックの 6 サイクル後にローになります。

ビット 16 **HSEON**：HSE クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに入るときに、HSE オシレータを停止するためにハードウェアによってクリアされます。HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合は、このビットをリセットできません。

0：HSE オシレータをオフ

1：HSE オシレータをオン

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSIRDY**：HSI16 クロック･レディ･フラグ

HSI16 オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは HSION をセットし、ソフトウェアによって HSI16 が有効になった場合のみセットされます。

0：HSI16 オシレータはレディ状態ではありません。

1：HSI16 オシレータはレディ状態です。

注： HSION ビットがクリアされると、HSIRDY は、HSI16 クロックの 6 サイクル後にローになります。

ビット 9 **HSIKERON**：HSI16 はペリフェラルカーネルに対して常に有効になります。

STOP モードでも HSI16 を強制的にオンにするために、ソフトウェアによってセット／クリアされます。HSI16 でカーネルクロックとして設定された場合、HSI16 だけが、USART および I^2C ペリフェラルに供給できます。STOP モードで HSI16 をオンのままにしておくと、HSI16 起動時間による通信速度の低下を回避できます。このビットは、HSION の値への影響はありません。

0：HSI16 オシレータへの影響はありません。

1：HSI16 オシレータは、STOP モードでも強制的にオンになります。

ビット 8 **HSION**：HSI16 クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに入るときに、HSI16 オシレータを停止するためにハードウェアによってクリアされます。

STOP モードを終了するときに STOPWUCK=1 または HSIASFS = 1 の場合、または HSE クリスタルオシレータに障害が発生した場合、HSI16 オシレータを強制的にオンにするために、ハードウェアによってセットされます。

このビットは、HSI16 が直接または間接的にシステムクロックとして使用されている場合に、ハードウェアによってセットされます。

0：HSI16 オシレータをオフ

1：HSI16 オシレータをオン

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

7.4.2 内部クロックソース較正レジスタ（RCC_ICSCR）

アドレスオフセット：0x04

リセット値：0x40XX：00XX

ここで、X は出荷時にプログラムされる値です。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:24 **HSITRIM[6:0]**：HSI16 クロックトリミング

これらのビットにより、ユーザプログラミング可能なトリミング値が使用でき、この値は HSICAL[7:0] ビットに加算されます。HSI16 の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。

デフォルト値は 64 で、HSICAL 値に追加すると、HSI16 を 16 MHz ± 1 % にトリミングします。

ビット 23:16 **HSICAL[7:0]**：HSI16 クロック較正

これらのビットは起動時に、出荷時にプログラムされた HSI16 較正トリミング値で初期化されます。HSITRIM が書き込まれると、HSICAL が HSITRIM と出荷時トリミング値の合計で更新されます。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

7.4.3 クロック設定レジスタ（RCC_CFGR）

アドレスオフセット：0x08

リセット値：0x0000 0005

アクセス：0 ≤ ウェイトステート ≤ 2、ワード、ハーフワード、およびバイトアクセス

クロックソースの切り替え中にアクセスが発生した場合に限り、1 または 2ウェイトステートが挿入されます。

APB または AHB プリスケーラ値の更新中にアクセスが発生した場合、0〜15 のウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCOPRE[2:0]			MCOSEL[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	rw	rw	rw	rw	rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	PPRE2 [2:0]			PPRE1 [2:0]			HPRE[3:0]				SWS[1:0]		SW[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	r	r	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **MCOPRE[2:0]**：マイクロコントローラ･クロック出力プリスケーラ

これらのビットは、ソフトウェアによってセット／クリアされます。

MCO 出力を有効にする前に、このプリスケーラを変更することを強く推奨します。

000：MCO を 1 分周

001：MCO を 2 分周

010：MCO を 4 分周

011：MCO を 8 分周

100：MCO を 16 分周

その他：設定禁止

ビット 27:24 **MCOSEL[3:0]**：マイクロコントローラ･クロック出力

ソフトウェアでセット／クリアされます。

0000：MCO 出力無効、MCO にクロックなし

0001：SYSCLK システム･クロックの選択

0010：予約済みであり、リセット値に保持する必要があります。

0011：HSI16 クロックの選択

0100：HSE クロックの選択

0101：メイン PLL クロックの選択

0110：LSI クロックの選択

0111：LSE クロックの選択

1000：内部 HSI48 クロックの選択

その他：予約済みです。

注： このクロック出力では、起動時または MCO クロック･ソースの切り替え時に、サイクルが切り捨てられる場合があります。

ビット 23:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:11 **PPRE2[2:0]**：APB2 プリスケーラ

APB2 クロック（PCLK2）の分周比を制御するために、ソフトウェアによってセット／クリアされます。

0xx：HCLK は分周されません。

100：HCLK は 2 分周されます。

101：HCLK は 4 分周されます。

110：HCLK は 8 分周されます。

111：HCLK は 16 分周されます。

ビット 10:8 **PPRE1[2:0]：**APB1 プリスケーラ

APB1 クロック（PCLK1）の分周比を制御するために、ソフトウェアによってセットまたはクリアされます。

0xx：HCLK は分周されません。

100：HCLK は 2 分周されます。

101：HCLK は 4 分周されます。

110：HCLK は 8 分周されます。

111：HCLK は 16 分周されます。

ビット 7:4 **HPRE[3:0]**：AHB プリスケーラ

AHB クロックの分周比を制御するために、ソフトウェアによってセット／クリアされます。

注意： デバイスの電圧レンジに応じて、ソフトウェアではこれらのビットを正しく設定し、システム周波数が最大許容周波数を超えないようにします（詳細については、セクション 6.1.5 : ダイナミック電圧スケーリングの管理を参照）。これらのビットの書込み動作後、電圧レンジを下げる前に、このレジスタを読み出して新しい値が正しく反映されていることを確認する必要があります。

0xxx：SYSCLK は分周されません。
1000：SYSCLK は 2 分周されます。
1001：SYSCLK は 4 分周されます。
1010：SYSCLK は 8 分周されます。
1011：SYSCLK は 16 分周されます。
1100：SYSCLK は 64 分周されます。
1101：SYSCLK は 128 分周されます。
1110：SYSCLK は 256 分周されます。
1111：SYSCLK は 512 分周されます。

ビット 3:2 **SWS[1:0]**：システム･クロック･スイッチ･ステータス

どのクロックソースがシステムクロックとして使用されているかを示すために、ハードウェアによってセット／クリアされます。
00：予約済みであり、リセット値に保持する必要があります。
01：HSI16 オシレータがシステム･クロックとして使用されています。
10：HSE がシステム･クロックとして使用されています。
11：PLL がシステム･クロックとして使用されています。

ビット 1:0 **SW[1:0]**：システム･クロック･スイッチ

システムクロックソース（SYSCLK）を選択するために、ソフトウェアによってセット／クリアされます。STOP モードと STANDBY モードの終了時、または HSE オシレータに障害が発生した場合に、HSI16 オシレータを強制的に選択するためにハードウェアによって設定されます。
00：予約済みであり、リセット値に保持する必要があります。
01：HSI16 がシステム･クロックとして選択されます。
10：HSE がシステム･クロックとして選択されます。
11：PLL がシステム･クロックとして選択されます。

7.4.4 PLL 設定レジスタ（RCC_PLLCFGR）

アドレスオフセット：0x0C

リセット値：0x0000 1000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタは、次の式に従って PLL クロック出力を設定するために使用します。

- f（VCO クロック）= f（PLL クロック入力）×（PLLN / PLLM）
- f（PLL_P）= f（VCO クロック）/ PLLP
- f（PLL_Q）= f（VCO クロック）/ PLLQ
- f（PLL_R）= f（VCO クロック）/ PLLR

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PLLPDIV[4:0]					PLLR[1:0]		PLL REN	Res.	PLLQ[1:0]		PLL QEN	Res.	Res.	PLLP	PLL PEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw			rw	rw

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

ビット 31:27 **PLLPDIV[4:0]**：メイン PLLP 分周比

PLL 「P」周波数を制御するために、ソフトウェアによってセット／クリアされます。PLL 「P」出力クロック周波数 = VCO 周波数 / PLLPDIV

00000：PLL 「P」クロック は PLLP ビットで制御されます。

00001：予約済み。

00010：PLL 「P」クロック = VCO / 2

...

11111：PLL 「P」クロック = VCO / 31

ビット 26:25 **PLLR[1:0]**：PLL 「R」クロック（システム･クロック）のためのメイン PLL 分周比

メイン PLL 出力クロック PLLCLK の周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、システムクロックで選択できます。これらのビットに書き込めるのは、PLL が無効なときだけです。

PLL 「R」出力クロック周波数 = VCO 周波数 / PLLR（PLLR = 2、4、6、または 8）

00：PLLR = 2

01：PLLR = 4

10：PLLR = 6

11：PLLR = 8

注意： このドメインで 170 MHz を超えないように、ソフトウェアでこれらのビットを正しくセットする必要があります。

ビット 24 **PLLREN**：PLL 「R」クロック出力有効化

PLL の PLL 「R」クロック出力（システムクロックとして使用）を有効化するために、ソフトウェアによってセット／リセットされます。

PLL の PLL 「R」クロック出力がシステムクロックとして使用されている場合は、このビットに書込みできません。

電力を削減するために、PLL の PLL 「R」クロック出力を使用しない場合は、PLLREN の値を 0 にする必要があります。

0：PLL 「R」クロック出力無効

1：PLL 「R」クロック出力有効化

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **PLLQ[1:0]**：PLL 「Q」クロックのためのメイン PLL 分周比

メイン PLL 出力クロック PLL 「Q」クロックの周波数を制御するために、ソフトウェアによってセット／クリアされます。この出力は、USB、RNG、SAI （48 MHz クロック)で選択できます。これらのビットに書き込めるのは、PLL が無効なときだけです。

PLL 「Q」出力クロック周波数 = VCO 周波数 / PLLQ（PLLQ = 2、4、6、または8）

00：PLLQ = 2

01：PLLQ = 4

10：PLLQ = 6

11：PLLQ = 8

注意： このドメインで 170 MHz を超えないように、ソフトウェアでこれらのビットを正しくセットする必要があります。

ビット 20 **PLLQEN**：メイン PLL 「Q」クロック出力有効化

PLL の PLL「Q」クロック出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、PLL の PLL 「Q」クロック出力を使用しない場合は、PLLQEN の値を 0 にする必要があります。

0：PLL「Q」クロック出力無効

1：PLL 「Q」クロック出力有効

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **PLLP**：PLL 「P」クロックのためのメイン PLL 分周比

メイン PLL 出力クロック PLL 「P」クロックの周波数を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL が無効なときだけです。

PLLPDIV[4:0] が “00000” に設定されている場合、PLL 「P」出力クロック周波数 = VCO 周波数 / PLLP（PLLP = 7 または 17）

0：PLLP = 7

1：PLLP = 17

注意： このドメインで 170 MHz を超えないように、ソフトウェアでこれらのビットを正しくセットする必要があります。

ビット 16 **PLLPEN**：メイン PLL の PLL 「P」クロック出力有効化

PLL の PLL「P」クロック出力を有効化するために、ソフトウェアによってセット／リセットされます。電力を削減するために、PLL の PLL 「P」クロック出力を使用しない場合は、PLLPEN の値を 0 にする必要があります。

0：PLL 「P」クロック出力無効

1：PLL 「P」クロック出力有効

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **PLLN[6:0]**：メイン PLL の VCO に対する逓倍数

VCO の逓倍数を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL が無効なときだけです。

VCO 出力周波数 = VCO 入力周波数 x PLLN（8 =< PLLN =< 127）

0000000：PLLN = 0、誤った設定

0000001：PLLN = 1、誤った設定

.....

0000111：PLLN = 7、誤った設定

0001000：PLLN = 8

0001001：PLLN = 9

.....

1111111：PLLN = 127

注意： VCO 出力周波数がデバイスのデータシートで規定された範囲内に確実に収まるように、これらのビットをソフトウェアで正しく設定する必要があります。

ビット 7:4 **PLLM[3:0]：**メイン PLL 入力クロックのための分周比

PLL 入力クロックを VCO の前で分周するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL がすべて無効なときだけです。

VCO 入力周波数 = PLL 入力クロック周波数 / PLLM（1 <= PLLM <= 16）

0000：PLLM = 1

0001：PLLM = 2

0010：PLLM = 3

0011：PLLM = 4

0100：PLLM = 5

0101：PLLM = 6

0110：PLLM = 7

0111：PLLM = 8

1000：PLLSYSM = 9

.....

1111：PLLSYSM = 16

注意： VCO 入力周波数がデバイスのデータシートで規定された範囲内に確実に収まるように、これらのビットをソフトウェアで正しく設定する必要があります。

ビット 3:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PLLSRC**[1:0]：メイン PLL 入力クロックソース

PLL クロックソースを選択するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL が無効なときだけです。

電力を削減するために、PLL を使用しない場合は、PLLSRC の値を 00 にする必要があります。

00：PLL にクロックが送信されません。

01：PLL にクロックが送信されません。

10：HSI16 クロックが PLL クロック入力としてとして選択されます。

11：HSE クロックが PLL クロック入力としてとして選択されます。

7.4.5 クロック割込み有効レジスタ（RCC_CIER）

アドレスオフセット：0x18

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48 RDYIE	LSE CSSIE	Res.	Res.	Res.	PLL RDYIE	HSE RDYIE	HSI RDYIE	Res.	LSE RDYIE	LSI RDYIE
					rw	rw				rw	rw	rw		rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYIE**：HSI48 レディ割込み有効化

内部 HSI48 オシレータによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI48 レディ割込みは無効です。

1：HSI48 レディ割込みは有効です。

ビット 9 **LSECSSIE**：LSE クロックセキュリティシステム割込み有効化

LSE のクロックセキュリティシステムによって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：LSE クロック障害によるクロック･セキュリティ割込みは無効です。

1：LSE クロック障害によるクロック･セキュリティ割込みは有効です。

ビット 8:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **PLLRDYIE**：PLL レディ割込みイネーブル

PLL ロックによって発生する割込みを有効または無効にするために、ソフトウェアによってセットまたはクリアします。

0：PLL ロック割込みは無効です。

1：PLL ロック割込みは有効です。

ビット 4 **HSERDYIE**：HSE レディ割込みイネーブル

HSE オシレータの安定化によって発生する割込みを有効または無効にするために、ソフトウェアによってセットまたはクリアします。

0：HSE レディ割込みは無効です。

1：HSE レディ割込みは有効です。

ビット 3 **HSIRDYIE**：HSI16 レディ割込み有効化

HSI16 オシレータの安定化によって発生する割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：HSI16 レディ割込みは無効です。

1：HSI16 レディ割込みは有効です。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSERDYIE**：LSE レディ割込みイネーブル

LSE オシレータの安定化によって発生する割込みを有効または無効にするために、ソフトウェアによってセットまたはクリアします。

0：LSE レディ割込みは無効です。

1：LSE レディ割込みは有効です。

ビット 0 **LSIRDYIE**：LSI レディ割込みイネーブル

LSI オシレータの安定化によって発生する割込みを有効または無効にするために、ソフトウェアによってセットまたはクリアします。

0：LSI レディ割込みは無効です。

1：LSI レディ割込みは有効です。

7.4.6 クロック割込みフラグ･レジスタ（RCC_CIFR）

アドレスオフセット：0x1C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48RDYF	LSECSSF	CSSF	Res.	Res.	PLLRDYF	HSERDYF	HSIRDYF	Res.	LSERDYF	LSIRDYF
					r	r	r			r	r	r		r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYF**：HSI48 レディ割込みフラグ

HSI48 クロックが安定し、HSI48ON（クロックリカバリ RC レジスタ（RCC_CRRCR）を参照）の設定に応じて HSI48RDYIE がセットされたときに、ハードウェアによってセットされます。
HSI48RDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSI48 オシレータによるクロック･レディ割込みは発生していません。
1：HSI48 オシレータによるクロック･レディ割込みが発生しました。

ビット 9 **LSECSSF**：LSE クロックセキュリティシステム割込みフラグ

LSE オシレータで障害が検出されたときに、ハードウェアによってセットされます。
LSECSSC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：LSE クロック障害によるクロック･セキュリティ割込みは発生していません。
1：LSE クロック障害によってクロック･セキュリティ割込みが発生しました。

ビット 8 **CSSF**：クロック･セキュリティ･システム割込みフラグ

HSE オシレータで障害が検出されたときに、ハードウェアによってセットされます。
CSSC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSE クロック障害によるクロック･セキュリティ割込みは発生していません。
1：HSE クロック障害によってクロック･セキュリティ割込みが発生しました。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **PLLRDYF**：PLL レディ割込みフラグ

PLL がロックされ、PLLRDYDIE がセットされているときに、ハードウェアによってセットされます。
PLLRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：PLL ロックによるクロック･レディ割込みは発生していません。
1：PLL ロックによるクロック･レディ割込みが発生しました。

ビット 4 **HSERDYF**：HSE レディ割込みフラグ

HSE クロックが安定し、HSERDYDIE がセットされているときに、ハードウェアによってセットされます。
HSERDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSE オシレータによるクロック･レディ割込みは発生していません。
1：HSE オシレータによるクロック･レディ割込みが発生しました。

ビット 3 **HSIRDYF**：HSI16 レディ割込みフラグ

HSI16 クロックが安定し、HSION（クロック制御レジスタ（RCC_CR）を参照）の設定に応じて HSIRDYDIE がセットされたときに、ハードウェアによってセットされます。HSION がセットされず、ただしクロックリクエストを通じてペリフェラルによって HSI16 オシレータが有効化された場合、このビットはセットされず、割込みも生成されません。
HSIRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：HSI16 オシレータによるクロック･レディ割込みは発生していません。
1：HSI16 オシレータによるクロック･レディ割込みが発生しました。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSERDYF**：LSE レディ割込みフラグ

LSE クロックが安定し、LSERDYDIE がセットされているときに、ハードウェアによってセットされます。
LSERDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：LSE オシレータによるクロック･レディ割込みは発生していません。
1：LSE オシレータによるクロック･レディ割込みが発生しました。

ビット 0 **LSIRDYF**：LSI レディ割込みフラグ

LSI クロックが安定し、LSIRDYDIE がセットされているときに、ハードウェアによってセットされます。
LSIRDYC ビットをセットすることによって、ソフトウェアによってクリアされます。
0：LSI オシレータによるクロック･レディ割込みは発生していません。
1：LSI オシレータによるクロック･レディ割込みが発生しました。

7.4.7 クロック割込みクリアレジスタ（RCC_CICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	HSI48 RDYC	LSE CSSC	CSSC	Res.	Res.	PLL RDYC	HSE RDYC	HSI RDYC	Res.	LSE RDYC	LSI RDYC
					w	w	w			w	w	w		w	w

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **HSI48RDYC**：HSI48 オシレータレディ割込みクリア

このビットは、HSI48RDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：HSI48RDYC フラグをクリアします。

ビット 9 **LSECSSC**：LSE クロックセキュリティシステム割込みクリア

このビットは、LSECSSF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：LSECSSF フラグをクリアします。

ビット 8 **CSSC**：クロック･セキュリティ･システム割込みクリア

このビットは、CSSF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：CSSF フラグをクリアします。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **PLLRDYC**：PLL レディ割込みクリア

このビットは、PLLRDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：PLLRDYF フラグをクリアします。

ビット 4 **HSERDYC**：HSE レディ割込みクリア

このビットは、HSERDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：HSERDYF フラグをクリアします。

ビット 3 **HSIRDYC**：HSI16 レディ割込みクリア

このビットは、HSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：HSIRDYF フラグをクリアします。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSERDYC**：LSE レディ割込みクリア

このビットは、LSERDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：LSERDYF がクリアされます。

ビット 0 **LSIRDYC**：LSI レディ割込みクリア

このビットは、LSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：LSIRDYF がクリアされます。

7.4.8 AHB1 ペリフェラルリセットレジスタ（RCC_AHB1RSTR）

アドレス･オフセット：0x28

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRC RST	Res.	Res.	Res.	FLASH RST	Res.	Res.	Res.	FMACR ST	CORDIC RST	DMAMUX1 RST	DMA2R ST	DMA1R ST
			rw				rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCRST**：CRC リセット

ソフトウェアでセットまたはクリアされます。

0：影響なし。

1：CRC をリセットします。

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FlashRST**：Flash メモリ･インタフェース･リセット

ソフトウェアでセット／クリアされます。このビットは、Flashメモリがパワーダウンモードの場合にのみ有効になります。

0：影響なし。

1：Flash メモリ･インタフェースをリセットします。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **FMACRST**：ソフトウェアでセット／クリアされます。

0：影響なし。

1：FMAC をリセットします。

ビット 3 **CORDICRST**：ソフトウェアでセット／クリアされます。

0：影響なし。

1：CORDIC をリセットします。

ビット 2 **DMAMUX1RST**：ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMAMUX1 をリセットします。

ビット 1 **DMA2RST**：DMA2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMA2 をリセットします。

ビット 0 **DMA1RST**：DMA1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DMA1 をリセットします。

7.4.9 AHB2 ペリフェラルリセットレジスタ（RCC_AHB2RSTR）

アドレス･オフセット：0x2C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	RNG RST	Res.	AES RST	Res.	Res.	Res.	Res.	DAC4 RST	DAC3 RST	DAC2 RST	DAC1 RST
					rw		rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ADC345 RST	ADC12 RST	Res.	Res.	Res.	Res.	Res.	Res.	GPIOGR ST	GPIOFR ST	GPIOER ST	GPIODR ST	GPIOCR ST	GPIOBR ST	GPIOA RST
	rw	rw							rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **RNGRST**：RNG リセット
ソフトウェアでセットまたはクリアされます。
0：影響なし。
1：RNG をリセットします。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **AESRST**：AESRST リセット
ソフトウェアでセットまたはクリアされます。
0：影響なし。
1：AES をリセットします。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DAC4RST**：DAC4 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DAC4 をリセットします。

ビット 18 **DAC3RST**：DAC3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DAC3 をリセットします。

ビット 17 **DAC2RST**：DAC2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DAC2 をリセットします。

ビット 16 **DAC1RST**：DAC1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：DAC1 をリセットします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ADC345RST**：ADC345 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：ADC345 をリセットします。

ビット 13 **ADC12RST**：ADC12 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：ADC12 インタフェースをリセットします。

ビット 12:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **GPIOGRST**：IO ポート G リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート G をリセットします。

ビット 5 **GPIOFRST**：IO ポート F リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート F をリセットします。

ビット 4 **GPIOERST**：IO ポート E リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート E をリセットします。

ビット 3 **GPIODRST**：IO ポート D リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート D をリセットします。

ビット 2 **GPIOCRST**：IO ポート C リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート C をリセットします。

ビット 1 **GPIOBRST**：IO ポート B リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート B をリセットします。

ビット 0 **GPIOARST**：IO ポート A リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：IO ポート A をリセットします。

7.4.10 AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR）

アドレス･オフセット：0x30

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **QSPIRST**：QUADSPI リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：QUADSPI をリセットします。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCRST**：フレキシブル･スタティック･メモリ･コントローラ･モジュールのリセット
ソフトウェアでセットまたはクリアされます。
0：影響なし。
1：FSMC をリセットします。

7.4.11 APB1 ペリフェラル･リセット･レジスタ 1（RCC_APB1RSTR1）

アドレス･オフセット：0x38

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 RST	I2C3 RST	Res.	PWR RST	Res.	Res.	FDCAN RST	Res.	USB RST	I2C2 RST	I2C1 RST	UART5 RST	UART4 RST	USART3 RST	USART2 RST	Res.
rw	rw		rw			rw		rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 RST	SPI2 RST	Res.	Res.	Res.	Res.	Res.	CRS RST	Res.	Res.	TIM7 RST	TIM6 RST	TIM5 RST	TIM4 RST	TIM3 RST	TIM2 RST
rw	rw						rw			rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1RST**：低電力タイマ 1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LPTIM1 をリセットします。

ビット 30 **I2C3RST**：I2C3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C3 インタフェースをリセットします。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **PWRRST**：電源インタフェース･リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：PWR をリセットします。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **FDCANRST**：FDCAN リセット
ソフトウェアでセット / クリアされます。
0：影響なし。
1：FDCAN をリセットします。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **USBRST**：USB デバイス･リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USB デバイスをリセットします。

ビット 22 **I2C2RST**：I2C2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C2 をリセットします。

ビット 21 **I2C1RST**：I2C1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C1 をリセットします。

ビット 20 **UART5RST**：UART5 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：UART5 をリセットします。

ビット 19 **UART4RST**：UART4 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：UART4 をリセットします。

ビット 18 **USART3RST**：USART3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART3 をリセットします。

ビット 17 **USART2RST**：USART2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART2 をリセットします。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3RST**：SPI3 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI3 をリセットします。

ビット 14 **SPI2RST**：SPI2 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI2 をリセットします。

ビット 13:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CRSRST**：CRS リセット
ソフトウェアでセットまたはクリアされます。
0：影響なし。
1：CRS をリセットします。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7RST**：TIM7 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM7 をリセットします。

ビット 4 **TIM6RST**：TIM6 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM7 をリセットします。

ビット 3 **TIM5RST**：TIM5 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM5 をリセットします。

ビット 2 **TIM4RST**：TIM3 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM3 をリセットします。

ビット 1 **TIM3RST**：TIM3 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM3 をリセットします。

ビット 0 **TIM2RST**：TIM2 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM2 をリセットします。

7.4.12 APB1 ペリフェラルリセットレジスタ 2（RCC_APB1RSTR2）

アドレス･オフセット：0x3C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1 RST	Res.	Res.	Res.	Res.	Res.	Res.	I2C4 RST	LP UART1 RST
							rw							rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UCPD1RST**：UCPD1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：UCPD1 をリセットします。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **I2C4RST**：I2C4 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：I2C4 をリセットします。

ビット 0 **LPUART1RST**：低電力 UART 1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：LPUART1 をリセットします。

7.4.13 APB2 ペリフェラルリセットレジスタ（RCC_APB2RSTR）

アドレス･オフセット：0x40

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	HRTIM1 RST	Res.	Res.	Res.	Res.	SAI1 RST	TIM20 RST	Res.	TIM17 RST	TIM16 RST	TIM15RST
					rw					rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI4 RST	USART1 RST	TIM8 RST	SPI1 RST	TIM1 RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYS CFG RST
rw	rw	rw	rw	rw											rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **HRTIM1RST**：HRTIM1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：HRTIM1 をリセットします。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI1RST**：シリアルオーディオインタフェース 1（SAI1）リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SAI1 をリセットします。

ビット 20 **TIM20RST**：TIM20 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM20 をリセットします。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17RST**：TIM17 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM17 タイマをリセットします。

ビット 17 **TIM16RST**：TIM16 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM16 タイマをリセットします。

ビット 16 **TIM15RST**：TIM15 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM15 タイマをリセットします。

ビット 15 **SPI4RST**：SPI4 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI4 をリセットします。

ビット 14 **USART1RST**：USART1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：USART1 をリセットします。

ビット 13 **TIM8RST**：TIM8 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM8 タイマをリセットします。

ビット 12 **SPI1RST**：SPI1 リセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：SPI1 をリセットします。

ビット 11 **TIM1RST**：TIM1 タイマリセット
ソフトウェアでセット／クリアされます。
0：影響なし。
1：TIM1 タイマをリセットします。

ビット 10:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGRST**：SYSCFG + COMP + OPAMP + VREFBUF リセット

0：影響なし。

1：SYSCFG + COMP + OPAMP + VREFBUF をリセットします。

7.4.14 AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1ENR）

アドレス･オフセット：0x48

リセット値：0x0000 0100

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	FLASHEN	Res.	Res.	Res.	FMACEN	CORDICEN	DMAMUX1EN	DMA2EN	DMA1EN
			rw				rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCEN**：CRC クロック有効化

ソフトウェアでセット／クリアされます。

0：CRC クロックを無効化

1：CRC クロックを有効化

ビット 11:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FlashEN**：Flash メモリ･インタフェース･クロック有効化

ソフトウェアでセット／クリアされます。このビットは、Flashがパワーダウンモードの場合にのみ無効にできます。

0：Flash メモリ･インタフェース･クロックを無効化

1：Flash メモリ･インタフェース･クロックを有効化

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **FMACEN**：FMAC 有効化

ソフトウェアでセット / クリアされます。

0：FMAC クロックを無効化

1：FMAC クロック有効

ビット 3 **CORDICEN**：CORDIC クロック有効化

ソフトウェアでセット／クリアされます。

0：CORDIC クロックを無効化

1：CORDIC クロック有効

ビット 2 **DMAMUX1EN**：DMAMUX1 クロック有効化

ソフトウェアでセット／クリアされます。

0：DMAMUX1 クロック無効

1：DMAMUX1 クロック有効

ビット 1 **DMA2EN**：DMA2 クロック有効化
ソフトウェアでセット／クリアされます。
0：DMA2 クロックを無効化
1：DMA2 クロックを有効化

ビット 0 **DMA1EN**：DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：DMA1 クロックを無効化
1：DMA1 クロックを有効化

7.4.15 AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2ENR）

アドレス･オフセット：0x4C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	RNG EN	Res.	AES EN	Res.	Res.	Res.	Res.	DAC4 EN	DAC3 EN	DAC2 EN	DAC1 EN
					rw		rw					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ADC345 EN	ADC12 EN	Res.	Res.	Res.	Res.	Res.	Res.	GPIOG EN	GPIOF EN	GPIOE EN	GPIOD EN	GPIOC EN	GPIOB EN	GPIOA EN
	rw	rw							rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **RNGEN**：RNG 有効化
ソフトウェアでセットまたはクリアされます。
0：RNG は無効です。
1：RNG は有効です。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **AESEN**：AES クロック有効化
ソフトウェアでセット／クリアされます。
0：AES クロックを無効化
1：AES クロック有効

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DAC4EN**：DAC4 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC4 クロック無効
1：DAC4 クロック有効

ビット 18 **DAC3EN**：DAC3 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC3 クロック無効
1：DAC3 クロック有効

ビット 17 **DAC2EN**：DAC2 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC2 クロック無効
1：DAC2 クロック有効

ビット 16 **DAC1EN**：DAC1 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC1 クロック無効
1：DAC1 クロック有効

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ADC345EN**：ADC345 クロック有効化
ソフトウェアでセット／クリアされます。
0：ADC345 クロック無効
1：ADC345 クロック有効

ビット 13 **ADC12EN**：ADC12 クロック有効化
ソフトウェアでセット／クリアされます。
0：ADC12 クロック無効
1：ADC12 クロック有効

ビット 12:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **GPIOGEN**：IO ポート G クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート G クロック無効
1：IO ポート G クロック有効

ビット 5 **GPIOFEN**：IO ポート F クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート F クロック無効
1：IO ポート F クロック有効

ビット 4 **GPIOEEN**：IO ポート E クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート E クロック無効
1：IO ポート E クロック有効

ビット 3 **GPIODEN**：IO ポート D クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート D クロック無効
1：IO ポート D クロック有効

ビット 2 **GPIOCEN**：IO ポート C クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート C クロック無効
1：IO ポート C クロック有効

ビット 1 **GPIOBEN**：IO ポート B クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート B クロック無効
1：IO ポート B クロック有効

ビット 0 **GPIOAEN**：IO ポート A クロック有効化
ソフトウェアでセット／クリアされます。
0：IO ポート A クロック無効
1：IO ポート A クロック有効

7.4.16 AHB3 ペリフェラル･クロック有効化レジスタ（RCC_AHB3ENR）

アドレス･オフセット：0x50

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **QSPIEN**：QUADSPI メモリ･インタフェース･クロック有効化
ソフトウェアでセット／クリアされます。
0：QUADSPI クロックは無効です。
1：QUADSPI クロックは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCEN**：フレキシブル･スタティック･メモリ･コントローラ･モジュール･クロック有効化
ソフトウェアでセットまたはクリアされます。
0：FSMC クロックを無効化
1：FSMC クロックを有効化

7.4.17 APB1 ペリフェラル･クロック有効化レジスタ 1（RCC_APB1ENR1）

アドレス：0x58

リセット値：0x0000 0400

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 EN	I2C3 EN	Res.	PWR EN	Res.	Res.	FDCAN EN	Res.	USB EN	I2C2 EN	I2C1 EN	UART5 EN	UART4 EN	USART3 EN	USART2 EN	Res.
rw	rw		rw			rw		rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 EN	SPI2 EN	Res.	Res.	WWDG EN	RTCAPB EN	Res.	CRS EN	Res.	Res.	TIM7 EN	TIM6 EN	TIM5 EN	TIM4 EN	TIM3 EN	TIM2 EN
rw	rw			rs	rw		rw			rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1EN**：低電力タイマ 1 クロック有効化
ソフトウェアでセット／クリアされます。
0：LPTIM1 クロック無効
1：LPTIM1 クロック有効

ビット 30 **I2C3EN**：I2C3 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C3 クロック無効
1：I2C3 クロック有効

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **PWREN**：電源インタフェース･クロック有効化
ソフトウェアでセット／クリアされます。
0：電源インタフェース･クロック無効
1：電源インタフェース･クロック有効

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **FDCANEN：**FDCAN クロック有効化
ソフトウェアでセット／クリアされます。
0：FDCAN クロックを無効化
1：FDCAN クロック有効

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **USBEN**：USB デバイス･クロック有効化
ソフトウェアでセット／クリアされます。
0：USB デバイス･クロック無効
1：USB デバイス･クロック有効

ビット 22 **I2C2EN**：I2C2 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C2 クロック無効
1：I2C2 クロック有効

ビット 21 **I2C1EN**：I2C1 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C1 クロック無効
1：I2C1 クロック有効

ビット 20 **UART5EN**：UART5 クロック有効化
ソフトウェアでセット／クリアされます。
0：UART5 クロック無効
1：UART5 クロック有効

ビット 19 **UART4EN**：UART4 クロック有効化
ソフトウェアでセット／クリアされます。
0：UART4 クロック無効
1：UART4 クロック有効

ビット 18 **USART3EN**：USART3 クロック有効化
ソフトウェアでセット／クリアされます。
0：USART3 クロック無効
1：USART3 クロック有効

ビット 17 **USART2EN**：USART2 クロック有効化
ソフトウェアでセット／クリアされます。
0：USART2 クロック無効
1：USART2 クロック有効

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3EN**：SPI3 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI3 クロック無効
1：SPI3 クロック有効

ビット 14 **SPI2EN**：SPI2 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI2 クロック無効
1：SPI2 クロック有効

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **WWDGEN**：ウィンドウ型ウォッチドッグ･クロック有効化
ウィンドウ型ウォッチドッグクロックを有効にするために、ソフトウェアによってセットされます。ハードウェアのシステムリセットによってリセットされます。このビットは、WWDG_SW オプションビットがリセットされたときに、ハードウェアでもセットできます。
0：ウィンドウ型ウォッチドッグ･クロック無効
1：ウィンドウ型ウォッチドッグ･クロック有効

ビット 10 **RTCAPBEN**：RTC APB クロック有効化
ソフトウェアでセット／クリアされます。
0：RTC APB クロック無効
1：RTC APB クロック有効

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CRSEN**：CRS リカバリ･システム･クロック有効化
ソフトウェアでセット／クリアされます。
0：CRS クロックを無効化
1：CRS クロック有効

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7EN**：TIM7 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM7 クロック無効
1：TIM7 クロック有効

ビット 4 **TIM6EN**：TIM6 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM6 クロック無効
1：TIM6 クロック有効

ビット 3 **TIM5EN**：TIM5 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM5 クロック無効
1：TIM5 クロック有効

ビット 2 **TIM4EN**：TIM4 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM4 クロック無効
1：TIM4 クロック有効

ビット 1 **TIM3EN**：TIM3 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM3 クロック無効
1：TIM3 クロック有効

ビット 0 **TIM2EN**：TIM2 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM2 クロック無効
1：TIM2 クロック有効

7.4.18 APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1ENR2）

アドレス･オフセット：0x5C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1 EN	Res.	Res.	Res.	Res.	Res.	Res.	I2C4EN	LP UART1 EN
							rw							rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UCPD1EN :** UCPD1 クロック有効化
ソフトウェアでセット／クリアされます。
0：UCPD1 クロックを無効化
1：UCPD1 クロック有効化

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **I2C4EN**：I2C4 クロック有効化
ソフトウェアでセット／クリアされます。
0：I2C4 クロック無効
1：I2C4 クロック有効

ビット 0 **LPUART1EN**：低電力 UART 1 クロック有効化
ソフトウェアでセット／クリアされます。
0：LPUART1 クロックを無効化
1：LPUART1 クロック有効化

7.4.19 APB2 ペリフェラルクロック有効レジスタ（RCC_APB2ENR）

アドレス：0x60

リセット値：0x0000 0000

アクセス：ワード、ハーフワード、およびバイトアクセス

注： ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの読出し／書込みアクセスはサポートされません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	HRTIM1 EN	Res.	Res.	Res.	Res.	SAI1 EN	TIM20 EN	Res.	TIM 17EN	TIM16 EN	TIM15 EN
					rw					rw	rw		rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI4 EN	USART1 EN	TIM8 EN	SPI1 EN	TIM1 EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYS CFGEN
rw	rw	rw	rw	rw											rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **HRTIM1EN**：HRTIM1 クロック有効化
ソフトウェアでセット／クリアされます。
0：HRTIM1 クロック無効
1：HRTIM1 クロック有効化

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI1EN**：SAI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SAI1 クロック無効
1：SAI1 クロック有効

ビット 20 **TIM20EN**：TIM20 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM20 クロック無効
1：TIM20 クロック有効

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17EN**：TIM17 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM17 タイマ･クロック無効
1：TIM17 タイマ･クロック有効

ビット 17 **TIM16EN**：TIM16 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM16 タイマ･クロック無効
1：TIM16 タイマ･クロック有効

ビット 16 **TIM15EN**：TIM15 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM15 タイマ･クロック無効
1：TIM15 タイマ･クロック有効

ビット 15 **SPI4EN**：SPI4 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI4 クロック無効
1：SPI4 クロック有効

ビット 14 **USART1EN**：USART1clock 有効化
ソフトウェアでセット／クリアされます。
0：USART1 クロック無効
1：USART1 クロック有効

ビット 13 T**IM8EN**：TIM8 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM8 タイマ･クロック無効
1：TIM8 タイマ･クロック有効

ビット 12 **SPI1EN**：SPI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SPI1 クロック無効
1：SPI1 クロック有効

ビット 11 **TIM1EN**：TIM1 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：TIM1 タイマ･クロック無効
1：TIM1P タイマ･クロック有効

ビット 10:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGEN**：SYSCFG + COMP + VREFBUF + OPAMP クロック有効化
ソフトウェアでセット／クリアされます。
0：SYSCFG + COMP + VREFBUF + OPAMP クロック無効
1：SYSCFG + COMP + VREFBUF + OPAMP クロック有効

7.4.20 SLEEP モードおよび STOP モードにおける AHB1 ペリフェラルクロック有効レジスタ（RCC_AHB1SMENR）

アドレス･オフセット：0x68

リセット値：0x0000：130F

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CRCSMEN	Res.	Res.	SRAM1SMEN	FLASHSMEN	Res.	Res.	Res.	FMACSMEN	CORDICSMEN	DMAMUX1SMEN	DMA2SMEN	DMA1SMEN
			rw			rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CRCSMEN**：SLEEP モードおよび STOP モード時 CRC クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、CRC クロックはクロック･ゲーティング(1) により無効になります。
1：SLEEP モードおよび STOP モード時、CRC クロックはクロック･ゲーティング (1) により有効になります。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SRAM1SMEN**：SLEEP モードおよび STOP モード時 SRAM1 インタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SRAM1 インタフェース･クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時 SRAM1 インタフェース･クロックはクロック･ゲーティング (1) により有効になります。

ビット 8 **FlashSMEN**：SLEEP モードおよび STOP モード時 Flash メモリ･インタフェース･クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、Flash メモリ･インタフェース･クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、Flash メモリ･インタフェース･クロックはクロック･ゲーティング (1) により有効になります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **FMACSMEN**：FMACSM クロックの有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、FMACSM クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、FMACSM クロックはクロック･ゲーティング (1) により有効になります。

ビット 3 **CORDICSMEN**：CORDICSM クロックの有効化
ソフトウェアでセット／クリアされます。
0：CORDICSM クロック無効
1：CORDICSM クロック有効

ビット 2 **DMAMUX1SMEN**：SLEEP モードおよび STOP モード時 DMAMUX1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DMAMUX1 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DMAMUX1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 1 **DMA2SMEN**：SLEEP モードおよび STOP モード時 DMA2 クロック有効化
SLEEP モード時、ソフトウェアによってセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DMA2 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DMA2 クロックはクロック･ゲーティング (1) により有効になります。

ビット 0 **DMA1SMEN**：SLEEP モードおよび STOP モード時 DMA1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、DMA1 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、DMA1 クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタでは、クロック･ソース自体ではなくクロック･ゲーティングのみを設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

7.4.21 SLEEP モードおよび STOP モードにおける AHB2 ペリフェラルクロック有効レジスタ（RCC_AHB2SMENR）

アドレス･オフセット：0x6C

リセット値：0x050F：667F

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	RNG EN	Res.	AESM EN	Res.	Res.	Res.	Res.	DAC4 SMEN	DAC3 SMEN	DAC2 SMEN	DAC1 SMEN
					rw		rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ADC345 SMEN	ADC12 SMEN	Res.	Res.	SRAM2S MEN	CCMSRAM SMEN	Res.	Res.	GPIOG SMEN	GPIOF SMEN	GPIOE SMEN	GPIOD SMEN	GPIOC SMEN	GPIOB SMEN	GPIOA SMEN
	rw	rw			rw	rw			rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **RNGEN**：RNG 有効化
ソフトウェアでセットまたはクリアされます。
0：RNG は無効です。
1：RNG は有効です。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **AESMEN**：AESM クロックの有効化
ソフトウェアでセット／クリアされます。
0：AESM クロック無効
1：AESM クロック有効

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **DAC4SMEN**：DAC4 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC4 クロック無効
1：SLEEP モードおよび STOP モード時 DAC4 クロック有効

ビット 18 **DAC3SMEN**：DAC3 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC3 クロック無効
1：SLEEP モードおよび STOP モード時 DAC3 クロック有効

ビット 17 **DAC2SMEN**：DAC2 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC2 クロック無効
1：SLEEP モードおよび STOP モード時 DAC2 クロック有効

ビット 16 **DAC1SMEN**：DAC1 クロック有効化
ソフトウェアでセット／クリアされます。
0：DAC1 クロック無効
1：SLEEP モードおよび STOP モード時 DAC1 クロック有効

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **ADC345SMEN**：ADC345 クロック有効化

ソフトウェアでセット／クリアされます。

0：ADC345 クロック無効

1：ADC345 クロック有効

ビット 13 **ADC12SMEN**：SLEEP モードおよび STOP モード時 ADC12 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、ADC12 クロックはクロック･ゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、ADC12 クロックはクロック･ゲーティング (1) により有効になります。

ビット 12:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **SRAM2SMEN**：SLEEP モードおよび STOP モード時 SRAM2 インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SRAM2 インタフェース･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時 SRAM2 インタフェース･クロックはクロック･ゲーティング (1) により有効になります。

ビット 9 **CCMSRAMSMEN**：SLEEP モードおよび STOP モード時 CCM SRAM インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CCM SRAM インタフェース･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、CCM SRAM インタフェース･クロックはクロック･ゲーティング (1) により有効になります。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **GPIOGSMEN**：SLEEP モードおよび STOP モード時 IO ポート G クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート G クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート G クロックはクロック･ゲーティング (1) により有効になります。

ビット 5 **GPIOFSMEN**：SLEEP モードおよび STOP モード時 IO ポート F クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート F クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート F クロックはクロック･ゲーティング (1) により有効になります。

ビット 4 **GPIOESMEN**：SLEEP モードおよび STOP モード時 IO ポート E クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート E クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート E クロックはクロック･ゲーティング (1) により有効になります。

ビット 3 **GPIODSMEN**：SLEEP モードおよび STOP モード時 IO ポート D クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、IO ポート D クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、IO ポート D クロックはクロック･ゲーティング (1) により有効になります。

ビット 2 **GPIOCSMEN**：SLEEP モードおよび STOP モード時 IO ポート C クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、IO ポート C クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、IO ポート C クロックはクロック･ゲーティング (1) により有効になります。

ビット 1 **GPIOBSMEN**：SLEEP モードおよび STOP モード時 IO ポート B クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、IO ポート B クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、IO ポート B クロックはクロック･ゲーティング (1) により有効になります。

ビット 0 **GPIOASMEN**：SLEEP モードおよび STOP モード時 IO ポート A クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、IO ポート A クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、IO ポート A クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタでは、クロック･ソース自体ではなくクロック･ゲーティングのみを設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

7.4.22 SLEEP モードおよび STOP モードにおける AHB3 ペリフェラル･クロック有効化レジスタ（RCC_AHB3SMENR）

アドレス･オフセット：0x70

リセット値：0x0000 0101

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **QSPISMEN**：SLEEP モードおよび STOP モード時 QUADSPI メモリインタフェースクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、QUADSPI クロックはクロック･ゲーティング(1) により無効になります。
1：SLEEP モードおよび STOP モード時、QUADSPI クロックはクロック･ゲーティング (1) により有効になります。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **FMCSMEN**：SLEEP モードおよび STOP モード時フレキシブル･スタティック･メモリ･コントローラ･クロック有効化

ソフトウェアでセットまたはクリアされます。

0：SLEEP モードおよび STOP モード時、FSMC クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、FSMC クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタで、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

7.4.23 SLEEP モードおよび STOP モードにおける APB1 ペリフェラル･クロック有効化レジスタ 1（RCC_APB1SMENR1）

アドレス：0x78

リセット値：0xD2FE：CD3F

アクセス：ノーウェイトステート、ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 SMEN	I2C3 SMEN	Res.	PWR SMEN	Res.	Res.	FDCAN SMEN	Res.	USB SMEN	I2C2 SMEN	I2C1 SMEN	UART5 SMEN	UART4 SMEN	USART3 SMEN	USART2 SMEN	Res.
rw	rw		rw			rw		rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI3 SMEN	SPI2 SMEN	Res.	Res.	WWDG SMEN	RTCAPB SMEN	Res.	CRS SMEN	Res.	Res.	TIM7 SMEN	TIM6 SMEN	TIM5 SMEN	TIM4 SMEN	TIM3 SMEN	TIM2 SMEN
rw	rw			rw	rw		rw			rw	rw	rw	rw	rw	rw

ビット 31 **LPTIM1SMEN**：SLEEP モードおよび STOP モード時低電力タイマ 1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、LPTIM1 クロックはクロック･ゲーティング(1)により無効になります。

1：SLEEP モードおよび STOP モード時、LPTIM1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 30 **I2C3SMEN**：SLEEP モードおよび STOP モード時 I2C3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C3 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C3 クロックはクロック･ゲーティング (1) により有効になります。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **PWRSMEN**：SLEEP モードおよび STOP モード時電源インタフェースクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、電源インタフェース･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、電源インタフェース･クロックはクロック･ゲーティング (1) により有効になります。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **FDCANSMEN**：SLEEP モードおよび STOP モード時 FDCAN クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、FDCAN クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、FDCAN クロックはクロック･ゲーティング (1) により有効になります。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **USBSMEN**：SLEEP モードおよび STOP モード時 USB デバイスクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USB デバイス･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USB デバイス･クロックはクロック･ゲーティング (1) により有効になります。

ビット 22 **I2C2SMEN**：SLEEP モードおよび STOP モード時 I2C2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C2 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C2 クロックはクロック･ゲーティング (1) により有効になります。

ビット 21 **I2C1SMEN**：SLEEP モードおよび STOP モード時 I2C1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C1 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 20 **UART5SMEN**：SLEEP モードおよび STOP モード時 UART5 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、UART5 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、UART5 クロックはクロック･ゲーティング (1) により有効になります。

ビット 19 **UART4SMEN**：SLEEP モードおよび STOP モード時 UART4 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、UART4 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、UART4 クロックはクロック･ゲーティング (1) により有効になります。

ビット 18 **USART3SMEN**：SLEEP モードおよび STOP モード時 USART3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USART3 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USART3 クロックはクロック･ゲーティング (1) により有効になります。

ビット 17 **USART2SMEN**：SLEEP モードおよび STOP モード時 USART2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、USART2 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、USART2 クロックはクロック･ゲーティング (1) により有効になります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **SPI3SMEN**：SLEEP モードおよび STOP モード時 SPI3 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SPI3 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SPI3 クロックはクロック･ゲーティング (1) により有効になります。

ビット 14 **SPI2SMEN**：SLEEP モードおよび STOP モード時 SPI2 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SPI2 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SPI2 クロックはクロック･ゲーティング (1) により有効になります。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **WWDGSMEN**：SLEEP モードおよび STOP モード時ウィンドウ型ウォッチドッグクロック有効化

ソフトウェアでセットまたはクリアされます。このビットは、ハードウェア WWDG オプションが有効である場合、ハードウェアで強制的に「1」になります。

0：SLEEP モードおよび STOP モード時、ウィンドウ型ウォッチドッグ･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、ウィンドウ型ウォッチドッグ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 10 **RTCAPBSMEN**：SLEEP モードおよび STOP モード時 RTC APB クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、RTC APB クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、RTC APB クロックはクロック･ゲーティング (1) により有効になります。

ビット 9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **CRSSMEN**：SLEEP モードおよび STOP モード時 CRS タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、CRS クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、CRS クロックはクロック･ゲーティング (1) により有効になります。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TIM7SMEN**：SLEEP モードおよび STOP モード時 TIM7 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM7 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM7 クロックはクロック･ゲーティング (1) により有効になります。

ビット 4 **TIM6SMEN**：SLEEP モードおよび STOP モード時 TIM6 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM6 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM6 クロックはクロック･ゲーティング (1) により有効になります。

ビット 3 **TIM5SMEN**：SLEEP モードおよび STOP モード時 TIM5 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM5 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM5 クロックはクロック･ゲーティング (1) により有効になります。

ビット 2 **TIM4SMEN**：SLEEP モードおよび STOP モード時 TIM4 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM4 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM4 クロックはクロック･ゲーティング (1) により有効になります。

ビット 1 **TIM3SMEN**：SLEEP モードおよび STOP モード時 TIM3 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM3 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM3 クロックはクロック･ゲーティング (1) により有効になります。

ビット 0 **TIM2SMEN**：SLEEP モードおよび STOP モード時 TIM2 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM2 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM2 クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタでは、クロック･ソース自体ではなくクロック･ゲーティングのみを設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

7.4.24 SLEEP モードおよび STOP モードにおける APB1 ペリフェラルクロック有効レジスタ 2（RCC_APB1SMENR2）

アドレス･オフセット：0x7C

リセット値：0x0000 0103

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1 SMEN	Res.	Res.	Res.	Res.	Res.	Res.	I2C4 SMEN	LP UART1 SMEN
							rw							rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UCPD1SMEN**：SLEEP モードおよび STOP モード時 UCPD1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、UCPD1 クロックはクロック･ゲーティング(1) により無効になります。

1：SLEEP モードおよび STOP モード時、UCPD1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **I2C4SMEN**：SLEEP モードおよび STOP モード時 I2C4 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、I2C4 クロックはクロック･ゲーティング(2) により無効になります。

1：SLEEP モードおよび STOP モード時、I2C4 クロックはクロック･ゲーティング (1) により有効になります。

ビット 0 **LPUART1SMEN**：SLEEP モードおよび STOP モード時低電力 UART 1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、LPUART1 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、LPUART1 クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタでは、クロック･ソース自体ではなくクロック･ゲーティングのみを設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。
2. このレジスタは、クロックソース自体ではなく、クロックゲーティングのみ設定します。ほとんどのペリフェラルは、STOP モードで必ず無効になる単一のクロック（AHB クロックまたは APB クロック）で駆動されています。この場合、STOP モードではこのビットは効果がありません。

7.4.25 SLEEP モードおよび STOP モードにおける APB2 ペリフェラルクロック有効レジスタ（RCC_APB2SMENR）

アドレス：0x80

リセット値：0x0437：F801

アクセス：ワード、ハーフワード、バイト･アクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	HRTIM1 SMEN	Res.	Res.	Res.	Res.	SAI1 SMEN	TIM20 SMEN	Res.	TIM17 SMEN	TIM16 SMEN	TIM15 SMEN
					rw					rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPI4 SMEN	USART1 SMEN	TIM8 SMEN	SPI1 SMEN	TIM1 SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYS CFG SMEN
rw	rw	rw	rw	rw											rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **HRTIM1SMEN**：SLEEP モードおよび STOP モード時 HRTIM1 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、HRTIM1 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、HRTIM1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 25:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SAI1SMEN**：SLEEP モードおよび STOP モード時 SAI1 クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SAI1 クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SAI1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 20 **TIM20SMEN**：SLEEP モードおよび STOP モード時 TIM20 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM20 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM20 クロックはクロック･ゲーティング (1) により有効になります。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TIM17SMEN**：SLEEP モードおよび STOP モード時 TIM17 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM17 タイマ･クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM17 タイマ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 17 **TIM16SMEN**：SLEEP モードおよび STOP モード時 TIM16 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM16 タイマ･クロックはクロック･ゲーティング(1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM16 タイマ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 16 **TIM15SMEN**：SLEEP モードおよび STOP モード時 TIM15 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM15 タイマ･クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM15 タイマ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 15 **SPI4SMEN**：SLEEP モードおよび STOP モード時 SPI4 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SPI4 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SPI4 クロックはクロック･ゲーティング (1) により有効になります。

ビット 14 **USART1SMEN**：SLEEP モードおよび STOP モード時 USART1クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、USART1 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、USART1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 13 T**IM8SMEN**：SLEEP モードおよび STOP モード時 TIM8 タイマクロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、TIM8 タイマ･クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、TIM8 タイマ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 12 **SPI1SMEN**：SLEEP モードおよび STOP モード時 SPI1 クロック有効化
ソフトウェアでセット／クリアされます。
0：SLEEP モードおよび STOP モード時、SPI1 クロックはクロック･ゲーティング (1) により無効になります。
1：SLEEP モードおよび STOP モード時、SPI1 クロックはクロック･ゲーティング (1) により有効になります。

ビット 11 **TIM1SMEN**：SLEEP モードおよび STOP モード時 TIM1 タイマクロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、TIM1 タイマ･クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、TIM1P タイマ･クロックはクロック･ゲーティング (1) により有効になります。

ビット 10:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SYSCFGSMEN**：SLEEP モードおよび STOP モード時 SYSCFG + COMP + VREFBUF + OPAMP クロック有効化

ソフトウェアでセット／クリアされます。

0：SLEEP モードおよび STOP モード時、SYSCFG + COMP + VREFBUF + OPAMP クロックはクロック･ゲーティング (1) により無効になります。

1：SLEEP モードおよび STOP モード時、SYSCFG + COMP + VREFBUF + OPAMP クロックはクロック･ゲーティング (1) により有効になります。

1. このレジスタでは、クロック･ソース自体ではなくクロック･ゲーティングのみを設定します。ほとんどのペリフェラルは単一のクロック（AHB または APB クロック）で供給されており、STOP モードでは常に無効にされています。この場合、STOP モードではこのビットは効果がありません。

7.4.26 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR）

アドレス：0x88

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC345SEL [1:0]		ADC12SEL [1:0]		CLK48SEL [1:0]		FDCANSEL[1:0]		I2S23SEL[1:0]		SAI1SEL [1:0]		LPTIM1SEL[1:0		I2C3SEL[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C2SEL [1:0]		I2C1SEL [1:0]		LPUART1SEL [1:0]		UART5SEL [1:0]		UART4SEL [1:0]		USART3SEL [1:0]		USART2SEL [1:0]		USART1SEL [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 **ADC345SEL[1:0]**：ADC3/4/5 クロックソースの選択

これらのビットは、ADC345 インタフェースで使用されるクロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：どのクロックも選択されません。

01：PLL 「P」クロックが ADC3/4/5 クロックとして選択されます。

10：システム･クロックが ADC3/4/5 クロックとして選択されます。

11：予約済み。

ビット 29:28 **ADC12SEL[1:0]**：ADC1/2 クロックソースの選択

これらのビットは、ADC インタフェースで使用されるクロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：どのクロックも選択されません。

01：PLL 「P」クロックが ADC1/2 クロックとして選択されます。

10：システム･クロックが ADC1/2 クロックとして選択されます。

11：予約済みです。

ビット 27:26 **CLK48SEL[1:0]**：48MHz クロックソース選択

これらのビットは、USB デバイス FS および RNGで使用される 48 MHz クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：HSI48 クロックが 48 MHz クロックとして選択されます。

01：予約済みです。

10：PLL 「Q」クロック（PLL48M1CLK）が 48 MHz クロックとして選択されます。

11：予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **FDCANSEL[1:0]：**クロック･ソース選択

これらのビットは、FDCAN クロック･ソースを選択するために、ソフトウェアによってセット / クリアされます。

00：HSE クロックが FDCAN クロックとして選択されます。

01：PLL 「Q」クロックが FDCAN クロックとして選択されます。

10：PCLK クロックが FDCAN クロックとして選択されます。

11：予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **I2S23SEL[1:0]：**クロック･ソース選択

これらのビットは、I2S23 クロック･ソースを選択するために、ソフトウェアによってセット / クリアされます。

00：システム･クロックが I2S23 クロックとして選択されます。

01：PLL 「Q」クロックが I2S23 クロックとして選択されます。

10：I2S_CKIN ピンに供給されるクロックが I2S23 クロックとして選択されます。

11：HSI16 クロックが I2S23 クロックとして選択されます。

ビット 21:20 **SAI1SEL[1:0]：**クロック･ソース選択

これらのビットは、SAI クロック･ソースを選択するために、ソフトウェアによってセット / クリアされます。

00：システム･クロックが SAI クロックとして選択されます。

01：PLL 「Q」クロックが SAI クロックとして選択されます。

10：I2S_CKIN ピンに供給されるクロックが SAI クロックとして選択されます。

11：HSI16 クロックが SAI クロックとして選択されます。

ビット 19:18 **LPTIM1SEL[1:0]**：低電力タイマ 1 クロックソース選択

これらのビットは、LPTIM1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が LPTIM1 クロックとして選択されます。

01：LSI クロックが LPTIM1 クロックとして選択されます。

10：HSI16 クロックが LPTIM1 クロックとして選択されます。

11：LSE クロックが LPTIM1 クロックとして選択されます。

ビット 17:16 **I2C3SEL[1:0]**：I2C3 クロックソースの選択

これらのビットは、I2C3 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C3 クロックとして選択されます。

01：システム･クロック（SYSCLK）が I2C3 クロックとして選択されます。

10：HSI16 クロックが I2C3 クロックとして選択されます。

11：予約済みです。

ビット 15:14 **I2C2SEL[1:0]**：I2C2 クロックソースの選択

これらのビットは、I2C2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C2 クロックとして選択されます。

01：システム･クロック（SYSCLK）が I2C2 クロックとして選択されます。

10：HSI16 クロックが I2C2 クロックとして選択されます。

11：予約済みです。

ビット 13:12 **I2C1SEL[1:0]**：I2C1 クロックソースの選択

これらのビットは、I2C1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が I2C1 クロックとして選択されます。

01：システム･クロック（SYSCLK）が I2C1 クロックとして選択されます。

10：HSI16 クロックが I2C1 クロックとして選択されます。

11：予約済みです。

ビット 11:10 **LPUART1SEL[1:0]**：LPUART1 クロックソースの選択

これらのビットは、LPUART1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が LPUART1 クロックとして選択されます。

01：システム･クロック（SYSCLK）が LPUART1 クロックとして選択されます。

10：HSI16 クロックが LPUART1 クロックとして選択されます。

11：LSE クロックが LPUART1 クロックとして選択されます。

ビット 9:8 **UART5SEL[1:0]**：UART5 クロックソースの選択

これらのビットは、UART5 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が UART5 クロックとして選択されます。

01：システム･クロック（SYSCLK）が UART5 クロックとして選択されます。

10：HSI16 クロックが UART5 クロックとして選択されます。

11：LSE クロックが UART5 クロックとして選択されます。

ビット 7:6 **UART4SEL[1:0]**：UART4 クロックソースの選択

このビットは、UART4 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が UART4 クロックとして選択されます。

01：システム･クロック（SYSCLK）が UART4 クロックとして選択されます。

10：HSI16 クロックが UART4 クロックとして選択されます。

11：LSE クロックが UART4 クロックとして選択されます。

ビット 5:4 **USART3SEL[1:0]**：USART3 クロックソースの選択

このビットは、USART3 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART3 クロックとして選択されます。

01：システム･クロック（SYSCLK）が USART3 クロックとして選択されます。

10：HSI16 クロックが USART3 クロックとして選択されます。

11：LSE クロックが USART3 クロックとして選択されます。

ビット 3:2 **USART2SEL[1:0]**：USART2 クロックソースの選択

このビットは、USART2 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART2 クロックとして選択されます。

01：システム･クロック（SYSCLK）が USART2 クロックとして選択されます。

10：HSI16 クロックが USART2 クロックとして選択されます。

11：LSE クロックが USART2 クロックとして選択されます。

ビット 1:0 **USART1SEL[1:0]**：USART1 クロックソースの選択

このビットは、USART1 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。

00：PCLK が USART1 クロックとして選択されます。

01：システム･クロック（SYSCLK）が USART1 クロックとして選択されます。

10：HSI16 クロックが USART1 クロックとして選択されます。

11：LSE クロックが USART1 クロックとして選択されます。

7.4.27 RTC ドメイン制御レジスタ（RCC_BDCR）

アドレスオフセット：0x90

リセット値：0x0000 0000

RTC ドメインのパワーオンリセットでのみリセットされる LSCOSEL、LSCOEN および BDRST を除き、RTC ドメインリセットでリセットされます。
アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス
このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

注： **RTC ドメイン制御レジスタ（RCC_BDCR）のビットは、V_{CORE} ドメイン外にあります。このため、リセット後、これらのビットは書込み保護されるので、これらを変更するには、セクション 6.4.1：電源制御レジスタ 1（PWR_CR1）の DBP ビットをセットする必要があります。詳細については、セクション 6.1.3：226 ページのバッテリバックアップドメインを参照してください。これらのビット（LSCOSEL、LSCOEN および BDRST を除く）は、RTC ドメインリセット後にのみリセットされます（セクション 7.1.3：RTC ドメインリセットを参照）。内部または外部リセットは、これらのビットに影響しません。**

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL[1:0]		Res.	LSECSSD	LSECSSON	LSEDRV[1:0]		LSEBYP	LSERDY	LSEON
rw						rw	rw		r	rw	rw	rw	rw	r	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **LSCOSEL**：低速クロック出力選択
ソフトウェアでセット／クリアされます。
0：LSI クロックの選択
1：LSE クロックの選択

ビット 24 **LSCOEN**：低速クロック出力有効化
ソフトウェアでセット／クリアされます。
0：低速クロック出力（LSCO）は無効です。
1：低速クロック出力（LSCO）は有効です。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **BDRST**：RTC ドメイン･ソフトウェア･リセット
ソフトウェアでセット／クリアされます。
0：リセットは有効ではありません。
1：RTC ドメイン全体をリセットします。

ビット 15 **RTCEN**：RTC クロック有効化
ソフトウェアでセット／クリアされます。
0：RTC クロックを無効化
1：RTC クロック有効

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RTCSEL[1:0]**：RTC クロック･ソース選択

RTC のクロック･ソースを選択するために、ソフトウェアによってセットされます。RTC クロックソースが選択されると、RTC ドメインがリセットされた場合や、LSE で障害が検出された（LSECSSD がセットされている）場合以外は変更できません。BDRST ビットを使用して、リセットできます。

00：クロックなし。

01：LSE オシレータ･クロックが RTC クロックとして使用されます。

10：LSI オシレータ･クロックが RTC クロックとして使用されます。

11：32 分周された HSE オシレータ･クロックが RTC クロックとして使用されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LSECSSD**：LSE の CSS による障害の検出

外部 32 kHz オシレータ（LSE）のクロックセキュリティシステムによって障害が検出されたときに、ハードウェアによってセットされます。

0：LSE（32 kHz オシレータ）の障害は検出されていません。

1：LSE（32 kHz オシレータ）の障害が検出されました。

ビット 5 **LSECSSON**：LSE の CSS 有効化

LSE（32 kHz オシレータ）のクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。

LSECSSON は、LSE オシレータを有効（LSEON ビットを有効化）かつレディ状態（ハードウェアで LSERDY フラグをセット）にし、RTCSEL ビットを選択した後で、有効にする必要があります。

このビットは、一度有効にすると、LSE の障害検出（LSECSSD =1）の後を除き、

無効にすることはできません。この場合、LSECSSON ビットをソフトウェアで無効にする必要があります。

0：LSE （32 kHz 外部オシレータ）の CSS オフ

1：LSE （32 kHz 外部オシレータ）の CSS オン

ビット 4:3 **LSEDRV[1:0]**：LSE オシレータの駆動能力

LSE オシレータの駆動能力を変調するためにソフトウェアによってセットされます。

00：「Xtal モード」低駆動

01：「Xtal モード」中低駆動

10：「Xtal モード」中高駆動

11：「Xtal モード」高駆動

オシレータは、バイパスモードでない場合、Xtal モードです。

ビット 2 **LSEBYP**：LSE オシレータ･バイパス

デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。このビットに書き込めるのは、外部 32 kHz オシレータが無効なときだけです（LSEON=0 および LSERDY=0）。

0：LSE オシレータはバイパスされません。

1：LSE オシレータはバイパスされます。

ビット 1 **LSERDY**：LSE オシレータのレディ状態

外部 32 kHz オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSEON ビットがクリアされた後、LSERDY は、外部低速オシレータクロックの 6 サイクル後にローになります。

0：LSE オシレータはレディ状態ではありません。

1：LSE オシレータはレディ状態です。

ビット 0 **LSEON**：LSE オシレータ･イネーブル

ソフトウェアでセットまたはクリアされます。

0：LSE オシレータをオフ

1：LSE オシレータオン

7.4.28 制御／ステータスレジスタ（RCC_CSR）

アドレス：0x94

リセット値：0x0C00 0000

システムリセットによってリセットされます。ただし、リセットフラグは電源リセット時のみリセットされます。

アクセス：0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWR RSTF	WWDG RSTF	IWDG RSTF	SFT RSTF	BOR RSTF	PIN RSTF	OBL RSTF	Res.	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	r	r	r	r	r		rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSI RDY	LSION
														r	rw

ビット 31 **LPWRRSTF**：低電力リセットフラグ

STOP、STANDBY または SHUTDOWN モードへの不正な移行によってリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：不正なモード･リセットは発生していません。
1：不正なモード･リセットが発生しました。

ビット 30 **WWDGRSTF**：ウィンドウ型ウォッチドッグリセットフラグ

ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：ウィンドウ型ウォッチドッグ･リセットは発生していません。
1：ウィンドウ型ウォッチドッグ･リセットが発生しました。

ビット 29 **IWDGRSTF**：独立型ウィンドウウォッチドッグリセットフラグ

独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：独立型ウォッチドッグ･リセットは発生していません。
1：独立型ウォッチドッグ･リセットが発生しました。

ビット 28 **SFTRSTF**：ソフトウェアリセットフラグ

ソフトウェアリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：ソフトウェア･リセットは発生していません。
1：ソフトウェア･リセットが発生しました。

ビット 27 **BORRSTF**：BOR フラグ

BOR が発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むことによってクリアされます。
0：BOR は発生していません。
1：BOR が発生しました。

ビット 26 **PINRSTF**：ピンリセットフラグ

NRST ピンに外部リセット信号が入力され、リセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むことによってクリアされます。

0：NRST ピンからのリセットは発生していません。

1：NRST ピンからリセットが発生しました。

ビット 25 **OBLRSTF**：オプションバイトローダリセットフラグ

オプションバイトローディングからリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むことによってクリアされます。

0：オプション･バイト･ローディングからのリセットは発生していません。

1：オプション･バイト･ローディングからのリセットが発生しました。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **RMVF**：リセット･フラグ解除

リセットフラグをクリアするために、ソフトウェアによってセットされます。

0：影響なし。

1：リセット･フラグをクリアします。

ビット 22:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **LSIRDY**：LSI オシレータのレディ状態

LSI オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSION ビットがクリアされた後、LSIRDY は、LSI オシレータクロックの 3 サイクル後にローになります。このビットは、LSI が LSE のクロックセキュリティシステムによってリクエストされたときに LSION = 0 であっても、独立型ウォッチドッグまたは RTC でセットできます。

0：LSI オシレータはレディ状態ではありません。

1：LSI オシレータはレディ状態です。

ビット 0 **LSION**：LSI オシレータ･イネーブル

ソフトウェアでセットまたはクリアされます。

0：LSI オシレータをオフ

1：LSI オシレータをオン

7.4.29 クロックリカバリ RC レジスタ（RCC_CRRCR）

アドレス：0x98

リセット値：0x0000：XXX0

ここで、X は出荷時にプログラムされる値です。

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HSI48CAL [8:0]									Res.	Res.	Res.	Res.	Res.	HSI48 RDY	HSI48 オン
r	r	r	r	r	r	r	r	r						r	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:7 **HSI48CAL[8:0]**：HSI48 クロック較正

これらのビットは起動時に、出荷時にプログラムされた HSI48 較正トリミング値で初期化されます。これらは読出し専用です。

ビット 6:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HSI48RDY**：HSI48 クロックレディフラグ

HSI48 オシレータが安定していることを示すために、ハードウェアによってセットされます。このビットは、HSI48 が有効になった場合のみソフトウェアによってHSI48ON をセットします。

0：HSI48 オシレータはレディ状態ではありません。

1：HSI48 オシレータはレディ状態です。

ビット 0 **HSI48ON**：HSI48 クロック有効化

ソフトウェアでセット／クリアされます。

STOP、STANDBY または SHUTDOWN モードに移行するときに、HSI48 を停止するためにハードウェアによってクリアされます。

0：HSI48 オシレータをオフ

1：HSI48 オシレータをオン

7.4.30 ペリフェラル独立クロック設定レジスタ（RCC_CCIPR2）

アドレス：0x9C

リセット値：0x0000 0000

アクセス：ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPISEL [1:0]		Res.	Res.	Res.	Res.
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C4SEL [1:0]	
														rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **QSPISEL[1:0]**：QUADSPI クロックソース選択
ソフトウェアでセット／クリアされます。
00：システム･クロックが QUADSPI カーネル･クロックとして選択されます。
01：HSI16 クロックが QUADSPI カーネル･クロックとして選択されます。
10：PLL 「Q」クロックが QUADSPI カーネル･クロックとして選択されます。
11：予約済み

ビット 19:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **I2C4SEL[1:0]**：I2C4 クロックソースの選択
これらのビットは、I2C4 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。
00：PCLK が I2C4 クロックとして選択されます。
01：システム･クロック（SYSCLK）が I2C4 クロックとして選択されます。
10：HSI16 クロックが I2C4 クロックとして選択されます。
11：予約済み

7.4.31 RCC レジスタマップ

次の表に、RCC レジスタマップとリセット値を示します。

表 51. RCC レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RCC_CR	Res.	Res.	Res.	Res.	Res.	Res.	PLLRDY	PLLON	Res.	Res.	Res.	Res.	CSSON	HSEBYP	HSERDY	HSEON	Res.	Res.	Res.	Res.	Res.	HSIRDY	HSIKERON	HSION	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値			0	0	0	0	0	0					0	0	0	0						0	0	0								
0x04	RCC_ICSCR	Res.	HSITRIM[6:0]							HSICAL[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値		1	0	0	0	0	0	0	X	X	X	X	X	X	X	X																
0x08	RCC_CFGR	Res.	MCOPRE [2:0]			MCOSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PPRE2 [2:0]			PPRE1 [2:0]			HPRE[3:0]				SWS [1:0]		SW [1:0]	
	リセット値		0	0	0	0	0	0	0											0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	RCC_PLL CFGR	PLLPDIV[4:0]					PLLR [1:0]		PLLREN	Res.	PLLQ [1:0]		PLLQEN	Res.	Res.	PLLP	PLLPEN	Res.	PLLN [6:0]							PLLM [3:0]				Res.	Res.	PLL SRC [1:0]	
	リセット値	0	0	0	0	0	0	0	0		0	0	0			0	0		0	0	1	0	0	0	0	0	0	0	0			0	0
0x18	RCC_CIER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYIE	LSECSSIE	Res.	Res.	Res.	PLLRDYIE	HSERDYIE	HSIRDYIE	Res.	LSERDYIE	LSIRDYIE
	リセット値																						0	0				0	0	0	0	0	0
0x1C	RCC_CIFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYF	LSECSSF	CSSF	Res.	Res.	PLLRDYF	HSERDYF	HSIRDYF	Res.	LSERDYF	LSIRDYF
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x20	RCC_CICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48RDYC	LSECSSC	CSSC	Res.	Res.	PLLRDYC	HSERDYC	HSIRDYC	Res.	LSERDYC	LSIRDYC
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x28	RCC_ AHB1RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCRST	Res.	Res.	Res.	FlashRST	Res.	Res.	Res.	FMACRST	CORDICRST	DMAMUX1RST	DMA2RST	DMA1RST
	リセット値																				0				0				0	0	0	0	0
0x2C	RCC_ AHB2RSTR	Res.	Res.	Res.	Res.	Res.	RNGRST	Res.	AESRST	Res.	Res.	Res.	Res.	DAC4RST	DAC3RST	DAC2RST	DAC1RST	Res.	ADC345RST	ADC12RST	Res.	Res.	Res.	Res.	Res.	Res.	GPIOGRST	GPIOFRST	GPIOERST	GPIODRST	GPIOCRST	GPIOBRST	GPIOARST
	リセット値						0		0					0	0	0	0		0	0							0	0	0	0	0	0	0

表 51. RCC レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	RCC_ AHB3RSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPIRST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCRST
	リセット値																								0								0
0x38	RCC_ APB1RSTR1	LPTIM1RST	I2C3RST	Res.	PWRRST	Res.	Res.	FDCANRST	Res.	USBRST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	Res.	SPI3RST	SPI2RST	Res.	Res.	Res.	Res.	Res.	CRSRST	Res.	Res.	TIM7RST	TIM6RST	TIM5RST	TIM4RST	TIM3RST	TIM2RST
	リセット値	0	0		0			0		0	0	0	0	0	0	0		0	0						0			0	0	0	0	0	0
0x3C	RCC_ APB1RSTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1RST	Res.	Res.	Res.	Res.	Res.	Res.	I2C4RST	LPUART1RST
	リセット値																								0							0	0
0x40	RCC_ APB2RSTR	Res.	Res.	Res.	Res.	Res.	HRTIMRST	Res.	Res.	Res.	Res.	SAI1RST	TIM20RST	Res.	TIM17RST	TIM16RST	TIM15RST	SPI4RST	USART1RST	TIM8RST	SPI1RST	TIM1RST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGRST
	リセット値						0					0	0		0	0	0	0	0	0	0	0												0
0x48	RCC_AHB1 ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCEN	Res.	Res.	Res.	FlashEN	Res.	Res.	Res.	FMACEN	CORDICEN	DMAMUX1EN	DMA2EN	DMA1EN
	リセット値																				0				1				0	0	0	0	0
0x4C	RCC_AHB2 ENR	Res.	Res.	Res.	Res.	Res.	RNGEN	Res.	AESEN	Res.	Res.	Res.	Res.	DAC4EN	DAC3EN	DAC2EN	DAC1EN	Res.	ADC345EN	ADC12EN	Res.	Res.	Res.	Res.	Res.	Res.	GPIOGEN	GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
	リセット値						0		0					0	0	0	0		0	0							0	0	0	0	0	0	0	0
0x50	RCC_AHB3 ENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPIEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCEN
	リセット値																								0								0
0x58	RCC_ APB1ENR1	LPTIM1EN	I2C3EN	Res.	PWREN	Res.	Res.	FDCANEN	Res.	USBEN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	Res.	SPI3EN	SPI2EN	Res.	Res.	WWDGEN	RTCAPBEN	Res.	CRSEN	Res.	Res.	TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN	
	リセット値	0	0		0			0		0	0	0	0	0	0	0		0	0			0	1		0			0	0	0	0	0	0
0x5C	RCC_ APB1ENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1EN	Res.	Res.	Res.	Res.	Res.	Res.	I2C4EN	LPUART1EN
	リセット値																								0							0	0

表 51. RCC レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x60	RCC_ APB2ENR	Res.	Res.	Res.	Res.	Res.	HRTIM1EN	Res.	Res.	Res.	Res.	SAI1EN	TIM20EN	Res.	TIM17EN	TIM16EN	TIM15EN	SPI4EN	USART1EN	TIM8EN	SPI1EN	TIM1EN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGEN
	リセット値						0					0	0		0	0	0	0	0	0	0	0												0
0x68	RCC_ AHB1SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCSMEN	Res.	Res.	SRAM1SMEN	FlashSMEN	Res.	Res.	Res.	FMACSMEN	CORDICSMEN	DMAMUX1SMEN	DMA2SMEN	DMA1SMEN
	リセット値																				1			1	1				1	1	1	1	1
0x6C	RCC_ AHB2SMENR	Res.	Res.	Res.	Res.	Res.	RNGSMEN	Res.	AESSMEN	Res.	Res.	Res.	Res.	DAC4SMEN	DAC3SMEN	DAC2SMEN	DAC1SMEN	Res.	ADC345SMEN	ADC12SMEN	Res.	Res.	CCMSRAMSMEN	SRAM2SMEN	Res.	Res.	GPIOGSMEN	GPIOFSMEN	GPIOESMEN	GPIODSMEN	GPIOCSMEN	GPIOBSMEN	GPIOASMEN
	リセット値						1		1					1	1	1	1		1	1			1	1			1	1	1	1	1	1	1
0x70	RCC_ AHB3SMENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPISMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FMCSMEN
	リセット値																								1								1
0x78	RCC_ APB1SM ENR1	LPTIM1SMEN	I2C3SMEN		PWRSMEN			FDCANSMEN		USBSMEN	I2C2SMEN	I2C1SMEN	UART5SMEN	UART4SMEN	USART3SMEN	USART2SMEN		SPI3SMEN	SPI2SMEN			WWDGSMEN	RTCAPBSMEN		CRSSMEN			TIM7SMEN	TIM6SMEN	TIM5SMEN	TIM4SMEN	TIM3SMEN	TIM2SMEN
	リセット値	1	1		1			1		1	1	1	1	1	1	1		1	1			1	1		1			1	1	1	1	1	1
0x7C	RCC_ APB1SM ENR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UCPD1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	I2C4SMEN	LPUART1SMEN
	リセット値																								0							1	1
0x80	RCC_ APB2SMENR	Res.	Res.	Res.	Res.	Res.	HRTIM1SMEN	Res.	Res.	Res.	Res.	SAI1SMEN	TIM20SMEN	Res.	TIM17SMEN	TIM16SMEN	TIM15SMEN	SPI4SMEN	USART1SMEN	TIM8SMEN	SPI1SMEN	TIM1SMEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SYSCFGSMEN
	リセット値						1					1	1		1	1	1	1	1	1	1	1												1
0x88	RCC_CCIPR	ADC345SEL [1:0]		ADC12SEL [1:0]		CLK48SEL [1:0]		FDCANSEL [1:0]		I2S23SEL [1:0]		SAI1SEL [1:0]		LPTIM1SEL [1:0]		I2C3SEL [1:0]		I2C2SEL [1:0]		I2C1SEL [1:0]		LPUART1SEL [1:0]		UART5SEL [1:0]		UART4SEL [1:0]		USART3SEL [1:0]		USART2SEL [1:0]		USART1SEL [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 51. RCC レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x90	RCC_BDCR	Res.	Res.	Res.	Res.	Res.	Res.	LSCOSEL	LSCOEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BDRST	RTCEN	Res.	Res.	Res.	Res.	Res.	RTCSEL [1:0]		Res.	LSECSSD	LSECSSON	LSE DRV [1:0]		LSEBYP	LSERDY	LSEON
	リセット値							0	0								0	0						0	0		0	0	0	0	0	0	0
0x94	RCC_CSR	LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	BORRSTF	PINRSTF	OBLRSTF	Res.	RMVF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LSIRDY	LSION
	リセット値	0	0	0	0	1	1	0		0																						0	0
0x98	RCC_CRRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSI48CAL [8:0]									Res.	Res.	Res.	Res.	Res.	HSI48RDY	HSI48ON
	リセット値																	X	X	X	X	X	X	X	X	X						0	0
0x9C	RCC_CCIPR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	QSPISEL [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	I2C4SEL [1:0]	
	リセット値											0	0																			0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

8 クロック･リカバリ･システム（CRS）

8.1 概要

クロックリカバリシステム（CRS）は、高精度トリミング可能な内蔵 RC オシレータ HSI48 に作用する、高度なデジタルコントローラです。選択可能な同期信号との比較に基づいて、オシレータ出力周波数チェックを行うための強力な手段を提供しています。CRS では、測定した周波数の誤差値に基づいてオシレータのトリミングを自動調整することが可能です。また、手動でトリミングすることも可能です。

CRS は、USB ペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USB ホストによって 1ms 間隔で送信される USB バス上のフレーム開始（SOF）パケット信号から同期信号を取得できます。

同期信号は、LSE オシレータ出力からも取得することができ、またユーザ･ソフトウェアから生成することもできます。

8.2 CRS の主な機能

- 次のプログラム可能なプリスケーラと極性を持つ、選択可能な同期ソース：
  - LSE オシレータ出力
  - パケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU 不要の自動オシレータトリミング機能
- より迅速なスタートアップのための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16 ビットの周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割込み／イベント：
  - 期待される同期（ESYNC）
  - 同期 OK（SYNCOK）
  - 同期警告（SYNCWARN）
  - 同期またはトリミングエラー（ERR）

8.3 CRS の実装

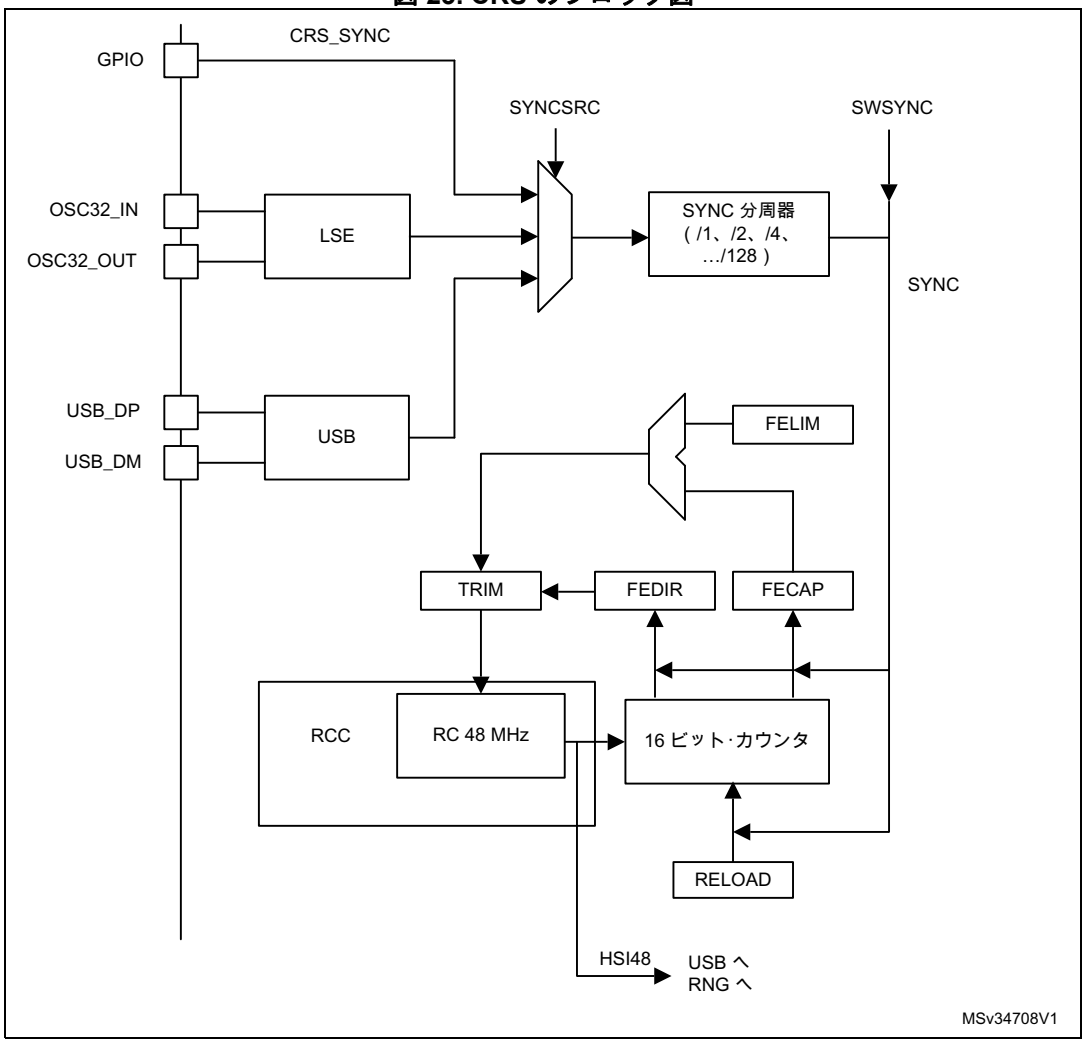
表 52. CRS の機能

機能	CRS1
TRIM 幅	7 ビット

8.4 CRS の機能詳細

8.4.1 CRS ブロック図

図 23. CRS のブロック図



8.4.2 同期入力

CRS_CFGR レジスタから選択可能な CRS 同期（SYNC）ソースは、LSE クロックからの信号または USB SOF 信号です。より安定した SYNC 入力を得るには、グリッチをフィルタするために、シンプルなデジタルフィルタ（3 対 2 の多数決方式、HSI48 クロックによるサンプリング方法）が導入されています。このソース信号には設定可能な極性があり、適切な周波数範囲（通常 1 kHz 前後）の同期信号を取得するために、プログラム可能なバイナリプリスケーラによって分周することができます。

CRS 同期ソース設定の詳細については、セクション 8.7.2 : CRS 設定レジスタ（CRS_CFGR）を参照してください。

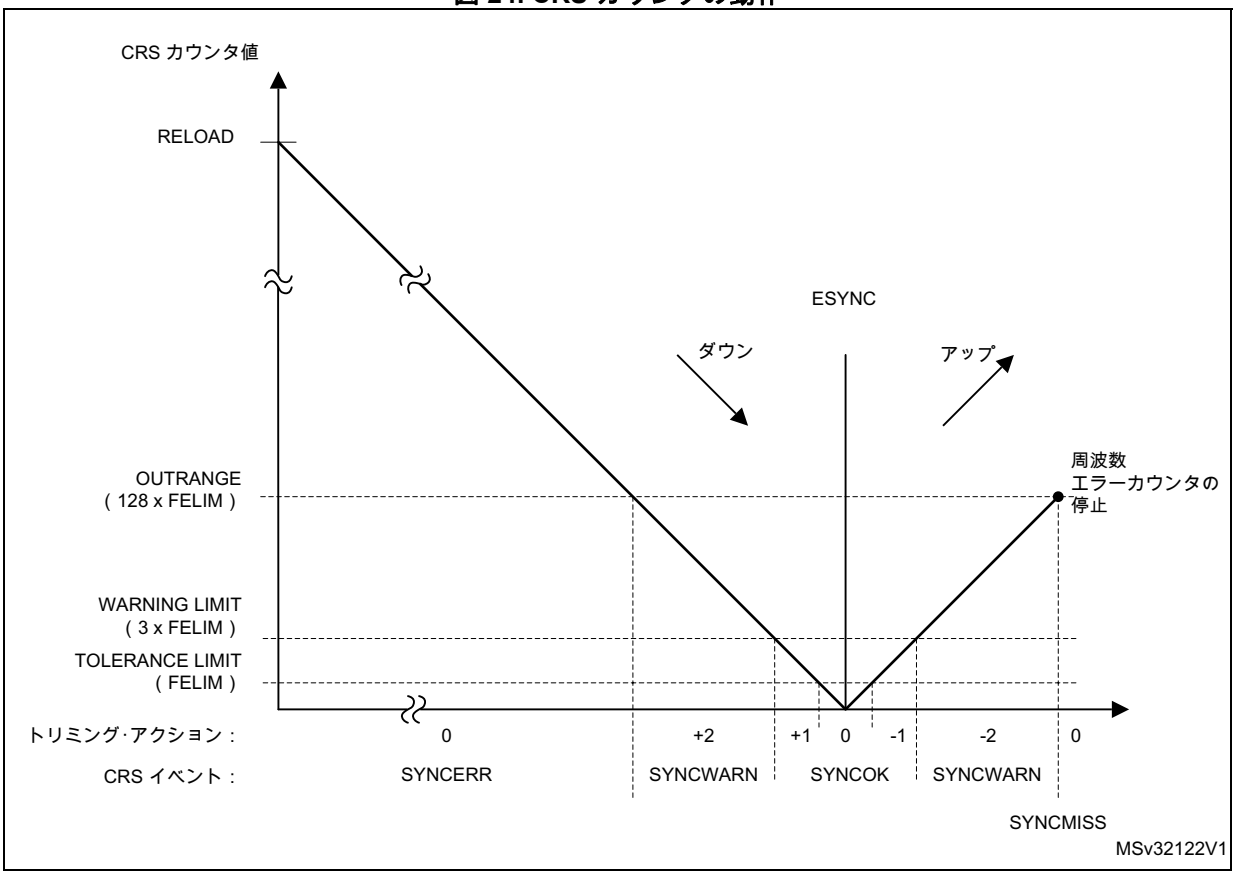
CRS_CR レジスタの SWSYNC ビットをセットして、ソフトウェアで同期イベントを生成することもできます。

8.4.3 周波数誤差測定

この周波数誤差カウンタは､各 SYNC イベントで RELOAD 値によって再ロードされる16 ビットのダウン／アップカウンタです。ESYNC （期待される同期）イベントが生成されるゼロ値に達するまでカウントダウンします。次に、OUTRANGE 制限によって最終的に停止する（SYNC イベントが受信されない場合）までカウントアップし、SYNCMISS イベントを生成します。OUTRANGE 制限は、周波数誤差制限(CRS_CFGR レジスタの FELIM フィールド)に 128 を乗算した値として定義されます。

SYNC イベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向が FECAP （周波数誤差キャプチャ）フィールドと CRS_ISR レジスタの FEDIR （周波数誤差方向）ビットに保管されます。ダウンカウントフェーズで（ゼロ値に達する前に）SYNC イベントが検出されたときは、実際の周波数がターゲットよりも小さいことを意味します（そのため、TRIM 値をインクリメントする必要がある）。アップカウントフェーズ中に検出されると、実際の周波数は高くなります（そのため、TRIM 値をデクリメントする必要がある)。

図 24. CRS カウンタの動作



8.4.4 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGR レジスタの FELIM フィールドに直接与えられる TOLERANCE LIMIT（許容制限）
- 3 x FELIM 値として定義される WARNING LIMIT（警告制限）
- 128 x FELIM 値として定義される OUTRANGE（エラー制限）

この比較の結果は、ステータスの表示を生成し、CRS_CR レジスタの AUTOTRIMEN ビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
  - SYNCOK ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1 回のトリミングステップによる調整で、十分適切な TRIM 値に達することができます。
  - SYNCOK ステータスが示されます。
  - AUTOTRIM モードで、1 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が警告制限以上であるものの、エラー制限を下回る場合は、より強力なトリミング動作が必要であることを意味します。次の周期までに適切な TRIM 値に達することができない危険性もあります。
  - SYNCWARN ステータスが示されます。
  - AUTOTRIM モードで、2 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合は、周波数がトリミング範囲外であることを意味します。これは、SYNC 入力がクリーンではなく、SYNC パルスが一部欠落している場合に発生する可能性があります（たとえば、USB SOF が 1 つ破損している場合）。
  - SYNCERR または SYNCMISS ステータスが示されます。
  - TRIM 値は、AUTOTRIM モードでは変更されません。

注： **TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM 値が誤差ちょうどに設定され、TRIMOVF ステータスが示されます。**

AUTOTRIM モード（CRS_CR レジスタにセットされた AUTOTRIMEN ビット）では、CRS_CR の TRIM フィールドがハードウェアによって調整され、読出し専用になります。

8.4.5 CRS の初期化と設定

RELOAD 値

再ロード値は、ターゲット周波数とプリスケーリング後の同期ソースの周波数間の比率に従って選択する必要があります。ゼロ値で期待される同期に達するために、1 ずつ減算します。計算式は、次のとおりです。

$$RELOAD = (f_{TARGET} / f_{SYNC}) - 1$$

RELOAD フィールドのリセット値は、48 MHz のターゲット周波数と 1 kHz の同期信号周波数（USB からの SOF 信号）に対応します。

FELIM 値

FELIM 値の選択は、HSI48 発振特性とその標準的なトリミングステップサイズに、深く関係しています。最適値は、HSI48 オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

$$FELIM = (f_{TARGET} / f_{SYNC}) * STEP[\%] / 100\% / 2$$

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられる必要があります。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM 値を少し上げることでヒステリシスを増やすことができます。

FELIM フィールドのリセット値は、$(f_{TARGET} / f_{SYNC}) = 48000$ と標準的なトリミングステップサイズ（0.14%）に対応しています。

注： **トリミングのステップ･サイズは製品によって異なります。正確な設定をデータシートで確認してください。**

注意： **RELOAD フィールドと FELIM フィールドの誤った設定に対するハードウェア保護はなく、不規則なトリム応答が発生する可能性があります。期待される動作モードでは、再ロード値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM 値よりも大きい値です（OUTRANGE 制限）。**

8.5 CRS 低電力モード

表 53. 低電力モードが CRS に与える影響

モード	説明
SLEEP	影響はありません。CRS 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	CRS レジスタが操作できなくなります。CRS によって、動作は STOP モードが終了し、HSI48 オシレータがリスタートするまで停止します。
STANDBY	CRS ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

8.6 CRS 割込み

表 54. 割込み制御ビット

割込みイベント	イベントフラグ	イネーブル制御ビット	クリアフラッグビット
期待される同期	ESYNCF	ESYNCIE	ESYNCC
同期 OK	SYNCOKF	SYNCOKIE	SYNCOKC
同期警告	SYNCWARNF	SYNCWARNIE	SYNCWARNC
同期またはトリミングエラー（TRIMOVF、SYNCMISS、SYNCERR）	ERRF	ERRIE	ERRC

8.7 CRS レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル·レジスタには、ワード（32 ビット）単位でのみアクセスできます。

8.7.1 CRS 制御レジスタ（CRS_CR）

アドレスオフセット：0x00

リセット値：0x0000 X000（7 ビット TRIM 幅をサポートしている製品では X=4、それ以外の製品では X=2）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TRIM[6:0]							SW SYNC	AUTO TRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNC WARNIE	SYNC OKIE
	rw	rw	rw	rw	rw	rw	rw	rt_w1	rw	rw		rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **TRIM[6:0]**：HSI48 オシレータ高精度トリミング

HSI48 オシレータの高精度トリミング値はデフォルトで 64 であり、トリミング範囲の中心値に相当します。

ビット 7 **SWSYNC**：ソフトウェア SYNC イベントの生成

このビットは、ソフトウェア SYNC イベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ソフトウェア SYNC イベントが生成されます。

ビット 6 **AUTOTRIMEN**：自動トリミング有効

このビットは、2 つの SYNC イベント間で測定した周波数誤差に応じて、TRIM ビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIM ビットが読出し専用になります。TRIM 値は、測定した周波数誤差値に応じて、ハードウェアで 1 度に 1 または 2 ステップずつ調整できます。詳細については、セクション 8.4.4： を参照してください。

0：自動トリミングは無効です。ユーザは、TRIM ビットを調整できます。

1：自動トリミングは有効です。TRIM ビットは読出し専用で、ハードウェアによって制御されます。

ビット 5 **CEN**：周波数誤差カウンタが有効

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。

0：周波数誤差カウンタは無効です。

1：周波数誤差カウンタは有効です。

このビットをセットすると、CRS_CFGR レジスタは書込み保護され、変更できません。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCIE**：期待される SYNC 割込み有効

0：期待される SYNC（ESYNCF）割込みは無効です。

1：期待される SYNC（ESYNCF）割込みは有効です。

ビット 2 **ERRIE**：同期またはトリミング·エラー割込みが有効

0：同期またはトリミングエラー（ERRF）割込みは無効です。

1：同期またはトリミングエラー（ERRF）割込みは有効です。

ビット 1 **SYNCWARNIE**：SYNC 警告割込み有効

0：SYNC 警告（SYNCWARNF）割込みは無効です。

1：SYNC 警告（SYNCWARNF）割込みは有効です。

ビット 0 **SYNCOKIE**：SYNC イベント OK 割込み有効

0：SYNC イベント OK（SYNCOKF）割込みは無効です。

1：SYNC イベント OK（SYNCOKF）割込みは有効です。

8.7.2 CRS 設定レジスタ（CRS_CFGR）

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです（CEN ビットは CRS_CR でクリアされる）。このカウンタを有効にすると、このレジスタは書込み保護されます。

アドレスオフセット：0x04

リセット値：0x2022：BB7F

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

ビット 31 **SYNCPOL**：SYNC 極性選択

このビットは、SYNC 信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。

0：SYNC は立ち上がりエッジでアクティブです（デフォルト）。

1：SYNC は立ち下がりエッジでアクティブです。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 SYNCSRC[1:0]：SYNC 信号ソース選択

これらのビットは、SYNC 信号ソースを選択するために、ソフトウェアによってセット／クリアされます。

00：SYNC 信号ソースとして GPIO を選択します。

01：SYNC 信号ソースとして LSE を選択します。

10：SYNC 信号ソースとして USB SOF を選択します（デフォルト）。

11：予約済みです。

注： USB LPM（Link Power Management、リンク電源管理）の使用時にデバイスが SLEEP モードである場合、周期的なUSBの SOF信号はHOSTからは生成されません。そのため、実行中に HSI48 オシレータを較正するために SYNC 信号が CRS に供給されることはありません。SLEEP モードからのウェイクアップ後に必要なクロック精度を確保するために、LSEまたはGPIOでの基準クロックを SYNC 信号として使用する必要があります。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **SYNCDIV[2:0]**：SYNC 分周器

これらのビットは、SYNC 信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。

000：SYNC は分周されません（デフォルト）。

001：SYNC は 2 分周されます。

010：SYNC は 4 分周されます。

011：SYNC は 8 分周されます。

100：SYNC は 16 分周されます。

101：SYNC は 32 分周されます。

110：SYNC は 64 分周されます。

111：SYNC は 128 分周されます。

ビット 23:16 **FELIM[7:0]**：周波数誤差制限

FELIM には、CRS_ISR レジスタの FECAP[15:0] ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP 評価の詳細については、セクション 8.4.4：を参照してください。

ビット 15:0 **RELOAD[15:0]**：カウンタ再ロード値

RELOAD は、SYNC イベントごとに周波数誤差カウンタにロードされる値です。

カウンタの動作の詳細については、セクション 8.4.3：を参照してください。

8.7.3 CRS 割込みおよびステータスレジスタ（CRS_ISR）

アドレスオフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FECAP[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEDIR	Res.	Res.	Res.	Res.	TRIM OVF	SYNC MISS	SYNC ERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNC WARNF	SYNC OKF
r					r	r	r					r	r	r	r

ビット 31:16 **FECAP[15:0]**：周波数誤差キャプチャ

FECAP は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタ値です。

FECAP の使用の詳細については、セクション 8.4.4：を参照してください。

ビット 15 **FEDIR**：周波数誤差方向

FEDIR は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。

0：アップカウント方向。実際の周波数はターゲットを上回ります。

1：ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **TRIMOVF**：トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによって TRIM 値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：トリミングエラーを信号化しません。

1：トリミングエラーを信号化します。

ビット 9 **SYNCMISS**：SYNC の欠落

このフラグは、周波数誤差カウンタが値 FELIM * 128 に達し、SYNC が検出されない場合にハードウェアによってセットされます。これは、SYNC パルスが失われているということ、または TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が高すぎる）、他の操作を行う必要があるということを意味します。この時点で、周波数誤差カウンタは停止し（次の SYNC を待つ）、CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC の欠落エラーを信号化しません。

1：SYNC の欠落エラーを信号化します。

ビット 8 **SYNCERR**：SYNC エラー

このフラグは、SYNC パルスが ESYNC イベントより前に発生し、測定した周波数誤差が FELIM * 128 以上である場合に、ハードウェアによってセットされます。これは、TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC エラーを信号化しません。

1：SYNC エラーを信号化します。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCF**：期待される SYNC フラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CR レジスタの ESYNCIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ESYNCC ビットをセットすることで、ソフトウェアによってクリアされます。

0：期待された SYNC を信号化しません。

1：期待された SYNC を信号化します。

ビット 2 **ERRF**：エラー･フラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、および SYNCERR ビットの論理和です。CRS_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、および SYNCERR ビットをクリアします。

0：同期またはトリミングエラーを信号化しません。

1：同期またはトリミングエラーを信号化します。

ビット 1 **SYNCWARNF**：SYNC 警告フラグ

このフラグは、測定した周波数誤差が FELIM * 3 以上で FELIM * 128 未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM 値を複数のステップで調整する必要があります。CRS_CR レジスタの SYNCWARNIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの SYNCWARNC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC 警告を信号化しません。

1：SYNC 警告を信号化します。

ビット 0 **SYNCOKF**：SYNC イベント OK フラグ

このフラグは、測定した周波数誤差が FELIM * 3 未満である場合に、ハードウェアによってセットされます。これは、TRIM 値の補正が必要であること、または周波数誤差の補正は 1 つのトリミングステップで十分であることを意味します。CRS_CR レジスタの SYNCOKIE ビットがセットされている場合、割込みが生成されます。CRS_ICR レジスタの SYNCOKC ビットをセットすることで、ソフトウェアによってクリアされます。

0：SYNC イベント OK フラグを信号化しません。

1：SYNC イベント OK フラグを信号化します。

8.7.4 CRS 割込みフラグクリアレジスタ（CRS_ICR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ESYNCC**：期待される SYNC クリア･フラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの ESYNCF フラグをクリアします。

ビット 2 **ERRC**：エラー･クリア･フラグ

このビットに 1 を書き込むと、TRIMOVF、SYNCMISS、および SYNCERR ビットがクリアされ、結果として CRS_ISR レジスタの ERRF フラグもクリアされます。

ビット 1 **SYNCWARNC**：SYNC 警告クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCWARNF フラグをクリアします。

ビット 0 **SYNCOKC**：SYNC イベント OK クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCOKF フラグをクリアします。

8.7.5 CRS レジスタマップ

表 55. CRS レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRS_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[6]	TRIM[5:0]						SWSYNC	AUTOTRIMEN	CEN	Res.	ESYNCIE	ERRIE	SYNCWARNIE	SYNCOKIE
	リセット値																		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	CRS_CFGR	SYNCPOL	Res.	SYNC SRC [1:0]		Res.	SYNC DIV [2:0]			FELIM[7:0]								RELOAD[15:0]															
	リセット値	0		1	0		0	0	0	0	0	1	0	0	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1	1	1	1
0x08	CRS_ISR	FECAP[15:0]																FEDIR	Res.	Res.	Res.	Res.	TRIMOVF	SYNCMISS	SYNCERR	Res.	Res.	Res.	Res.	ESYNCF	ERRF	SYNCWARNF	SYNCOKF
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0					0	0	0	0
0x0C	CRS_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ESYNCC	ERRC	SYNCWARNC	SYNCOKC
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

9 汎用 I/O（GPIO）

9.1 概要

各汎用 I/O ポートにはそれぞれ 4 つの 32 ビット設定レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）、2 つの 32 ビットデータレジスタ（GPIOx_IDR、GPIOx_ODR）、および 1 つの 32 ビットセット／リセットレジスタ（GPIOx_BSRR）があります。さらに、すべての GPIO には、それぞれ 32 ビットロックレジスタ（GPIOx_LCKR）と 2 つの 32 ビットオルタネート機能選択レジスタ（GPIOx_AFRH、GPIOx_AFRL）があります。

9.2 GPIO の主な機能

- 出力状態：プッシュプルまたはオープン･ドレイン + プルアップまたはプルダウン
- 出力データレジスタ（GPIOx_ODR）またはペリフェラル（オルタネート機能出力）からの出力データ
- 各 I/O のスピード選択
- 入力状態：フローティング、プルアップ／プルダウン、アナログ
- 入力データレジスタ（GPIOx_IDR）またはペリフェラル（オルタネート機能入力）への入力データ
- GPIOx_ODR へのビット単位の書込みアクセス用のビットセット／リセットレジスタ（GPIOx_BSRR）
- I/O ポート設定を固定するロック機構（GPIOx_LCKR）
- アナログ機能
- オルタネート機能選択レジスタ
- 2 クロック･サイクルごとのレベル変化に対応できる高速トグル
- 柔軟性の高いピンの多重化により、I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用可能

9.3 GPIO の機能説明

各 I/O ポート固有のハードウェア特性については、データシートに記載されています。これに従って、汎用 I/O（GPIO）ポートの各ポート･ビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各 I/O ポートビットは自由にプログラム可能ですが、I/O ポートレジスタには 32 ビットワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRR レジスタを使用すると、任意の GPIOx_ODR レジスタにビット単位の読出し／変更アクセスを行うことができます。これにより、読出しと変更アクセスの間に IRQ が発生するリスクを回避できます。

図 25 および 図 26 に、標準 I/O ポート･ビットと 5 V トレラント I/O ポート･ビットの基本構成をそれぞれ示します。表 56 に、ポート･ビットに対して可能な設定を示します。

図 25. I/O ポートビットの基本構成

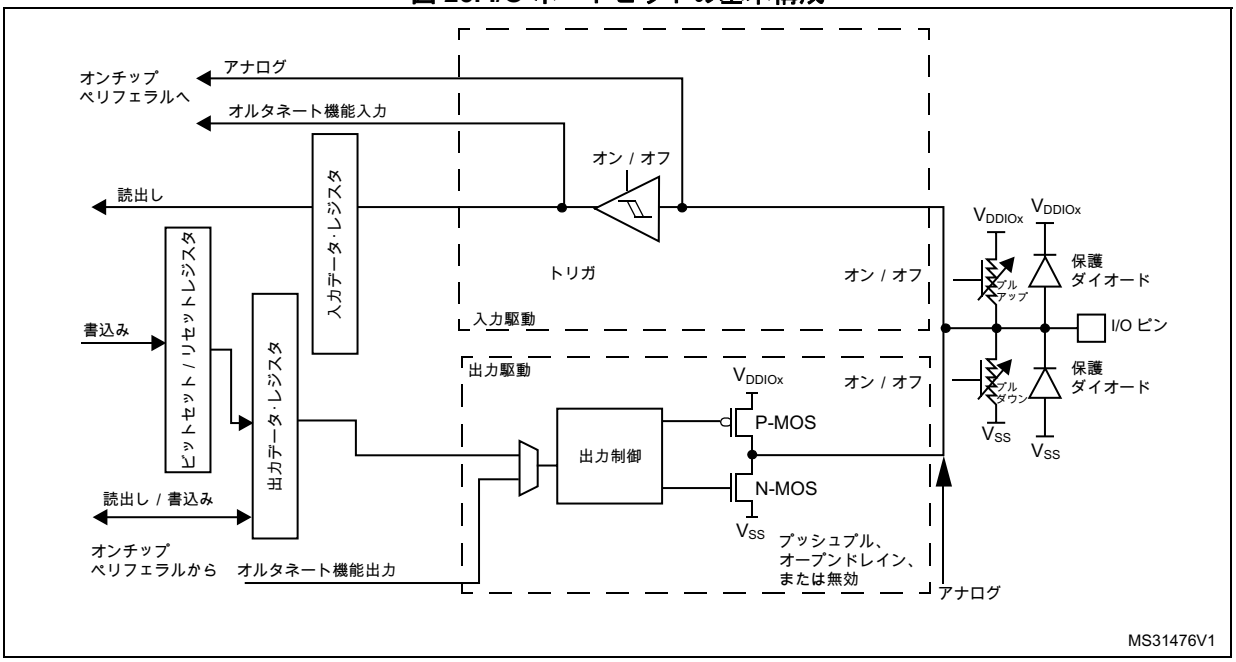
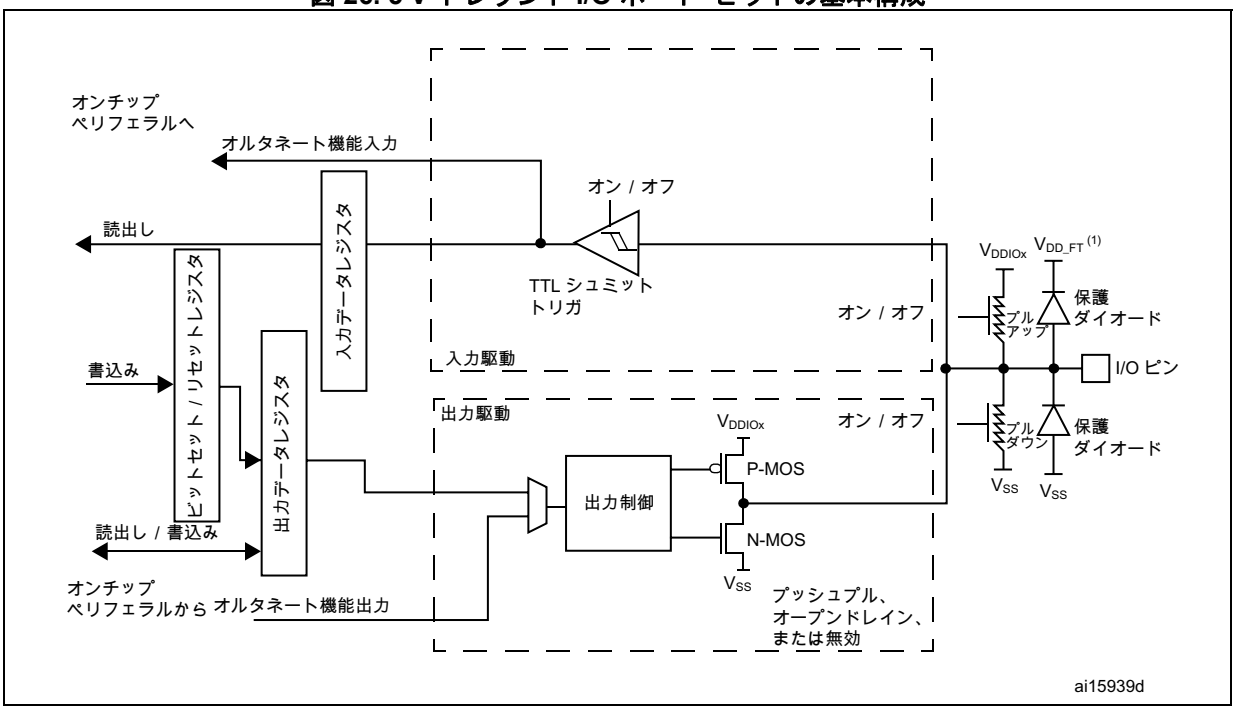


図 26. 5 V トレラント I/O ポート･ビットの基本構成



1. V_{DD_FT} は、5 V トレラント I/O に固有の電位であり、V_{DD} とは異なります。

表 56. ポートビット設定表(1)

MODE(i) [1:0]	OTYPE(i)	OSPEED(i) [1:0]		PUPD(i) [1:0]		I/O 設定	
01	0	SPEED [1:0]		0	0	GP 出力	PP
	0			0	1	GP 出力	PP + PU
	0			1	0	GP 出力	PP + PD
	0			1	1	予約済みです。	
	1			0	0	GP 出力	OD
	1			0	1	GP 出力	OD + PU
	1			1	0	GP 出力	OD + PD
	1			1	1	予約済み（GP 出力 OD）	
10	0	SPEED [1:0]		0	0	AF	PP
	0			0	1	AF	PP + PU
	0			1	0	AF	PP + PD
	0			1	1	予約済みです。	
	1			0	0	AF	OD
	1			0	1	AF	OD + PU
	1			1	0	AF	OD + PD
	1			1	1	予約済みです。	
00	x	x	x	0	0	入力	フローティング
	x	x	x	0	1	入力	PU
	x	x	x	1	0	入力	PD
	x	x	x	1	1	予約済み（入力フローティング）	
11	x	x	x	0	0	入力／出力	アナログ
	x	x	x	0	1	予約済みです。	
	x	x	x	1	0	入力／出力	アナログ、PD
	x	x	x	1	1	予約済みです。	

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

9.3.1 汎用 I/O（GPIO）

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどの I/O ポートは アナログモードに設定されています。

リセット後、デバッグ･ピンはオルタネート機能のプルアップまたはプルダウンに設定されています。

- PA15：JTDI プルアップ
- PA14：JTCK/SWCLK プルダウン
- PA13：JTMS/SWDAT プルアップ
- PB4：NJTRST プルアップ
- PB3：JTDO フローティング状態、プルアップもプルダウンもなし

PB8/BOOT0 は、リセット中の少なくともオプション･バイトのローディング段階が終了するまでは、入力モードになります。セクション 9.3.15：PB8 を GPIO として使用を参照してください。

ピンが出力として設定されている場合、出力データ･レジスタ（GPIOx_ODR）に書き込まれた値が I/O ピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード（ローレベルのみが駆動され、ハイレベルはハイインピーダンス）で使用することができます。

入力データレジスタ（GPIOx_IDR）は、AHB クロックサイクルごとに、I/O ピン上のデータをキャプチャします。

すべての GPIO ピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDR レジスタの値によってこれらを有効化/無効化できます。

9.3.2 I/O ピンオルタネート機能マルチプレクサと配置

デバイスの I/O ピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に 1 つのペリフェラルオルタネート機能（AF）のみが 1 つの I/O ピンに接続可能となっています。この方法により、同じ I/O ピンを共有するペリフェラル間での競合を無くすことができます。

各 I/O ピンは、最大16 のオルタネート機能入力（AF0～AF15）を持つマルチプレクサを内蔵しており、これらの入力は、GPIOx_AFRL（ピン 0～7）と GPIOx_AFRH（ピン 8～15）レジスタを介して設定することができます。

- リセット後、マルチプレクサの選択はオルタネート機能 0（AF0）です。I/O は、GPIOx_MODER レジスタを通してオルタネート機能モードで設定されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。

この柔軟性の高い I/O 多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれの I/O ピンに配置されており、さらに小型のパッケージで使用できるペリフェラルの数を最適化します。

I/O を任意の設定で使用するには、次の手順に従います。

- **デバッグ機能**：各デバイスのリセット後、これらのピンはデバッガホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **GPIO**：必要とする I/O を、GPIOx_MODER レジスタで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**
  - I/O を GPIOx_AFRL または GPIOx_AFRH レジスタの どちらかで必要とする AFx に接続します。
  - タイプ、プルアップ／プルダウン、出力スピードをそれぞれ GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタで選択します。
  - 必要とする I/O を、GPIOx_MODER レジスタでオルタネート機能として設定します。

- **追加機能**：
  - ADC、DAC、OPAMP、COMP については、目的の I/O を GPIOx_MODER レジスタでアナログ·モードに設定したうえで、ADC、DAC、OPAMP、COMP の各レジスタで必要な機能を設定します。

    上述のとおり、追加機能（D/A コンバータや OPAMP など）では、出力は該当のペリフェラルによって制御されます。ペリフェラル制御レジスタで追加機能出力を有効にする前に、I/O ポートアナログ機能を選択するよう注意する必要があります。
  - RTC、WKUPx、オシレータなどの追加機能については、関連する　RTC、PWR、および RCC レジスタで必要な機能を設定します。これらの機能は、標準の GPIO レジスタの設定よりも優先されます。

オルタネート機能 I/O ピンの配置に関する詳細は、デバイス·データシートの「オルタネート機能配置」表を参照してください。

9.3.3 I/O ポート制御レジスタ

各 GPIO ポートには 32 ビットメモリマップド制御レジスタが 4 つ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR）あり、最大 16 個の I/O を設定します。GPIOx_MODER レジスタは I/O モード（入力、出力、AF、アナログ）を選択するために使用されます。GPIOx_OTYPER および GPIOx_OSPEEDR レジスタは、出力タイプ（プッシュプルまたはオープンドレイン）およびスピードを選択するために使用されます。I/O の方向がどちらであっても、GPIOx_PUPDR レジスタは、プルアップ／プルダウンを選択するために使用されます。

9.3.4 I/O ポートデータレジスタ

各 GPIO には、16 ビットメモリマップドデータレジスタが 2 つあります。入力データレジスタ（GPIOx_IDR）と出力データレジスタ（GPIOx_ODR）です。GPIOx_ODR は出力されるデータを格納し、読出し/書込みアクセスが可能です。I/O から入力されるデータは読出し専用の入力データレジスタ（GPIOx_IDR）に格納されます。

レジスタの説明は、セクション 9.4.5 : GPIO ポート入力データ·レジスタ（GPIOx_IDR）（x = A ~ G）および セクション 9.4.6 : GPIO ポート出力データレジスタ（GPIOx_ODR）（x = A ~ G）を参照してください。

9.3.5 I/O データのビット単位の操作

ビットセット／リセットレジスタ（GPIOx_BSRR）は 32 ビットレジスタで、アプリケーションによる出力データレジスタ（GPIOx_ODR）のビット単位のセット／リセットを可能にします。ビットセットリセットレジスタは GPIOx_ODR の 2 倍のサイズです。

GPIOx_ODR の各ビットには GPIOx_BSRR の 2 つの制御ビットBS(i) と BR(i) が対応します。ビットBS(i) に 1 を書き込むと、対応する ODR(i) ビットが**セット** されます。ビットBR(i) に 1 を書き込むと、対応する ODR(i) ビットが**リセット** されます。

GPIOx_BSRR のいかなるビットに 0 を書き込んでも GPIOx_ODR の対応するビットには影響しません。仮に、GPIOx_BSRR のビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRR レジスタを使用した GPIOx_ODR 内の個々のビットの変更には、1 回限りの効果しかなく、GPIOx_ODR ビットを固定するものではありません。GPIOx_ODR のビットは常に直接アクセスすることができます。GPIOx_BSRR レジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODR をビットレベルでプログラムする場合は、ソフトウェアで割込みを無効にする必要はありません。1 回の不可分な AHB 書込みアクセスで 1 ビットまたは複数ビットを変更することができます。

9.3.6 GPIO ロック機構

GPIOx_LCKR レジスタへ特定の書込みシーケンスを行うことにより、GPIO 制御レジスタをロックすることができます。ロックされるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH です。

GPIOx_LCKR レジスタに書き込むには、特定の書込み／読出しシーケンスを行う必要があります。このレジスタのビット 16 に適切な LOCK シーケンスを行う場合、LCKR[15:0] の値を使用して I/O の設定を固定します（この書込みシーケンス中、LCKR[15:0] の値は同じである必要があります）。あるポートビットに LOCK シーケンスが行われると、次の MCU リセットまたはペリフェラルリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ（GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH）の対応するビットが停止されます。

GPIOx_LCKR ビット 16 を [15:0] ビットと同時に設定する必要があるため、この LOCK シーケンス（セクション 9.4.8 : GPIO ポート設定ロック･レジスタ（GPIOx_LCKR）（x = A ~ G）を参照）は、GPIOx_LCKR レジスタへのワード（32 ビット長）アクセスを使用してのみ実行できます。

詳細については、セクション 9.4.8 : GPIO ポート設定ロック･レジスタ（GPIOx_LCKR）（x = A ~ G）の LCKR レジスタの説明を参照してください。

9.3.7 I/O オルタネート機能の入力／出力

各 I/O が使用できるオルタネート機能入力／出力の 1 つを選択するため、2 つのレジスタが用意されています。これらのレジスタを使用し、必要に応じて、アプリケーションでオルタネート機能を他のピンに接続することができます。

つまり、GPIOx_AFRL および GPIOx_AFRH オルタネート機能レジスタを使用していくつかの使用可能なペリフェラル機能が、各 GPIO に多重化されることになります。こうして各 I/O に使用可能な機能のどれか 1 つをアプリケーションによって選択できます。AF 選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、任意の I/O が持つオルタネート機能入力／出力に対し 1 つのチャネルが選択されます。

どの機能が各 GPIO ピンに多重化されているかについてはデバイスのデータシートを参照してください。

9.3.8 外部割込み／ウェイクアップ･ライン

すべてのポートに外部割込み機能があります。外部割込みラインを使用するには、ポートを入力モードに設定する必要があります。

セクション 15 : 拡張割込み／イベント･コントローラ（EXTI）およびセクション 15.3.2 : ウェイクアップイベント管理を参照してください。

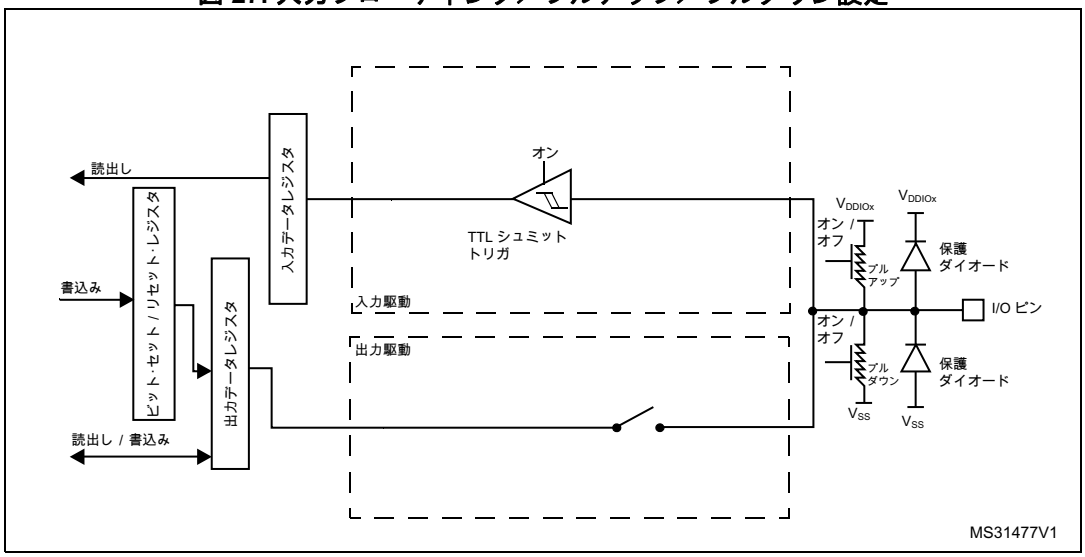
9.3.9 入力設定

I/O ポートが入力としてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。

図 27 は、I/O ポートビットの入力設定を示しています。

図 27. 入力フローティング／プルアップ／プルダウン設定



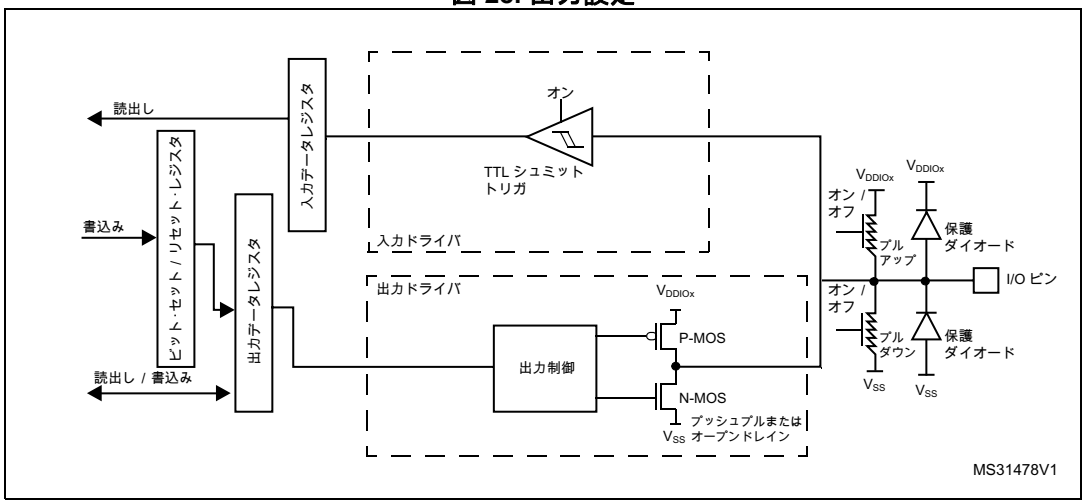
9.3.10 出力設定

I/O ポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
  - オープンドレインモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときはポートはハイインピーダンス状態のままです（P-MOS が有効になることはない）。
  - プッシュプルモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときは P-MOS が有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。
- 出力データレジスタの読出しアクセスによって、最後に書き込まれたデータが得られます。

図 28 は、I/O ポートビットの出力設定を示しています。

図 28. 出力設定



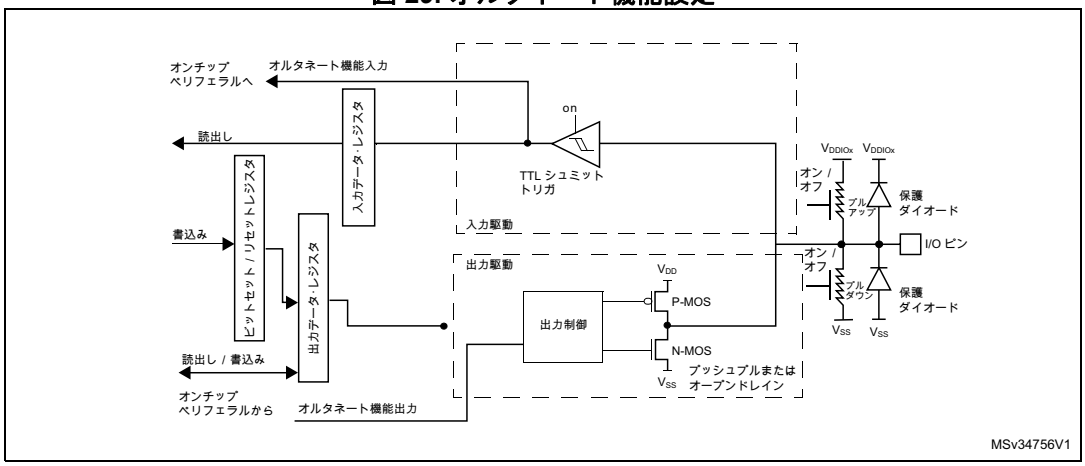
9.3.11 オルタネート機能設定

I/O ポートをオルタネート機能としてプログラムすると以下の動作が得られます。

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル（トランスミッタイネーブルおよびデータ）からの信号で駆動されます。
- シュミットトリガ入力が有効になります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読出しアクセスによって、I/O 状態が得られます。

図 29 は、I/O ポートビットのオルタネート機能設定 を示しています。

図 29. オルタネート機能設定



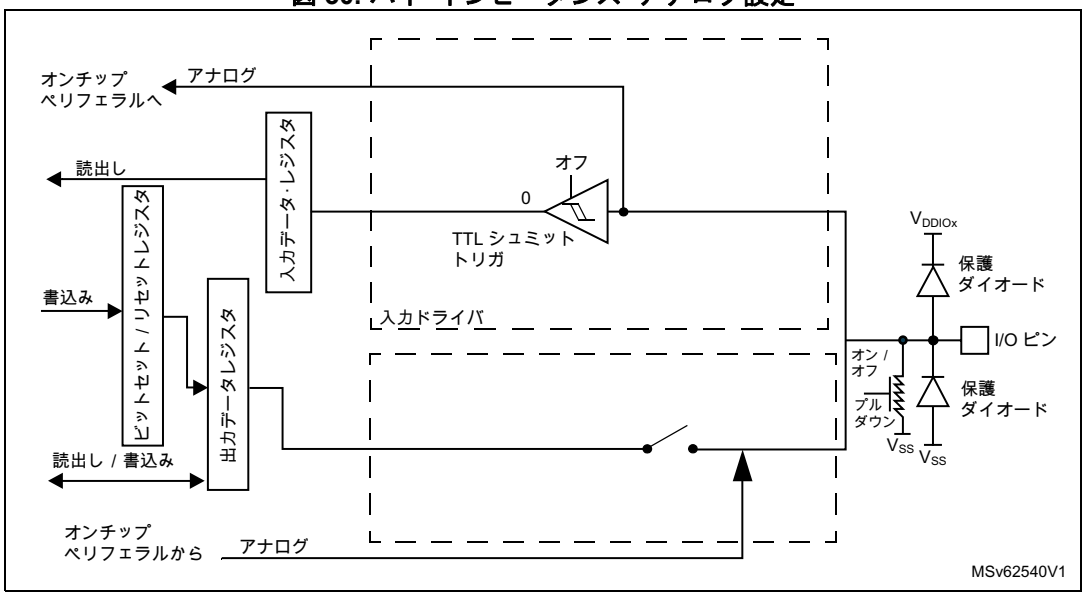
9.3.12 アナログ設定

I/O ポートをアナログとしてプログラムすると以下の動作が得られます。

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップはハードウェアによって無効にされます。ウィークプルダウンは設定可能です。
- 入力データ･レジスタに読出しアクセスすると値“0”が得られます。

図 30 は、I/O ポートビットのハイインピーダンスアナログ入力設定を示しています。

図 30. ハイ･インピーダンス･アナログ設定



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

HSE または LSE オシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータ･ピンを通常の GPIO として使用できます。

HSE または LSE オシレータがスイッチオンされた場合（RCC_CSR レジスタの HSEON または LSEON ビットをセット）、関連ピンはオシレータによって制御され、GPIO 設定の影響を受けません。

オシレータをユーザ外部クロック･モードに設定すると、OSC_IN ピンまたは OSC32_IN ピンのみがクロック入力として確保され、OSC_OUT または OSC32_OUT ピンは引き続き通常の GPIO として使用できます。

9.3.14 GPIO ピンを RTC 供給ドメインで使用

コア電源ドメインの電源がオフになると（デバイスが STANDBY モードに移行）、PC13/PC14/PC15 の GPIO が機能しなくなります。この場合、それらの GPIO 設定が RTC 設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

RTC による I/O 制御の詳細は、セクション 35.3 : RTC の機能説明を参照してください。

9.3.15 PB8 を GPIO として使用

PB8 はブート・ピン（BOOT0）または GPIO として使用できます。ユーザ・オプション・バイトの nSWBOOT0 ビットに応じて、入力モードからアナログ入力モードに切り替わります。

- nSWBOOT0 = 1 の場合、オプションバイトローディング後
- nSWBOOT0 = 0 の場合、リセット後

注： **PB8 ピンを未接続のままにする場合は、消費電流を抑えるためにアナログ・モード以外に設定することを推奨します。**

9.3.16 PG10 を GPIO として使用

PG10 はリセットピン（NRST）または GPIO として使用できます。ユーザオプションバイトの NRST_MODE ビットに応じて、これらのモードに切り替わります。

- リセット入出力：パワーオンリセット時またはオプションバイトローディング NRST_MODE = 3 後のデフォルト
- リセット入力のみ：オプションバイトローディング NRST_MODE = 1 後
- GPIO PG10 モード：オプションバイトローディング NRST_MODE = 2 後

NRST ピンの詳細については、セクション 7.1.2：システムリセットを参照してください。

9.4 GPIO レジスタ

このセクションには、GPIO レジスタの詳細な説明が記載されています。

レジスタビット、レジスタアドレスオフセット、リセット値の概要は、表 57 を参照してください。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

9.4.1 GPIO ポートモードレジスタ（GPIOx_MODER）（x = A ~ G）

アドレスオフセット：0x00

リセット値：0xABFF FFFF（ポート A）

リセット値：0xFFFF FEBF（ポート B）

リセット値：0xFFFF FFFF（ポート C ~ G)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODE15 [1:0]		MODE14 [1:0]		MODE13 [1:0]		MODE12 [1:0]		MODE11 [1:0]		MODE10 [1:0]		MODE9 [1:0]		MODE8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODE7 [1:0]		MODE6 [1:0]		MODE5 [1:0]		MODE4 [1:0]		MODE3 [1:0]		MODE2 [1:0]		MODE1 [1:0]		MODE0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **MODE[15:0][1:0]**：ポート x I/O ピン y（y = 15 ~ 0）の設定

これらのビットは、I/O モードを設定するために、ソフトウェアによって書き込まれます。

00：入力モード

01：汎用出力モード

10：オルタネート機能モード

11：アナログ･モード（リセット状態）

注： **PB8 ピンを未接続のままにする場合は、消費電流を抑えるためにアナログ･モード以外に設定することを推奨します。**

9.4.2 GPIO ポート出力タイプ･レジスタ（GPIOx_OTYPER）（x = A ~ G）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **OT[15:0]**：ポート x 設定用 I/O ピン y（y = 15～0）

これらのビットは、I/O 出力タイプを設定するために、ソフトウェアによって書き込まれます。

0：出力プッシュプル（リセット状態）

1：出力オープンドレイン

9.4.3 GPIO ポート出力スピードレジスタ（GPIOx_OSPEEDR）（x = A ~ G）

アドレスオフセット：0x08

リセット値：0x0C00 0000（ポート A）

リセット値：0x0000 0000（他のポート）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEED15 [1:0]		OSPEED14 [1:0]		OSPEED13 [1:0]		OSPEED12 [1:0]		OSPEED11 [1:0]		OSPEED10 [1:0]		OSPEED9 [1:0]		OSPEED8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OSPEED7 [1:0]		OSPEED6 [1:0]		OSPEED5 [1:0]		OSPEED4 [1:0]		OSPEED3 [1:0]		OSPEED2 [1:0]		OSPEED1 [1:0]		OSPEED0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **OSPEED[15:0][1:0]**：ポート x I/O ピン y（y = 15 ~ 0）の設定

これらのビットは、I/O の出力スピードを設定するために、ソフトウェアによって書き込まれます。

00：ロースピード

01：ミディアムスピード

10：ハイスピード

11：超ハイスピード

注： 周波数仕様、およびスピード別の電源や負荷条件については、デバイスのデータシートを参照してください。

9.4.4 GPIO ポート･プルアップ／プルダウン･レジスタ（GPIOx_PUPDR）（x = A ~ G）

アドレスオフセット：0x0C

リセット値：0x6400 0000（ポート A）

リセット値：0x0000 0100（ポート B）

リセット値：0x0000 0000（他のポート）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPD15 [1:0]		PUPD14 [1:0]		PUPD13 [1:0]		PUPD12 [1:0]		PUPD11 [1:0]		PUPD10 [1:0]		PUPD9 [1:0]		PUPD8 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPD7 [1:0]		PUPD6 [1:0]		PUPD5 [1:0]		PUPD4 [1:0]		PUPD3 [1:0]		PUPD2 [1:0]		PUPD1 [1:0]		PUPD0 [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **PUPD[15:0][1:0]**：ポート x I/O ピン y（y = 15 ~ 0）の設定

これらのビットは、I/O のプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

00：プルアップ／プルダウンなし

01：プルアップ

10：プルダウン

11：予約済みです。

9.4.5 GPIO ポート入力データ・レジスタ（GPIOx_IDR）（x = A ~ G）

アドレスオフセット：0x10

リセット値：0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ID[15:0]**：ポート x 入力データ用 I/O ピン y（y = 15 ~ 0）
これらのビットは読出し専用です。これらのビットには、対応する I/O ポートの入力値が格納されています。

9.4.6 GPIO ポート出力データ レジスタ（GPIOx_ODR）（x = A ~ G）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **OD[15:0]**：ポート x 出力データ用 I/O ピン y（y = 15 ~ 0）
これらのビットは、ソフトウェアによって読出し／書込みができます。

注： GPIOx_BSRR レジスタ（x = A ~ G）への書込みによって各 OD ビットを個々にセットまたはリセットすることで、ビットのアトミックなセットまたはリセットができます。

9.4.7 GPIO ポート･ビット･セット／リセット･レジスタ（GPIOx_BSRR）（x = A ~ G）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 **BR[15:0]**：ポート x リセット用 I/O ピン y（y = 15～0）

これらのビットは書込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODx ビットの値は変化しません。

1：対応する ODx ビットをリセットします。

注： BSx ビットと BRx ビットの両方がセットされた場合、BSx ビットが優先されます。

ビット 15:0 **BS[15:0]**：ポート x セット用 I/O ピン y（y = 15～0）

これらのビットは書込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODx ビットの値は変化しません。

1：対応する ODx ビットをセットします。

9.4.8 GPIO ポート設定ロック･レジスタ（GPIOx_LCKR）（x = A ~ G）

このレジスタは、ビット 16（LCKK）に正しい書込みシーケンスが行われたときに、ポート･ビットの設定をロックするために使用されます。ビット [15:0] の値は、GPIO の設定をロックするために使用されます。書込みシーケンスの間は、LCKR[15:0] の値を変更することはできません。あるポートビットに LOCK シーケンスが適用されると、次の MCU リセットまたはペリフェラルリセットまで、このポートビットの値を変更できなくなります。

注： **GPIOx_LCKR レジスタへの書込みには特定の書込みシーケンスが使われます。このロックシーケンスではワードアクセス（32 ビット長）のみ可能です。**

各ロックビットによって、特定の設定レジスタ（制御レジスタおよびオルタネート機能レジスタ）が固定されます。

アドレスオフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LCKK**：ロック･キー

このビットは常時読出しができます。ロックキー書込みシーケンスを使用しなければ変更できません。

0：ポート設定ロックキーは無効です。

1：ポート設定ロックキーは有効です。GPIOx_LCKR レジスタは、次の MCU リセットまたはペリフェラルリセットまでロックされます。

ロックキー書込みシーケンス：

書込み LCKR[16] = “1” + LCKR[15:0]

書込み LCKR[16] = “0” + LCKR[15:0]

書込み LCKR[16] = “1” + LCKR[15:0]

読出し LCKR

読出し LCKR[16] = “1”（この読出し操作は任意だが、ロックが有効であることを確認できる。）

注： **ロックキー書込みシーケンスの間は、LCK[15:0] の値を変更することはできません。**

ロックシーケンス中にエラーが発生すると、ロックは中止されます。

ポートの任意のビットの最初のロックシーケンスの後、次の MCU リセットまたはペリフェラルリセットまでは、LCKK ビットのいかなる読出しアクセスに対しても、“1”が返されます。

ビット 15:0 **LCK[15:0]**：ポート x ロック用 I/O ピン y（y = 15〜0）

これらのビットは読出し／書込みができますが、書き込めるのは LCKK ビットが“0”のときだけです。

0：ポート設定はロックされません。

1：ポート設定はロックされます。

9.4.9 GPIO オルタネート機能下位レジスタ（GPIOx_AFRL）（x = A ~ G）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL7 [3:0]				AFSEL6 [3:0]				AFSEL5 [3:0]				AFSEL4 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL3 [3:0]				AFSEL2 [3:0]				AFSEL1 [3:0]				AFSEL0 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **AFSEL[7:0][3:0]**：ポート x I/O ピン y（y = 7 ~ 0）に対するオルタネート機能の選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

9.4.10 GPIO オルタネート機能上位レジスタ（GPIOx_AFRH）（x = A ~ G）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFSEL15 [3:0]				AFSEL14 [3:0]				AFSEL13 [3:0]				AFSEL12 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFSEL11 [3:0]				AFSEL10 [3:0]				AFSEL9 [3:0]				AFSEL8 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **AFSEL[15:8][3:0]**：ポート x I/O ピン y（y = 15 ~ 8）に対するオルタネート機能の選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

0000：AF0
0001：AF1
0010：AF2
0011：AF3
0100：AF4
0101：AF5
0110：AF6
0111：AF7
1000：AF8
1001：AF9
1010：AF10
1011：AF11
1100：AF12
1101：AF13
1110：AF14
1111：AF15

9.4.11 GPIO ポート･ビット･リセット･レジスタ（GPIOx_BRR）（x = A ~ G）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BR[15:0]**：ポート x I/O ピン y（y = 15 ~ 0）のリセット

これらのビットは書込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0：対応する ODx ビットの値は変化しません。

1：対応する ODx ビットをリセットします。

9.4.12 GPIO レジスタマップ

次の表に、GPIO レジスタ･マップとリセット値を示します。

表 57. GPIO レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**GPIOA_MODER**	MODE15 [1:0]		MODE14 [1:0]		MODE13 [1:0]		MODE12 [1:0]		MODE11 [1:0]		MODE10 [1:0]		MODE9 [1:0]		MODE8 [1:0]		MODE7 [1:0]		MODE6 [1:0]		MODE5 [1:0]		MODE4 [1:0]		MODE3 [1:0]		MODE2 [1:0]		MODE1 [1:0]		MODE0 [1:0]	
	リセット値	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x00	**GPIOB_MODER**	MODE15 [1:0]		MODE14 [1:0]		MODE13 [1:0]		MODE12 [1:0]		MODE11 [1:0]		MODE10 [1:0]		MODE9 [1:0]		MODE8 [1:0]		MODE7 [1:0]		MODE6 [1:0]		MODE5 [1:0]		MODE4 [1:0]		MODE3 [1:0]		MODE2 [1:0]		MODE1 [1:0]		MODE0 [1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
0x00	**GPIOx_MODER**（x = **C ~ G**）	MODE15 [1:0]		MODE14 [1:0]		MODE13 [1:0]		MODE12 [1:0]		MODE11 [1:0]		MODE10 [1:0]		MODE9 [1:0]		MODE8 [1:0]		MODE7 [1:0]		MODE6 [1:0]		MODE5 [1:0]		MODE4 [1:0]		MODE3 [1:0]		MODE2 [1:0]		MODE1 [1:0]		MODE0 [1:0]	
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	**GPIOx_OTYPER**（x = **A ~ G**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**GPIOA_OSPEEDR**	OSPEED15 [1:0]		OSPEED14 [1:0]		OSPEED13 [1:0]		OSPEED12 [1:0]		OSPEED11 [1:0]		OSPEED10 [1:0]		OSPEED9 [1:0]		OSPEED8 [1:0]		OSPEED7 [1:0]		OSPEED6 [1:0]		OSPEED5 [1:0]		OSPEED4 [1:0]		OSPEED3 [1:0]		OSPEED2 [1:0]		OSPEED1 [1:0]		OSPEED0 [1:0]	
	リセット値	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**GPIOx_OSPEEDR**（x = **A ~ G**）	OSPEED15 [1:0]		OSPEED14 [1:0]		OSPEED13 [1:0]		OSPEED12 [1:0]		OSPEED11 [1:0]		OSPEED10 [1:0]		OSPEED9 [1:0]		OSPEED8 [1:0]		OSPEED7 [1:0]		OSPEED6 [1:0]		OSPEED5 [1:0]		OSPEED4 [1:0]		OSPEED3 [1:0]		OSPEED2 [1:0]		OSPEED1 [1:0]		OSPEED0 [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**GPIOA_PUPDR**	PUPD15 [1:0]		PUPD14 [1:0]		PUPD13 [1:0]		PUPD12 [1:0]		PUPD11 [1:0]		PUPD10 [1:0]		PUPD9 [1:0]		PUPD8 [1:0]		PUPD7 [1:0]		PUPD6 [1:0]		PUPD5 [1:0]		PUPD4 [1:0]		PUPD3 [1:0]		PUPD2 [1:0]		PUPD1 [1:0]		PUPD0 [1:0]	
	リセット値	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**GPIOB_PUPDR**	PUPD15 [1:0]		PUPD14 [1:0]		PUPD13 [1:0]		PUPD12 [1:0]		PUPD11 [1:0]		PUPD10 [1:0]		PUPD9 [1:0]		PUPD8 [1:0]		PUPD7 [1:0]		PUPD6 [1:0]		PUPD5 [1:0]		PUPD4 [1:0]		PUPD3 [1:0]		PUPD2 [1:0]		PUPD1 [1:0]		PUPD0 [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0x0C	**GPIOx_PUPDR**（x = ）	PUPD15 [1:0]		PUPD14 [1:0]		PUPD13 [1:0]		PUPD12 [1:0]		PUPD11 [1:0]		PUPD10 [1:0]		PUPD9 [1:0]		PUPD8 [1:0]		PUPD7 [1:0]		PUPD6 [1:0]		PUPD5 [1:0]		PUPD4 [1:0]		PUPD3 [1:0]		PUPD2 [1:0]		PUPD1 [1:0]		PUPD0 [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**GPIOx_IDR**（x = **A ~ G**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
	リセット値																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x14	**GPIOx_ODR**（x = **A ~ G**）	Res.		Res.		Res.		Res.		Res.		Res.		Res.		Res.		OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 57. GPIO レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x18	**GPIOx_BSRR**（x = **A ~ G**）	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0	BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**GPIOx_LCKR**（x = **A ~ G**）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LCKK	LCK15	LCK14	LCK13	LCK12	LCK11	LCK10	LCK9	LCK8	LCK7	LCK6	LCK5	LCK4	LCK3	LCK2	LCK1	LCK0
	リセット値																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**GPIOx_AFRL**（x = **A ~ G**）	7[3:0]				6[3:0]				5[3:0]				4[3:0]				3[3:0]				2[3:0]				1[3:0]				0[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	**GPIOx_AFRH**（x = **A ~ G**）	15[3:0]				14[3:0]				13[3:0]				12[3:0]				11[3:0]				10[3:0]				9[3:0]				8[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**GPIOx_BRR**（x = **A ~ G**）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

10 システム設定コントローラ（SYSCFG）

10.1 SYSCFG の主な機能

STM32G4 シリーズ デバイスは設定レジスタ群を持っています。システム設定コントローラの主な目的は次の通りです。

- メモリ領域の再配置
- 外部割込みラインのGPIO への接続の管理
- 堅牢性の管理機能
- CCM RAM 書込み保護およびソフトウェア消去の設定
- FPU 割込みの設定
- I/O アナログスイッチの 電圧ブースタおよび幾つかのI/Oに割り当てられた I^2C 高速モードプラス駆動機能の有効化／無効化

10.2 SYSCFG レジスタ

10.2.1 SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP）

このレジスタは、メモリ再配置時の特定の設定に使用されます。

アドレスオフセット：0x00

リセット値：0x0000 000X（X は BOOT0 ピンおよび BOOT1 オプションビットで選択されたメモリモード）

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

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **FB_MODE**：Flashバンクモード選択

0：Flash バンク 1 は 0x0800 0000 にマッピングされ（0x0000 0000 にエイリアスされます(1)）、Flash バンク 2 は 0x0804 0000 にマッピングされます（0x0008 0000 にエイリアスされます）。

1：Flash バンク 2 は 0x0800 0000 にマッピングされ（0x0000 0000 にエイリアスされます (1)）、Flash バンク 1 は 0x0804 0000 にマッピングされます（0x0008 0000 にエイリアスされます）。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **MEM_MODE**：メモリマッピング選択

これらのビットによって、アドレス 0x0000 0000 のメモリ内部マッピングを制御します。これらのビットを用いて、ソフトウェアによる物理的な再配置を選択すると、BOOT ピンおよびオプションビット設定はバイパスされます。リセット後、これらのビットは、BOOT0 ピン（または nSWBOOT0 オプションビット）および BOOT1 オプションビットによって選択された値になります。

000：メイン Flash メモリは 0x00000000 にマッピングされます（1）。

001：システム Flash メモリは 0x00000000 にマッピングされます。

010：FSMC メモリ

011：SRAM1 は 0x00000000 にマッピングされます。

100：QUADSPI メモリは 0x0000 0000 にマッピングされます。

101：予約済みです。

111：予約済みです。

1. BFB2 ビットがセットされると、システムメモリのエイリアスは @0x0000 0000 のままとなります。

注： **FSMC がアドレス 0x0000 0000 に再マッピングされた場合、バンク 1 メモリ･コントローラ先頭の 2 つの領域（バンク 1 NOR/PSRAM 1 および NOR/PSRAM 2）のみ再マッピングできます。再割り当てモードでは、CPU がシステムバスではなく ICode バスを通じて外部メモリにアクセスできるため、パフォーマンスを高めることができます。**

10.2.2 SYSCFG 設定レジスタ 1（SYSCFG_CFGR1）

アドレス･オフセット：0x04

リセット値：0x7C00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FPU_IE[5..0]						Res	Res	I2C4_FMP	I2C3_FMP	I2C2_FMP	I2C1_FMP	I2C_PB9_FMP	I2C_PB8_FMP	I2C_PB7_FMP	I2C_PB6_FMP
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	Res	Res	Res	Res	Res	ANASWVDD	BOOSTEN	Res	Res	Res	Res	Res	Res	Res	Res
						rw	rw								

ビット 31:26 **FPU_IE[5..0]**：浮動小数点演算装置割込みイネーブルビット

FPU_IE[5]：不正確割込みイネーブル

FPU_IE[4]：入力非正規割込みイネーブル

FPU_IE[3]：オーバーフロー割込みイネーブル

FPU_IE[2]：アンダーフロー割込みイネーブル

FPU_IE[1]：0 分周割込みイネーブル

FPU_IE[0]：無効操作割込みイネーブル

ビット 25:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **I2C4_FMP**：高速モードプラス駆動機能有効化

このビットは、AF 選択ビットを通じて選択された I2C4 ピンの Fm+ 駆動モードを有効にします。

0：AF 選択ビットを通じて選択された I2C4 ピンで、Fm+ モードは無効です。

1：AF 選択ビットを通じて選択された I2C4 ピンで、Fm+ モードが有効になります。

ビット 22 **I2C3_FMP**：I2C3 高速モードプラス駆動機能有効化

このビットは、AF 選択ビットを通じて選択された I2C3 ピンの Fm+ 駆動モードを有効にします。

0：AF 選択ビットを通じて選択された I2C3 ピンで、Fm+ モードは無効です。

1：AF 選択ビットを通じて選択された I2C3 ピンで、Fm+ モードが有効になります。

ビット 21 **I2C2_FMP**：I2C2 高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C2 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C2 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C2 ピンで、Fm+ モードが有効になります。

ビット 20 **I2C1_FMP**：I2C1 高速モードプラス駆動機能有効化
このビットは、AF 選択ビットを通じて選択された I2C1 ピンの Fm+ 駆動モードを有効にします。
0：AF 選択ビットを通じて選択された I2C1 ピンで、Fm+ モードは無効です。
1：AF 選択ビットを通じて選択された I2C1 ピンで、Fm+ モードが有効になります。

ビット 19 **I2C_PB9_FMP**：PB9 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB9 の Fm+ 駆動モードを有効にします。
0：PB9 ピンは標準モードで動作します。
1：PB9 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 18 **I2C_PB8_FMP**：PB8 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB8 の Fm+ 駆動モードを有効にします。
0：PB8 ピンは標準モードで動作します。
1：PB8 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 17 **I2C_PB7_FMP**：PB7 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB7 の Fm+ 駆動モードを有効にします。
0：PB7 ピンは標準モードで動作します。
1：PB7 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 16 **I2C_PB6_FMP**：PB6 の高速モードプラス（Fm+）駆動機能有効化
このビットは、PB6 の Fm+ 駆動モードを有効にします。
0：PB6 ピンは標準モードで動作します。
1：PB6 ピンで Fm+ モードを有効にし、速度制御をバイパスします。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **ANASWVDD**：GPIO アナログスイッチ制御電圧選択
0：I/O アナログ･スイッチには、VDDA から給電されるか、ブースタがオンのときはブースタから給電されます。
1：I/O アナログ･スイッチは VDD から給電されます。
ビット 9 の設定については、表 58 を参照してください。

ビット 8 **BOOSTEN**：I/O アナログスイッチ電圧ブースタ有効化
0：I/O アナログ･スイッチは V_{DDA} 電圧から給電されます。これは、高い V_{DDA} 電圧入力で ADC を使用する場合の推奨設定です。
1：I/O アナログ･スイッチは、専用の電圧ブースタ（V_{DD} 電圧から給電）から給電されます。これは、低い V_{DDA} 電圧入力で ADC を使用する場合の推奨設定です。

ビット 7:0 予約済みであり、リセット値に保持する必要があります。

表 58 は、電圧設定に応じてビット 9（ANASWVDD）およびビット 8（BOOSTEN）の設定状態を示します。

注： **GPIO ピンで FM+ モードが有効化されると、（GPIOx_OSPEEDR レジスタの）GPIO の速度設定は無視されます。GPIOx_AFRH レジスタまたは GPIOx_AFRL レジスタを通じて AF 選択を行った後にのみこのレジスタをプログラムします。**

表 58. BOOSTEN および ANASWVDD のセット／リセット

VDD	VDDA	BOOSTEN	ANASWVDD
-	> 2.4 V	0	0
> 2.4 V	< 2.4 V	0	1
< 2.4 V	< 2.4 V	1	0

10.2.3 SYSCFG 外部割込み設定レジスタ 1（SYSCFG_EXTICR1）

アドレス･オフセット：0x08

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3 [3:0]				EXTI2 [3:0]				EXTI1 [3:0]				EXTI0 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI3[3:0]**：EXTI 3 設定ビット

これらのビットは、EXTI3 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[3] ピン
0001：PB[3] ピン
0010：PC[3] ピン
0011：PD[3] ピン
0100：PE[3] ピン
0101：PF[3] ピン
0110：PG[3] ピン

ビット 11:8 **EXTI2[3:0]**：EXTI 2 設定ビット

これらのビットは、EXTI2 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[2] ピン
0001：PB[2] ピン
0010：PC[2] ピン
0011：PD[2] ピン
0100：PE[2] ピン
0101：PF[2] ピン
0110：PG[2] ピン

ビット 7:4 **EXTI1[3:0]**：EXTI 1 設定ビット

これらのビットは、EXTI1 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[1] ピン
0001：PB[1] ピン
0010：PC[1] ピン
0011：PD[1] ピン
0100：PE[1] ピン
0101：PF[1] ピン
0110：PG[1] ピン

ビット 3:0 **EXTI0[3:0]**：EXTI 0 設定ビット

これらのビットは、EXTI0 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[0] ピン
0001：PB[0] ピン
0010：PC[0] ピン
0011：PD[0] ピン
0100：PE[0] ピン
0101：PF[0] ピン
0110：PG[0] ピン

10.2.4 SYSCFG 外部割込み設定レジスタ 2（SYSCFG_EXTICR2）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7 [3:0]				EXTI6 [3:0]				EXTI5 [3:0]				EXTI4 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI7[3:0]**：EXTI 7 設定ビット

これらのビットは、EXTI7 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[7] ピン
0001：PB[7] ピン
0010：PC[7] ピン
0011：PD[7] ピン
0100：PE[7] ピン
0101：PF[7] ピン
0110：PG[7] ピン

ビット 11:8 **EXTI6[3:0]**：EXTI 6 設定ビット

これらのビットは、EXTI6 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[6] ピン
0001：PB[6] ピン
0010：PC[6] ピン
0011：PD[6] ピン
0100：PE[6] ピン
0101：PF[6] ピン
0110：PG[6] ピン

ビット 7:4 **EXTI5[3:0]**：EXTI 5 設定ビット

これらのビットは、EXTI5 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[5] ピン
0001：PB[5] ピン
0010：PC[5] ピン
0011：PD[5] ピン
0100：PE[5] ピン
0101：PF[5] ピン
0110：PG[5] ピン

ビット 3:0 **EXTI4[3:0]**：EXTI 4 設定ビット

これらのビットは、EXTI4 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[4] ピン
0001：PB[4] ピン
0010：PC[4] ピン
0011：PD[4] ピン
0100：PE[4] ピン
0101：PF[4] ピン
0110：PG[4] ピン

注：　上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。

10.2.5 SYSCFG 外部割込み設定レジスタ 3（SYSCFG_EXTICR3）

アドレス･オフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11 [3:0]				EXTI10 [3:0]				EXTI9 [3:0]				EXTI8 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI11[3:0]**：EXTI 11 設定ビット

これらのビットは、EXTI11 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[11] ピン
0001：PB[11] ピン
0010：PC[11] ピン
0011：PD[11] ピン
0100：PE[11] ピン
0101：PF[11] ピン

ビット 11:8 **EXTI10[3:0]**：EXTI 10 設定ビット

これらのビットは、EXTI10外部割込みのソース入力をソフトウェアで設定します。

0000：PA[10] ピン
0001：PB[10] ピン
0010：PC[10] ピン
0011：PD[10] ピン
0100：PE[10] ピン
0101：PF[10] ピン
0110：PG[10] ピン

ビット 7:4 **EXTI9[3:0]**：EXTI 9 設定ビット

これらのビットは、EXTI9 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[9] ピン
0001：PB[9] ピン
0010：PC[9] ピン
0011：PD[9] ピン
0100：PE[9] ピン
0101：PF[9] ピン
0110：PG[9] ピン

ビット 3:0 **EXTI8[3:0]**：EXTI 8 設定ビット

これらのビットは、EXTI8 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[8] ピン
0001：PB[8] ピン
0010：PC[8] ピン
0011：PD[8] ピン
0100：PE[8] ピン
0101：PF[8] ピン
0110：PG[8] ピン

注：　上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。

10.2.6 SYSCFG 外部割込み設定レジスタ 4（SYSCFG_EXTICR4）

アドレス･オフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15 [3:0]				EXTI14 [3:0]				EXTI13 [3:0]				EXTI12 [3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EXTI15[3:0]**：EXTI 15 設定ビット

これらのビットは、EXTI15 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[15] ピン
0001：PB[15] ピン
0010：PC[15] ピン
0011：PD[15] ピン
0100：PE[15] ピン
0101：PF[15] ピン

ビット 11:8 **EXTI14[3:0]**：EXTI 14 設定ビット

これらのビットは、EXTI14 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[14] ピン
0001：PB[14] ピン
0010：PC[14] ピン
0011：PD[14] ピン
0100：PE[14] ピン
0101：PF[14] ピン

ビット 7:4 **EXTI13[3:0]**：EXTI 13 設定ビット

これらのビットは、EXTI13 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[13] ピン
0001：PB[13] ピン
0010：PC[13] ピン
0011：PD[13] ピン
0100：PE[13] ピン
0101：PF[13] ピン

ビット 3:0 **EXTI12[3:0]**：EXTI 12 設定ビット

これらのビットは、EXTI12 外部割込みのソース入力ピンを選択するために、ソフトウェアで設定します。

0000：PA[12] ピン
0001：PB[12] ピン
0010：PC[12] ピン
0011：PD[12] ピン
0100：PE[12] ピン
0101：PF[12] ピン

注：　上記レジスタの一部の I/O ピンは、小型のパッケージでは使用できない場合があります。

10.2.7 SYSCFG CCM SRAM 制御およびステータスレジスタ（SYSCFG_SCSR）

アドレスオフセット：0x18

システム･リセット値：0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CCMBSY**：消去操作による CCM SRAM ビジー

0：CCM SRAM の消去操作が進行中ではありません。

1：CCM SRAM の消去操作が進行中です。

ビット 0 **CCMER**：CCM SRAM の消去

このビットをセットすると、ハードウェアの CCM SRAM 消去操作が開始されます。このビットは CCM SRAM 消去操作の終了時に自動的にクリアされます。

注： このビットは、書込み保護されており、正しいキーシーケンスが SYSCFG_SKR レジスタに書き込まれた後のみ書込み可能です。

10.2.8 SYSCFG 設定レジスタ 2（SYSCFG_CFGR2）

アドレス･オフセット：0x1C

システム･リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	Res	Res	Res	Res	Res	Res	SPF	Res	Res	Res	Res	ECCL	PVDL	SPL	CLL
							rc_w1					rs	rs	rs	rs

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **SPF**：SRAM1 と CCM SRAM のパリティ･エラー･フラグ

このビットは、SRAM1 または CCM SRAM のパリティエラーが検出されたときに、ハードウェアによってセットされます。ソフトウェアで「1」を書き込むことによってクリアされます。

0：パリティ･エラーは検出されていません。

1：パリティ･エラーが検出されました。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **ECCL**：ECC ロック

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、Flash ECC エラーの TIM1/8/15/16/17/20 のブレーク入力および HRTIM1 の hrtim_sys_flt 入力への接続を有効にし、ロックするために使用できます。

0 : ECC エラーが、TIM1/8/15/16/17/20 のブレーク入力にも HRTIM1 の hrtim_sys_flt 入力にも接続されません。

1 : ECC エラーが、TIM1/8/15/16/17/20 のブレーク入力と HRTIM1 の hrtim_sys_flt 入力に接続されます。

ビット 2 **PVDL**：PVD ロック･イネーブル･ビット

このビットは、ソフトウェアでセットされ、システム･リセットでのみクリアされます。これは、TIM1/8/15/16/17/20 のブレーク入力および HRTIM1 の hrtim_sys_flt 入力への PVD 接続と、PWR_CR2 レジスタの PVDE と PLS[2:0] を有効にし、ロックするために使用できます。

0 : PVD 割込みが、TIM1/8/15/16/17/20 のブレーク入力にも HRTIM1 の hrtim_sys_flt 入力にも接続されません。PVDE および PLS[2:0] ビットは、アプリケーションでプログラムできます。

1 : PVD 割込みが、TIM1/8/15/16/17/20 のブレーク入力および HRTIM1 の hrtim_sys_flt 入力に接続されます。PVDE および PLS[2:0] ビットは読出し専用です。

ビット 1 **SPL**：SRAM1 と CCM SRAM のパリティ･ロック･ビット

このビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。これは、SRAM1 または CCM SRAM パリティエラー信号の TIM1/8/15/16/17/20 のブレーク入力および HRTIM1 の hrtim_sys_flt 入力への接続を有効にし、ロックするために使用できます。

0 : CCM SRAM パリティ･エラー信号が、TIM1/8/15/16/17/20 のブレーク入力にも HRTIM1 の hrtim_sys_flt 入力にも接続されません。

1 : CCM SRAM パリティ･エラー信号が、TIM1/8/15/16/17/20 のブレーク入力と HRTIM1 の hrtim_sys_flt 入力に接続されます。

ビット 0 **CLL**：Cortex®-M4 LOCKUP （ハードフォルト）出力イネーブル･ビット

このビットは、ソフトウェアでセットされ、システム･リセットでのみクリアされます。これは、FPU 搭載 Cortex®-M4 LOCKUP（ハードフォルト）出力の TIM1/8/15/16/17/20 のブレーク入力および HRTIM1 の hrtim_sys_flt 入力への接続を有効にし、ロックするために使用できます。

0：Cortex®-M4 LOCKUP 出力が、TIM1/8/15/16/17/20 のブレーク入力にも HRTIM1 の hrtim_sys_flt 入力にも接続されません。

1：Cortex®-M4 LOCKUP 出力が、TIM1/8/15/16/17/20 のブレーク入力と HRTIM1 の hrtim_sys_flt 入力に接続されます。

10.2.9 SYSCFG CCM SRAM 書込み保護レジスタ（SYSCFG_SWPR）

アドレスオフセット：0x20

システム･リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット 31:0 **PxWP**（x = 0～31）：CCM SRAM ページ x 書込み保護

これらのビットは、ソフトウェアでセットされ、システムリセットでのみクリアされます。

0：CCM SRAM ページ x の書込み保護は無効です。

1：CCM SRAM ページ x の書込み保護は有効です。

10.2.10 SYSCFG CCM SRAM キーレジスタ（SYSCFG_SKR）

アドレスオフセット：0x24

システム･リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY[7:0]**：ソフトウェア消去に対する CCM SRAM 書込み保護キー

SYSCFG_SCSR レジスタの CCMER ビットの書込み保護をアンロックするには、次の手順が必要です。

1. Key[7:0] に「0xCA」を書き込みます。
2. Key[7:0] に「0x53」を書き込みます。

誤ったキーを書き込むと、書込み保護が再度アクティブになります。

10.2.11 SYSCFG レジスタマップ

次の表に、SYSCFG レジスタ･マップとリセット値を示します。

表 59. SYSCFG レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	SYSCFG_ MEMRMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FB_MODE	Res.	Res.	Res.	Res.	Res.	MEM_ MODE		
	リセット値																								0						x	x	x
0x04	SYSCFG_CFGR1	FPU_IE[5..0]						Res.	Res.	I2C4_FMP	I2C3_FMP	I2C2_FMP	I2C1_FMP	I2C_PB9_FMP	I2C_PB8_FMP	I2C_PB7_FMP	I2C_PB6_FMP	Res.	Res.	Res.	Res.	Res.	Res.	ANASWVDD	BOOSTEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FWDIS
	リセット値	0	1	1	1	1	1			0	0	0	0	0	0	0	0							0	0					0	0	0	1
0x08	SYSCFG_EXTICR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI3 [3:0]				EXTI2 [3:0]				EXTI1 [3:0]				EXTI0 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	SYSCFG_EXTICR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI7 [3:0]				EXTI6 [3:0]				EXTI5 [3:0]				EXTI4 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	SYSCFG_EXTICR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI11 [3:0]				EXTI10 [3:0]				EXTI9 [3:0]				EXTI8 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	SYSCFG_EXTICR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EXTI15 [3:0]				EXTI14 [3:0]				EXTI13 [3:0]				EXTI12 [3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	SYSCFG_SCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCMBSY	CCMER
	リセット値																															0	0
0x1C	SYSCFG_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SPF	Res.	Res.	Res.	Res.	ECCL	PVDL	SPL	CLL
	リセット値																								0					0	0	0	0
0x20	SYSCFG_SWPR	P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP	P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x24	SYSCFG_SKR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	キー							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

11 ペリフェラル相互接続マトリックス

11.1 概要

いくつかのペリフェラルには相互に直接接続できるものがあります。

これにより、ペリフェラル間での自主的な通信や同期が可能になり、CPU リソースと消費電力を低減できます。

さらに、これらのハードウェア接続によってソフトウェアの遅延をなくし、予測可能なシステム設計が可能になります。

ペリフェラルに応じて、これらの相互接続は RUN、SLEEP、低電力 RUN および SLEEP、STOP 0 および STOP 1 のモードで動作可能です。

11.2 接続の一覧

表 60. STM32G4 シリーズペリフェラル相互接続マトリックス(1) (2)

転送元	転送先																																				
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM8	TIM15	TIM16	TIM17	TIM20	LPTIM1	HRTIM	ADC1	ADC2	ADC3	ADC4	ADC5	OPAMP1	OPAMP2	OPAMP3	OPAMP4	OPAMP5	OPAMP6	DAC1	DAC2	DAC3	DAC4	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7	IRTIM
TIM1	-	1	1	1	1	-	-	1	1	-	-	1	-	10 14	19	19	19	19	19	-	-	-	-	-	-	-	-	20	-	15	15	15	15	15	15	15	-
TIM2	1	-	1 2	-	1 2	-	-	1	1	-	-	1	-	10	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	15	15	15	-	15	15	-	-
TIM3	1	1 2	-	1 2	1 2	-	-	1	1	-	-	1	-	10	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	15	15	15	15	15	-	15	-
TIM4	1	1 2	1 2	-	1	-	-	1	1	-	-	1	-	-	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	-	-	-	-	-
TIM5	1	1 2	1	1 2	-	-	-	1	1	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
TIM6	-	-	-	-	-	-	-	-	-	-	-	-	-	10 13	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	-	-	-	-	-
TIM7	-	-	-	-	-	-	-	-	-	-	-	-	-	10 12	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	-	-	-	-	-
TIM8	1	1	1	1	1	-	-	-	1	-	-	1	-	-	19	19	19	19	19	-	-	-	-	-	-	20	20	-	20	15	15	15	15	15	15	15	-
TIM15	1	1	1	1	1	-	-	1	-	-	-	1	-	10	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	15	-	15	15	-
TIM16	1	1	1	1	1	-	-	1	1	-	-	1	-	12 13	19	19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16
TIM17	1	1	1	1	1	-	-	1	1	-	-	1	-	12 13	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	16
TIM20	1	1	1	1	1	-	-	1	1	-	-	1	-	-	19	19	19	19	19	-	-	-	-	-	-	-	-	-	-	15	15	15	15	15	15	15	-
LPTIM1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	19	19	19	19	19	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
HRTIM	1	1	1	1	1	-	-	1	1	-	-	1	-	-	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	-	-	-	-	-
ADC1	2	-	2	-	-	-	-	2	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ADC2	2	-	2	-	-	-	-	2	-	-	-	-	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ADC3	2	-	-	-	-	-	-	2	-	-	-	2	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ADC4	2	-	-	-	-	-	-	2	-	-	-	2	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
ADC5	2	-	-	-	-	-	-	2	-	-	-	2	-	10	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
温度センサ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21	-	-	-	21	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VBAT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21	-	21	-	21	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
VREFINT	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21	-	21	21	21	-	-	-	-	-	-	-	-	-	-	23	23	23	23	23	23	23	-
OPAMP1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OPAMP2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OPAMP3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	21 24	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OPAMP4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	-	-	21 24	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OPAMP5	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
OPAMP6	-	-	-	-	-	-	-	-	-	-	-	-	-	-	21 24	21 24	-	21 24	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

表 60. STM32G4 シリーズペリフェラル相互接続マトリックス(1) (2) （続き）

転送元	転送先																																				
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM8	TIM15	TIM16	TIM17	TIM20	LPTIM1	HRTIM	ADC1	ADC2	ADC3	ADC4	ADC5	OPAMP1	OPAMP2	OPAMP3	OPAMP4	OPAMP5	OPAMP6	DAC1	DAC2	DAC3	DAC4	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7	IRTIM
DAC1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22 23	22 23	22 23	22 23	22 23	-	-	-
DAC2	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22 23	22 23	-
DAC3	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22 24	-	22 24	-	-	22 24	-	-	-	-	22 23	22 23	22 23	22 23	-	-	-	-
DAC4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	22 24	22 24	-	-	-	-	-	-	-	-	-	22 23	22 23	22 23	-
HSE	-	-	-	-	-	-	-	-	-	4	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LSE	-	2	-	-	4	-	-	-	4	4	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
HSI16	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
LSI	-	-	-	-	4	-	-	-	-	4	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
MCO	-	-	-	-	-	-	-	-	-	4	4	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
EXTI	-	-	-	-	-	-	-	-	-	-	-	-	-	-	19	19	19	19	19	-	-	-	-	-	-	20	20	20	20	-	-	-	-	-	-	-	-
RTC	-	-	-	-	4	-	-	-	-	4	4	-	17	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP1	2 3 4 8 9	2 3 4 5 7	2 3 4 5	2 4 5	2 4 5	-	-	2 3 4 8 9	3 4 8	3 8	3 8	2 3 4 8 9	17	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP2	2 3 4 8 9	2 3 4 5 7	2 3 4 5	2 4 5	2 4 5	-	-	2 3 4 8 9	3 5 8	3 8	3 8	2 3 4 8 9	17	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP3	2 3 4 8 9	2 3 4 5	2 3 4 5 6	2 4 5	2 4 5	-	-	2 3 4 8 9	3 5 8	3 8	3 8	2 3 4 8 9	17	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP4	2 3 4 8 9	2 3 4 5 6	2 3 4 5	2 4 5	2 4 5	-	-	2 3 4 8 9	3 8	3 8	3 8	2 3 4 8 9	17	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP5	2 3 8 9	2 3 4	2 3 4 5	2 4 5 6	2 4 5	-	-	2 3 8 9	3 4 8	3 8	3 4 8	2 3 8 9	-	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
COMP6	2 3 8 9	2 3 5	2 3 4 5	2 4 5	2 4 5	-	-	2 3 8 9	3 5 8	3 5 8	3 8	2 3 8 9	-	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

表 60. STM32G4 シリーズペリフェラル相互接続マトリックス(1) (2) （続き）

転送元	転送先																																				
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM6	TIM7	TIM8	TIM15	TIM16	TIM17	TIM20	LPTIM1	HRTIM	ADC1	ADC2	ADC3	ADC4	ADC5	OPAMP1	OPAMP2	OPAMP3	OPAMP4	OPAMP5	OPAMP6	DAC1	DAC2	DAC3	DAC4	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7	IRTIM
COMP7	2 3 8 9	2 3	2 3 4 5	2 4 5	2 4 5	-	-	2 3 8 9	3 4 5 8	3 8	3 8	2 3 8 9	-	10 11	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
SYST ERR	18	-	-	-	-	-	-	18	18	18	18	18	-	18	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

1. 表内の数字は、セクション 11.3：相互接続の詳細に詳細が説明されている、対応相互接続番号にリンクしています。
2. 灰色のセルにある「-」の記号は、相互接続がないことを示します。

11.3 相互接続の詳細

11.3.1 タイマ（TIMx、HRTIM）からタイマ（TIMx）へ

表 61. 相互接続 1

タイマ入力トリガ信号	タイマ入力トリガ·ソース割り当て							
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM15	TIM20
timx_itr0	-	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo
timx_itr1	tim2_trgo	-	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo
timx_itr2	tim3_trgo	tim3_trgo	-	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo
timx_itr3	tim4_trgo	tim4_trgo	tim4_trgo	-	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo
timx_itr4	tim5_trgo	tim5_trgo	tim5_trgo	tim5_trgo	-	tim5_trgo	tim5_trgo	tim5_trgo
timx_itr5	tim8_trgo	tim8_trgo	tim8_trgo	tim8_trgo	tim8_trgo	-	tim8_trgo	tim8_trgo
timx_itr6	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	-	tim15_trgo
timx_itr7	tim16_oc	tim16_oc	tim16_oc	tim16_oc	tim16_oc	tim16_oc	xtim16_oc	tim16_oc
timx_itr8	tim17_oc	tim17_oc	tim17_oc	tim17_oc	tim17_oc	tim17_oc	tim17_oc	tim17_oc
timx_itr9	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo	-
timx_itr10	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2	hrtim_out_scout2

HRTIM バースト操作は、他の汎用タイマからのオンチップ·イベントによってトリガできます。

バーストモードコントローラのカウンタは、表 62 に示す汎用タイマをクロックとしても動作できます。

表 62. 相互接続 12

HRTIM バーストモードトリガイベント／クロック信号	HRTIM バーストモードトリガイベント／クロック信号の割り当て
hrtim_bm_trg	tim7_trgo
hrtim_bm_ck1	tim16_oc
hrtim_bm_ck2	tim17_oc
hrtim_bm_ck3	tim7_trgo‘

表 63. 相互接続 13

HRTIM 更新イネーブル信号	HRTIM 更新イネーブル割り当て
hrtim_upd_en1	tim16_oc
hrtim_upd_en2	tim17_oc
hrtim_upd_en3	tim6_oc

HRTIM は、表 64 に示す外部ソースによって同期可能です。

表 64. 相互接続 14

HRTIM 同期信号	HRTIM 同期信号割り当て
hrtim_in_sync2	tim1_trgo
hrtim_in_sync3	HRTIM_SCIN

タイマの同期や連携した動作のために、いくつかの TIMx タイマを内部で相互リンクすることができます。

マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

機能の説明については、次を参照してください。セクション 28.3.30 : タイマの同期 を参照してください。

同期のモードについては次のセクションに詳細が記載されています。

- セクション 28.3.30 : タイマの同期（高機能制御タイマ（TIM1/TIM1/TIM20）について）
- セクション 29.4.23 : タイマの同期（汎用タイマ（TIM2/TIM3/TIM4/TIM5）について）
- セクション 30.4.26 : タイマ同期（TIM15 のみ）（汎用タイマ（TIM15）について）

トリガ信号

出力（マスタから）は、設定可能なタイマイベントに従った信号 TIMx_TRGO（TIM1/TIM8/TIM20 では TIMx_TRGO2）で行われます。

入力（スレーブへ）は信号 TIMx_ITRx で行われます。

TIM1/TIM8/TIM20 の入力信号および出力信号は、図 269 : 高機能制御タイマのブロック図に示されています。

使用可能なマスタ／スレーブ接続を以下に示します。

- 表 251 : TIMx 内部トリガ接続

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.2 タイマ（TIMx、HRTIM）および EXTI から ADC（ADCx）へ

表 65. 相互接続 19

ADC トリガ選択 EXTSEL[4:0] または JEXTSEL[4:0]	ADC トリガ信号割り当て			
	ADC1/2		ADC3/4/5	
	レギュラ	インジェクト	レギュラ	インジェクト
0	tim1_cc1	tim1_trgo	tim3_cc1	tim1_trgo
1	tim1_cc2	tim1_cc4	tim2_cc3	tim1_cc4
2	tim1_cc3	tim2_trgo	tim1_cc3	tim2_trgo
3	tim2_cc2	tim2_cc1	tim8_cc1	tim8_cc2
4	tim3_trgo	tim3_cc4	tim3_trgo	tim4_cc3
5	tim4_cc4	tim4_trgo	exti2	tim4_trgo
6	exti11	exti15	tim4_cc1	tim4_cc4
7	tim8_trgo	tim8_cc4	tim8_trgo	tim8_cc4
8	tim8_trgo2	tim1_trgo2	tim8_trgo2	tim1_trgo2
9	tim1_trgo	tim8_trgo	tim1_trgo	tim8_trgo
10	tim1_trgo2	tim8_trgo2	tim1_trgo2	tim8_trgo2
11	tim2_trgo	tim3_cc3	tim2_trgo	tim1_cc3
12	tim4_trgo	tim3_trgo	tim4_trgo	tim3_trgo
13	tim6_trgo	tim3_cc1	tim6_trgo	exti3
14	tim15_trgo	tim6_trgo	tim15_trgo	tim6_trgo
15	tim3_cc4	tim15_trgo	tim2_cc1	tim15_trgo
16	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo
17	tim20_trgo2	tim20_trgo2	tim20_trgo2	tim20_trgo2
18	tim20_cc1	tim20_cc4	tim20_cc1	tim20_cc2
19	tim20_cc2	hrtim_adc_trg2	hrtim_adc_trg2	hrtim_adc_trg2
20	tim20_cc3	hrtim_adc_trg4	hrtim_adc_trg4	hrtim_adc_trg4
21	hrtim_adc_trg1	hrtim_adc_trg5	hrtim_adc_trg1	hrtim_adc_trg5
22	hrtim_adc_trg3	hrtim_adc_trg6	hrtim_adc_trg3	hrtim_adc_trg6
23	hrtim_adc_trg5	hrtim_adc_trg7	hrtim_adc_trg5	hrtim_adc_trg7
24	hrtim_adc_trg6	hrtim_adc_trg8	hrtim_adc_trg6	hrtim_adc_trg8
25	hrtim_adc_trg7	hrtim_adc_trg9	hrtim_adc_trg7	hrtim_adc_trg9
26	hrtim_adc_trg8	hrtim_adc_trg10	hrtim_adc_trg8	hrtim_adc_trg10
27	hrtim_adc_trg9	TIM16_CC1	hrtim_adc_trg9	hrtim_adc_trg1
28	hrtim_adc_trg10	-	hrtim_adc_trg10	hrtim_adc_trg3
29	lptim_out	lptim_out	lptim_out	lptim_out
30	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo
31	-	-	-	-

タイマ（TIMx、HRTIM）は、ADC トリガイベントを生成するために使用できます。

TIMx 同期については、セクション 28.3.31：ADC トリガ（TIM1/TIM8）で説明されています。

ADC 同期については、セクション 21.4.18：外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN、JEXTSEL、JEXTEN）を参照してください。

トリガ信号

出力（タイマから）は、信号 TIMx_TRGO、TIMx_TRGO2、または TIMx_CCx イベントで行われます。

入力（ADC へ）は信号 EXT[15:0]、JEXT[15:0] で行われます。

タイマと ADC の間の接続は次のモードで提供されます。

- 表 163：ADC1/2 - レギュラ･チャネルの外部トリガ
- 表 164：ADC1/2 - インジェクトチャネルの外部トリガ

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.3 ADC（ADCx）からタイマ（TIMx、HRTIM）へ

表 73：相互接続 2と表 81.：相互接続 10を参照してください。

ADC アナログウォッチドッグは、デジタル電源アプリケーション（ADC を用いたサイクルごとの電流調整）のために TIM1/8/20 に接続されています。

ADC アナログウォッチドッグ設定の説明については、セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

タイマのトリガ設定については、セクション 28.3.6：外部トリガ入力 を参照してください。

トリガ信号

（ADC からの）出力は ADCn_AWDx_OUT 信号、（タイマへの）入力は TIMx_ETR 信号（外部トリガ）または hrtim_eevx[4:1] です。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.4 タイマ（TIMx、HRTIM）および EXTI から DAC（DACx）へ

表 66. 相互接続 20

DAC トリガ選択（TSELx[3:0]、STRSTTRIGSELx[3:0]）	DAC トリガ信号割り当て							
	DAC1		DAC2		DAC3		DAC4	
	更新／リセット	Inc	更新／リセット	Inc	更新／リセット	Inc	更新／リセット	Inc
0	sw	-	sw	-	sw	-	sw	-
1	tim8_trgo	tim8_trgo	tim8_trgo	tim8_trgo	tim1_trgo	tim1_trgo	tim8_trgo	tim8_trgo
2	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo	tim7_trgo
3	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo
4	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo	tim2_trgo
5	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo	tim4_trgo
6	exti9	exti10	exti9	exti10	exti9	exti10	exti9	exti10
7	tim6_trgo	tim6_trgo	tim6_trgo	tim6_trgo	tim6_trgo	tim6_trgo	tim6_trgo	tim6_trgo
8	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo	tim3_trgo
9	hrtim_dac_reset_trg1	hrtim_step_trig1	hrtim_dac_reset_trg1	hrtim_step_trig1	hrtim_dac_reset_trg1	hrtim_step_trig1	hrtim_dac_reset_trg1	hrtim_step_trig1
10	hrtim_dac_reset_trg2	hrtim_step_trig2	hrtim_dac_reset_trg2	hrtim_step_trig2	hrtim_dac_reset_trg2	hrtim_steptrig_2	hrtim_rst_trig_2	hrtim_step_trig2
11	hrtim_dac_reset_trg3	hrtim_step_trig3	hrtim_dac_reset_trg3	hrtim_step_trig3	hrtim_dac_reset_trg3	hrtim_step_trig3	hrtim_dac_reset_trg3	hrtim_step_trig3
12	hrtim_dac_reset_trg4	hrtim_step_trig4	hrtim_dac_reset_trg4	hrtim_step_trig4	hrtim_dac_reset_trg4	hrtim_step_trig4	hrtim_dac_reset_trg4	hrtim_step_trig4
13	hrtim_dac_reset_trg5	hrtim_step_trig5	hrtim_dac_reset_trg5	hrtim_step_trig5	hrtim_dac_reset_trg5	hrtim_step_trig5	hrtim_dac_reset_trg5	hrtim_step_trig5
14	hrtim_dac_reset_trg6	hrtim_step_trig6	hrtim_dac_reset_trg6	hrtim_step_trig6	hrtim_dac_reset_trg6	hrtim_step_trig6	hrtim_dac_reset_trg6	hrtim_step_trig6
15	hrtim_dac_trg1	-	hrtim_dac_trg2	-	hrtim_dac_trg3	-	hrtim_dac_trg1	-

タイマ（TIMx、HRTIM）および EXTI は、DAC 変換を開始するトリガイベントとして使用できます。

トリガ信号

出力（タイマから）は、対応する DAC 入力に直接接続される 信号 TIMx_TRGOで行われます。

DAC の入力トリガの選択については、セクション 22.4.7 : DAC トリガ選択（シングルおよびデュアルモード）で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.5 HSE、LSE、LSI、HSI16、MCO、RTC からタイマ（TIMx）へ

表 75：相互接続 4 と表 73：相互接続 2 を参照してください。

外部クロック（HSE、LSE）、内部クロック（LSI、HSI16）、マイクロコントローラ出力クロック（MCO）、GPIO、および RTC ウェイクアップ割込みは、タイマ（TIMx）への入力として使用できます。

これによって、HSI16 の較正と LSI オシレータ周波数の精密測定が可能となります。

ロースピード外部（LSE）オシレータが使用されている場合、追加のハードウェア接続は必要ありません。

この機能については、セクション 7.2.16：TIM5/TIM15/TIM16/TIM17 を使用した内部／外部クロックの測定で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.6 RTC、COMPx から低電力タイマ（LPTIM1）へ

表 67. 相互接続 17

LPTIM トリガ入力信号（TRIGSEL[3:0]）	LPTIM1 トリガソース割り当て
lptim1_ext_trig0	LPTIM1_ETR
lptim1_ext_trig1	rtc_alra_trg
lptim1_ext_trig2	rtc_alrb_trg
lptim1_ext_trig3	RTC_TAMP1
lptim1_ext_trig4	RTC_TAMP2
lptim1_ext_trig5	RTC_TAMP3
lptim1_ext_trig6	comp1_out
lptim1_ext_trig7	comp2_out
lptim1_ext_trig8	comp3_out
lptim1_ext_trig9	comp4_out
lptim1_ext_trig10	comp5_out
lptim1_ext_trig11	comp6_out
lptim1_ext_trig12	comp7_out

RTC アラーム A/B、RTC_TAMP1/2/3 入力検出、COMPx_OUT は、LPTIM カウンタ（LPTIM1）を開始するためのトリガとして使用できます。

トリガ信号

このトリガ機能については、セクション 32.4.6：トリガマルチプレクサ（および次のセクション）で説明されています。

入力選択については、表 311：LPTIM1 外部トリガ接続で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP、STOP 0、STOP 1

11.3.7 タイマ（TIMx）からコンパレータ（COMPx）へ

表 68. 相互接続 15

コンパレータブランキング信号 BLANKSEL[2:0]	コンパレータブランキングソース割り当て						
	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7
1	tim1_oc5	tim1_oc5	tim1_oc5	tim3_oc4	tim2_oc3	tim8_oc5	tim1_oc5
2	tim2_oc3	tim2_oc3	tim3_oc3	tim8_oc5	tim8_oc5	tim2_oc4	tim8_oc5
3	tim3_oc3	tim3_oc3	tim2_oc4	tim15_oc1	tim3_oc3	tim15_oc2	tim3_oc3
4	tim8_oc5	tim8_oc5	tim8_oc5	tim1_oc5	tim1_oc5	tim1_oc5	tim15_oc2
5	tim20_oc5	tim20_oc5	tim20_oc5	tim20_oc5	tim20_oc5	tim20_oc5	tim20_oc5
6	tim15_oc1	tim15_oc1	tim15_oc1	tim15_oc1	tim15_oc1	tim15_oc1	tim15_oc1
7	tim4_oc3	tim4_oc3	tim4_oc3	tim4_oc3	tim4_oc3	tim4_oc3	tim4_oc3

タイマ（TIMx）は、COMPx（x = 1 ~ 7）のブランキング･ウィンドウとして使用できます。

ブランキング機能については、セクション 24.3.6 : COMP 出力のブランキングで説明されています。

ブランキングソースは､コンパレータ制御／ステータスレジスタ（COMP_CxCSR）の BLANKSEL[2:0] ビットに示されています。

トリガ信号

タイマ出力信号 TIMx_Ocx は、COMPx のブランキングソースへの入力です。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.8 内部アナログ･ソースから ADC（ADCx）、コンパレータ（COMPx）、OPAMP（OPAMPx）へ

表 69 に、GPIO または内部接続から、温度センサ（VTS）、内部基準電圧 VREFINT、VBAT/3、opampxint_vout への ADC チャネルのマッピングを示します。

表 69. 相互接続 21

ADC チャネル番号	ADC チャネルソース割り当て				
	ADC1	ADC2	ADC3	ADC4	ADC5
IN0	-	-	-	-	-
IN1	PA0	PA0	PB1/OPAMP3_VOUT	PE14	PA8/OPAMP5_VOUT
IN2	PA1	PA1	PE9	PE15	PA9
IN3	PA2/OPAMP1_VOUT	PA6/OPAMP2_VOUT	PE13	PB12/OPAMP4_VOUT	opamp5_int_vout(1)
IN4	PA3	PA7	PE7	PB14	温度センサ
IN5	PB14	PC4	PB13	PB15	opamp4_int_vout (1)
IN6	PC0	PC0	PE8	PE8	PE8

表 69. 相互接続 21 （続き）

ADC チャネル番号	ADC チャネルソース割り当て				
	ADC1	ADC2	ADC3	ADC4	ADC5
IN7	PC1	PC1	PD10	PD10	PD10
IN8	PC2	PC2	PD11	PD11	PD11
IN9	PC3	PC3	PD12	PD12	PD12
IN10	PF0	PF1	PD13	PD13	PD13
IN11	PB12/OPAMP4_VOUT	PC5	PD14	PD14	PD14
IN12	PB1/OPAMP3_VOUT	PB2	PB0	PD8	PD8
IN13	opamp1_int_vout (1)	PA5	opamp3_int_vout (1)	PD9	PD9
IN14	PB11/OPAMP6_VOUT	PB11/OPAMP6_VOUT	PE10	PE10	PE10
IN15	PB0	PB15	PE11	PE11	PE11
IN16	温度センサ	opamp2_int_vout (1)	PE12	PE12	PE12
IN17	$V_{BAT}/3$	PA4	$V_{BAT}/3$(2) opamp6_int_vout (1)(3)	opamp6_int_vout (1)	$V_{BAT}/3$
IN18	V_{REFINT}	opamp3_int_vout (1)	V_{REFINT}	V_{REFINT}	V_{REFINT}

1. ADC 入力のみに直接接続されている内部 OPAMP 出力（ピン上で外部的には使用不可）
2. カテゴリ 3 デバイスの場合のみ
3. カテゴリ 4 デバイスの場合のみ

DAC 出力は、GPIO 上で使用可能であるか、コンパレータとオペアンプに内部的に接続可能です。

表 70. 相互接続 22

転送先	転送元						
	DAC1		DAC2	DAC3		DAC4	
	CH1	CH2	CH1	CH1	CH2	CH1	CH2
GPIO	PA4	PA5	PA6	-	-	-	-
COMP1	x	-	-	x	-	-	-
COMP2	-	x	-	-	x	-	-
COMP3	x	-	-	x	-	-	-
COMP4	x	-	-	-	x	-	-
COMP5	-	x	-	-	-	x	-
COMP6	-	-	x	-	-		x
COMP7	-	-	x	-	-	x	-
OPAMP1	-	-	-	x	-	-	-
OPAMP2	-	-	-	-	-	-	-

表 70. 相互接続 22 （続き）

転送先	転送元						
	DAC1		DAC2	DAC3		DAC4	
	CH1	CH2	CH1	CH1	CH2	CH1	CH2
OPAMP3	-	-	-	-	x	-	-
OPAMP4	-	-	-	-	-	x	-
OPAMP5	-	-	-	-	-	-	x
OPAMP6	-	-	-	x	-	-	-

表 71. 相互接続 23

コンパレータ入出力信号		コンパレータ入出力信号割り当て										
		GPIO					DAC 出力(1)		VREFINT スケーラ			
COMP1	INP	PA1	PB1	-	-	-	-	-	-	-	-	-
	INM	PA0	PA4	-	-	-	DAC1_CH1 PA4	DAC3_CH1	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PA0	PF4	PA6	PA11	PB8	-	-	-	-	-	-
COMP2	INP	PA3	PA7	-	-	-	-	-	-	-	-	-
	INM	PA2	PA5	-	-	-	DAC1_CH2 PA5	DAC3_CH2	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PA2	PA7	PA12	PB9	-	-	-	-	-	-	-
COMP3	INP	PC1	PA0	-	-	-	-	-	-	-	-	-
	INM	PC0	PF1	-	-	-	DAC1_CH1 PA4	DAC3_CH1	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PC2	PB7	PB15	-	-	-	-	-	-	-	-
COMP4	INP	PB0	PE7	-	-	-	-	-	-	-	-	-
	INM	PB2	PE8	-	-	-	DAC1_CH1 PA4	DAC3_CH2	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PB1	PB6	PB14	-	-	-	-	-	-	-	-
COMP5	INP	PD12	PB13	-	-	-	-	-	-	-	-	-
	INM	PD13	PB10	-	-	-	DAC1_CH2 PA5	DAC4_CH1	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PC7	PA9	-	-	-	-	-	-	-	-	-
COMP6	INP	PD11	PB11	-	-	-	-	-	-	-	-	-
	INM	PD10	PB15	-	-	-	DAC2_CH1 PA6	DAC4_CH2	VREF INT	3/4 VREF INT	1/2 VREF INT	1/4 VREF INT
	OUT	PC6	PA10	-	-	-	-	-	-	-	-	-

表 71. 相互接続 23 （続き）

コンパレータ入出力信号		コンパレータ入出力信号割り当て											
		GPIO					DAC 出力(1)		VREFINT スケーラ				
COMP7	INP	PD14	PB14	-	-	-	-	-	-	-	-	-	-
	INM	PD15	PB12	-	-	-	DAC2_CH1 PA6	DAC4_CH1	VREFINT	3/4 VREFINT	1/2 VREFINT	1/4 VREFINT	
	OUT	PC8	PA8	-	-	-	-	-	-	-	-	-	-

1. 内部接続です。

表 72. 相互接続 24

オペアンプ入出力信号		オペアンプ入出力信号割り当て						
		GPIO				DAC 出力	GPIO の ADC 入力	ADC 内部入力
OPAMP1	VINP	PA1	PA3	PA7	-	DAC3_CH1	-	-
	VINM	PA3	PC5	-	-	-	-	-
	VOUT	PA2		-	-	-	ADC1_IN3	ADC1_IN13
OPAMP2	VINP	PA7	PB14	PB0	PD14	-	-	-
	VINM	PA5	PC5	-	-	-	-	-
	VOUT	PA6		-	-	-	ADC2_IN3	ADC2_IN16
OPAMP3	VINP	PB0	PB13	PA1	-	DAC3_CH2	-	-
	VINM	PB2	PB10	-	-	-	-	-
	VOUT	PB1	-	-	-	-	ADC3_IN1/ ADC1_IN12	ADC2_IN18/ ADC3_IN13
OPAMP4	VINP	PB13	PD11	PB11	-	DAC4_CH1	-	-
	VINM	PB10	PD8	-	-	-	-	-
	VOUT	PB12	-	-	-	-	ADC4_IN3/ ADC1_IN11	ADC5_IN5
OPAMP5	VINP	PB14	PD12	PC3	-	DAC4_CH2	-	-
	VINM	PB15	PA3	-	-	-	-	-
	VOUT	PA8		-	-	-	ADC5_IN1	ADC5_IN3
OPAMP6	VINP	PB12	PD9	PB13	-	DAC3_CH1	-	-
	VINM	PA1	PB1	-	-	-	-	-
	VOUT	PB11	-	-	-	-	ADC12_IN14	ADC4_IN17

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.9 コンパレータ（COMPx）からタイマ（TIMx、HRTIM）へ

コンパレータ（COMPx）出力値は、次のものに接続可能です。

- タイマ（TIMx）入力キャプチャ、TIMx_ETR 信号 または TIMx_OCREFCLR 信号
- hrtim_eevx[4:1] と hrtim_in_fltx[4:1]

コンパレータ（COMPx）出力値は、タイマ（TIMx）のブレーク入力信号を生成することもできます。セクション 30.3.17：双方向ブレーク入力を参照してください。

表 73. 相互接続 2

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て						
	TIM1	**TIM2**	**TIM3**	**TIM4**	**TIM5**	**TIM8**	**TIM20**
timx_etr0	TIM1_ETR	TIM2_ETR	TIM3_ETR	TIM4_ETR	TIM5_ETR	TIM8_ETR	TIM20_ETR
timx_etr1	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out
timx_etr2	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out
timx_etr3	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out
timx_etr4	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out
timx_etr5	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out
timx_etr6	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out
timx_etr7	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out
timx_etr8	adc1_awd1	tim3_etr	tim2_etr	tim3_etr	tim2_etr	adc2_awd1	adc3_awd1
timx_etr9	adc1_awd2	tim4_etr	tim4_etr	tim5_etr	tim3_etr	adc2_awd2	adc3_awd2
timx_etr10	adc1_awd3	tim5_etr	-	-	-	adc2_awd3	adc3_awd3
timx_etr11	adc4_awd1	lse_css_out	adc2_awd1	-	-	adc3_awd1	adc5_awd1
timx_etr12	adc4_awd2	-	adc2_awd2	-	-	adc3_awd2	adc5_awd2
timx_etr13	adc4_awd3	-	adc2_awd3	-	-	adc3_awd3	adc5_awd3

表 74. 相互接続 3

タイマ OCREF クリア信号	タイマ OCREF クリア信号割り当て							
	TIM1	TIM2	TIM3	TIM8	TIM15	TIM16	TIM17	TIM20
timx_ ocref_clr0	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out
timx_ ocref_clr1	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out
timx_ ocref_lr2	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out
timx_ ocref_clr3	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out
timx_ ocref_clr4	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out
timx_ ocref_clr5	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out
timx_ ocref_clr6	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out

表 75. 相互接続 4

タイマ TI1 入力信号	タイマ TI1 信号割り当て									
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM15	TIM16	TIM17	TIM20
timx_ ti1_in0	TIM1 外部 TI1 入力ピン	TIM2 外部 TI1 入力ピン	TIM3 外部 TI1 入力ピン	TIM4 外部 TI1 入力ピン	TIM5 外部 TI1 入力ピン	TIM8 外部 TI1 入力ピン	TIM15 外部 TI1 入力ピン	TIM16 外部 TI1 入力ピン	TIM17 外部 TI1 入力ピン	TIM20 外部 TI1 入力ピン
timx_ ti1_in1	comp1_ out	comp1_ out	comp1_ out	comp1_ out	LSI	comp1_ out	lse_ css_out	comp6_ out	comp5_ out	comp1_ out
timx_ ti1_in2	comp2_ out	comp2_ out	comp2_ out	comp2_ out	lse_ css_out	comp2_ out	comp1_ out	MCO	MCO	comp2_ out
timx_ ti1_in3	comp3_ out	comp3_ out	comp3_ out	comp3_ out	rtc_ ウェイクアップ	comp3_ out	comp2_ out	HSE_Div 32	HSE_Div 32	comp3_ out
timx_ ti1_in4	comp4_ out	comp4_ out	comp4_ out	comp4_ out	comp1_ out	comp4_ out	comp5_ out	rtc_ ウェイクアップ	rtc_ ウェイクアップ	comp4_ out
timx_ ti1_in5	-	comp5_ out	comp5_ out	comp5_ out	comp2_ out	-	comp7_ out	lse_ css_out	lse_ css_out	-
timx_ ti1_in6	-	-	comp6_ out	comp6_ out	comp3_ out	-	-	LSI	LSI	-
timx_ ti1_in7	-	-	comp7_ out	comp7_ out	comp4_ out	-	-	-	-	-
timx_ ti1_in8	-	-	-	-	comp5_ out	-	-	-	-	-
timx_ ti1_in9	-	-	-	-	comp6_ out	-	-	-	-	-
timx_ ti1_in10	-	-	-	-	comp7_ out	-	-	-	-	-

表 76. 相互接続 5

タイマ TI2 入力信号	タイマ TI2 信号割り当て							
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM15	TIM20
timx_ ti2_in0	TIM1 外部 TI2 入力ピン	TIM2 外部 TI2 入力ピン	TIM3 外部 TI2 入力ピン	TIM4 外部 TI2 入力ピン	TIM5 外部 TI2 入力ピン	TIM8 外部 TI2 入力ピン	TIM15 外部 TI2 入力ピン	TIM20 外部 TI2 入力ピン
timx_ ti2_in1	-	comp1_out	comp1_out	comp1_out	comp1_out	-	comp2_out	-
timx_ ti2_in2	-	comp2_out	comp2_out	comp2_out	comp2_out	-	comp3_out	-
timx_ ti2_in3	-	comp3_out	comp3_out	comp3_out	comp3_out	-	comp6_out	-
timx_ ti2_in4	-	comp4_out	comp4_out	comp4_out	comp4_out	-	comp7_out	-
timx_ ti2_in5	-	comp6_out	comp5_out	comp5_out	comp5_out	-	-	-
timx_ ti2_in6	-	-	comp6_out	comp6_out	comp6_out	-	-	-
timx_ ti2_in7	-	-	comp7_out	comp7_out	comp7_out	-	-	-

表 77. 相互接続 6

タイマ TI3 入力信号	タイマ TI3 信号割り当て						
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM20
timx_ ti3_in0	TIM1 外部 TI3 入力ピン	TIM2 外部 TI3 入力ピン	TIM3 外部 TI3 入力ピン	TIM4 外部 TI3 入力ピン	TIM5 外部 TI3 入力ピン	TIM8 外部 TI3 入力ピン	TIM20 外部 TI3 入力ピン
timx_ ti3_in1	-	comp4_out	comp3_out	comp5_out	-	-	-

表 78. 相互接続 7

タイマ TI4 入力信号	タイマ TI4 信号割り当て						
	TIM1	TIM2	TIM3	TIM4	TIM5	TIM8	TIM20
timx_ ti4_in0	TIM1 外部 TI4 入力ピン	TIM2 外部 TI4 入力ピン	TIM3 外部 TI4 入力ピン	TIM4 外部 TI4 入力ピン	TIM5 外部 TI4 入力ピン	TIM8 外部 TI4 入力ピン	TIM20 外部 TI4 入力ピン
timx_ ti4_in1	-	comp1_out	-	comp6_out	-	-	-
timx_ ti4_in2	-	comp2_out	-	-	-	-	-

タイマブレーク入力には、2 つのチャネルがあります。

- アプリケーション障害（入力ピンおよび内蔵コンパレータ）とシステムレベル障害（クロック障害、パリティエラーなど）の両方を収集するブレークチャネル
- アプリケーション障害（入力ピンおよび内蔵コンパレータ）のみを含む ブレーク2 チャネル

表 79、表 80、および表 83 を参照してください。

表 79. 相互接続 8

	タイマブレーク信号割り当て					
	TIM1 ブレーク（tim1_bk）	TIM8 ブレーク（tim8_bk）	TIM15 ブレーク（tim15_bk）	TIM16 ブレーク（tim16_bk）	TIM17 ブレーク（tim17_bk）	TIM20 ブレーク（tim20_bk）
タイマブレーク信号ソース	TIM1_BKIN ピン	TIM8_BKIN ピン	TIM15_BKIN ピン	TIM16_BKIN ピン	TIM17_BKIN ピン	TIM20_BKIN ピン
	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out	comp1_out
	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out	comp2_out
	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out	comp3_out
	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out	comp4_out
	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out	comp5_out
	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out	comp6_out
	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out	comp7_out

表 80. 相互接続 9

	タイマ ブレーク2 信号割り当て		
	TIM1 ブレーク 2（tim1_bk2）	TIM8 ブレーク 2（tim8_bk2）	TIM20 ブレーク 2（tim20_bk2）
タイマ ブレーク2 信号ソース	TIM1_BKIN2 ピン	TIM8_BKIN2 ピン	TIM20_BKIN2 ピン
	comp1_out	comp1_out	comp1_out
	comp2_out	comp2_out	comp2_out
	comp3_out	comp3_out	comp3_out
	comp4_out	comp4_out	comp4_out
	comp5_out	comp5_out	comp5_out
	comp6_out	comp6_out	comp6_out
	comp7_out	comp7_out	comp7_out

表 81. 相互接続 10

HRTIM 外部イベント入力信号	HRTIM 外部イベント信号割り当て			
	EExSRC[1:0] = 0（GPIO ピンから）	EExSRC[1:0] = 1	EExSRC[1:0] = 2	EExSRC[1:0] = 3
hrtim_eev1[4:1]	HRTIM_EEV1	comp2_out	tim1_trgo	adc1_AWD1
hrtim_eev2[4:1]	HRTIM_EEV2	comp4_out	tim2_trgo	adc1_AWD2
hrtim_eev3[4:1]	HRTIM_EEV3	comp6_out	tim3_trgo	adc1_AWD3
hrtim_eev4[4:1]	HRTIM_EEV4	comp1_out	comp5_out	adc2_AWD1
hrtim_eev5[4:1]	HRTIM_EEV5	comp3_out	comp7_out	adc2_AWD2
hrtim_eev6[4:1]	HRTIM_EEV6	comp2_out	comp1_out	adc2_AWD3
hrtim_eev7[4:1]	HRTIM_EEV7	comp4_out	tim7_trgo	adc3_AWD1
hrtim_eev8[4:1]	HRTIM_EEV8	comp6_out	comp3_out	adc4_AWD1
hrtim_eev9[4:1]	HRTIM_EEV9	comp5_out	tim15_trgo	comp4_out
hrtim_eev10[4:1]	HRTIM_EEV10	comp7_out	tim6_trgo	adc5_AWD1

表 82. 相互接続 11

フォールト チャネル	外部入力 FLTxSRC[1:0] = 00	オンチップソース FLTxSRC[1:0] = 01	外部入力 FLTxSRC[1:0] = 10	オンチップソース FLTxSRC[1:0] = 11
hrtim_flt1[4:1]	HRTIM_FLT1	comp2_out	EEV1_muxout	N/A
hrtim_flt2[4:1]	HRTIM_FLT2	comp4_out	EEV2_muxout	N/A
hrtim_flt3[4:1]	HRTIM_FLT3	comp6_out	EEV3_muxout	N/A
hrtim_flt4[4:1]	HRTIM_FLT4	comp1_out	EEV4_muxout	N/A
hrtim_flt5[4:1]	HRTIM_FLT5	comp3_out	EEV5_muxout	N/A
hrtim_flt6[4:1]	HRTIM_FLT6	comp5_out	EEV6_muxout	N/A

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

11.3.10 システムエラーからタイマ（TIMx、HRTIM）および HRTIM へ

TIMx（TIM1/TIM8/TIM20/TIM15/TIM16/TIM17）ブレーク入力と HRTIM システム障害入力は、以下からのマイクロコントローラ内部フォールトイベントを収集します。

- クロックセキュリティシステム（CSS）によって生成されたクロック障害イベント
- PVD 出力
- SRAM パリティエラー信号
- Cortex-M4 LOCKUP （ハードフォルト）出力
- Flash ECC ダブルエラー検出

ブレーク機能の目的は、タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。

この機能については、以下で説明されています。

- セクション 28.3.18：ブレーク機能の使用（TIM1/TIM8/TIM20）
- セクション 30.4.15：ブレーク機能の使用（TIM15/TIM16/TIM17/TIM20）

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

表 83. 相互接続 18

システムエラー信号ソース	システムエラー信号のタイマ信号への割り当て						
	TIM1	TIM8	TIM15	TIM16	TIM17	TIM20	HRTIM
Flash ECC エラー	tim1_bk	tim8_bk	tim15_bk	tim16_bk	tim17_bk	tim20_bk	hrtim_sys_flt
PVD 出力							
SRAM1/ CCM SRAM パリティ							
Cortex®-M4 ロックアップ（ハードフォールト）							
クロックセキュリティシステム（CSS）							

11.3.11 タイマ（TIM16/TIM17）から IRTIM へ

表 84. 相互接続 16

IRTIM 制御信号	IRTIM 制御信号割り当て
変調エンベロープ信号	tim16_oc1
キャリア信号	tim17_oc1

汎用タイマ（TIM16/TIM17）の出力チャネル TIMx_OC1 は、赤外線信号出力の波形を生成するために使用されます。

この機能については、セクション 33：赤外線インタフェース（IRTIM）で説明されています。

アクティブな電力モード

RUN、SLEEP、低電力 RUN、低電力 SLEEP

12 ダイレクト･メモリ･アクセス（DMA）コントローラ

12.1 概要

ダイレクトメモリアクセス（DMA）コントローラは、バスマスタとシステムペリフェラルで構成されます。

DMA を使用して、CPU を介さずにメモリマップペリフェラルやメモリの間でプログラム可能なデータ転送を実行します。

DMA コントローラは、シングル AHB マスタバスアーキテクチャを備えています。

DMA には 2 つのインスタンス、DMA1 と DMA2 があります（対応チャネル数については、表 85 : DMA1 および DMA2 の実装を参照）。

それぞれのチャネルは、1 つ以上のペリフェラルからのメモリアセスリクエストを管理する役割を担っています。各 DMA は DMA リクエスト間の優先順位を操作するためのアービタを内蔵しています。

12.2 DMA の主な機能

- シングル AHB マスタ
- ペリフェラルからメモリ、メモリからペリフェラル、メモリ間、ペリフェラル間のデータ転送
- 転送元および転送先として、Flashメモリ、SRAM、および AHB/APB ペリフェラルなどのオンチップ・メモリマップ・デバイスにアクセス
- 個別に設定可能なすべての DMA チャネル：
  - 各チャネルは、ペリフェラルからの DMA リクエスト信号、またはメモリ間転送におけるソフトウェアトリガに関連付けられます。この設定はソフトウェアで行われます。
  - リクエスト間の優先順位はソフトウェアでプログラム可能です（チャネル別に最高、高、中、低の 4 レベル）。またはレベルが等しい場合はハードウェアによって決まります（たとえばチャネル 1 へのリクエストはチャネル 2 へのリクエストよりも優先）。
  - 転送元および転送先の転送サイズ（バイト、ハーフワード、ワード）は個別に設定され、パッキング／アンパッキングをエミュレートします。転送元および転送先のアドレスはデータサイズに合わせる必要があります。
  - サーキュラバッファ管理によるペリフェラルとメモリの間の転送をサポートします。
  - プログラム可能な転送データ数：0～2^{16} - 1
- チャネルレベルですべての割込みリクエストの OR をとって、DMA1 コントローラに対する単一割込みリクエストを生成。割込みリクエストは、転送完了、1/2 転送、転送エラーの 3 種類の DMA イベントのいずれかによって生成されます。
- DMA2 コントローラに対して、チャネル別に割込みリクエストを生成。各割込みリクエストは 3 種類の DMA イベント、転送完了、1/2 転送、または転送エラーのいずれかによって生成されます。

12.3 DMA の実装

12.3.1 DMA1 および DMA2

DMA1および DMA2は、次の表に示すハードウェア構成パラメータによって実装されています。

表 85. DMA1 および DMA2 の実装

機能		DMA1	DMA2
チャネル数	カテゴリ 2 デバイス(1) カテゴリ 3 デバイス (1) カテゴリ 4 デバイス (1)	6 8 8	6 8 8

1. 表 1：STM32G4 シリーズメモリ容量を参照してください。

12.3.2 DMA リクエストマッピング

DMA コントローラは、AHB/APB ペリフェラルから DMAMUX ペリフェラルを介して送信される DMA リクエストに接続されます。

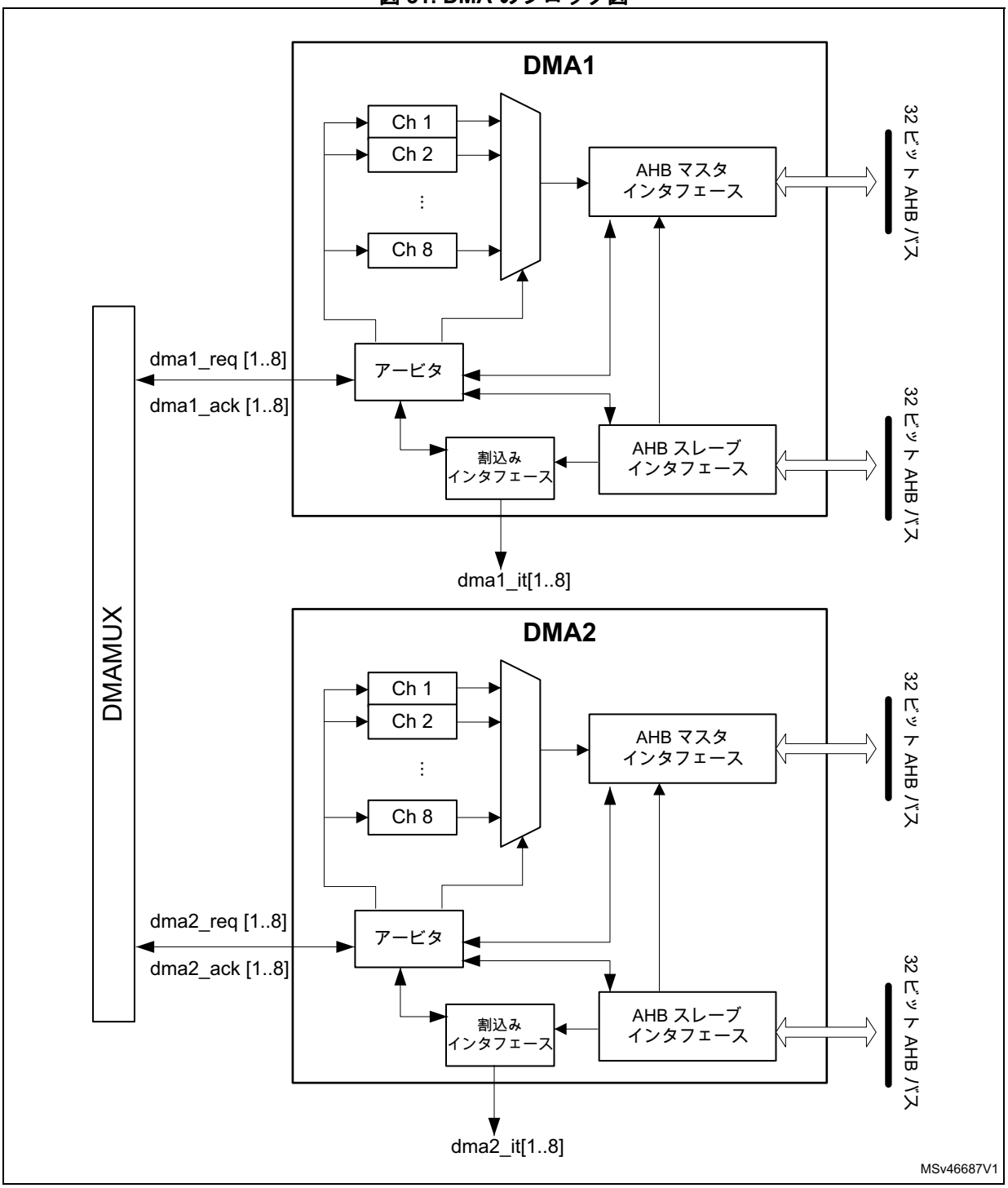
さまざまなリクエストのマッピングについては、DMAMUX のセクションを参照してください。

12.4 DMA の機能詳細

12.4.1 DMA のブロック図

以下に DMA のブロック図を示します。

図 31. DMA のブロック図



注：　機能の実装については、図 31：DMA のブロック図を参照してください。

DMA コントローラは、AHB システムバスを他のシステムマスタと共有することでダイレクトメモリ転送を行います。バスマトリックスはラウンドロビンスケジューリングを実装します。DMA リクエストは、CPU と DMA のターゲット転送先が同じである場合に（メモリまたはペリフェラル）、多少のバスサイクルの間、CPU のシステムバスへのアクセスを停止することがあります。

DMA コントローラは、AHB スレーブインタフェースによるその設定に応じて、DMA チャネルと関連の受信リクエストの間のアービトレーションを行います。また、シングル AHB ポートマスタによる DMA データ転送をスケジュールします。

また、チャネルごとの割込みコントローラへの割込みを生成します。

12.4.2 DMA ピンおよび内部信号

表 86. DMA 内部入力／出力信号

信号名	信号タイプ	説明
dma_req[x]	入力	DMA チャネル x リクエスト
dma_ack[x]	出力	DMA チャネル x 確認応答
dma_it[x]	出力	DMA チャネル x 割込み

12.4.3 DMA 転送

ソフトウェアは、AHB バス転送シーケンスから成るブロック転送を行うために DMA コントローラをチャネルレベルで設定します。

DMA ブロック転送は、ペリフェラルからリクエストするか、メモリ間転送の場合はソフトウェアでトリガすることができます。

イベントの後、シングル DMA 転送の次の手順が行われます。

1. ペリフェラルは、DMA コントローラにシングル DMA リクエスト信号を送信します。
2. DMA コントローラは、このペリフェラルのリクエストに関連付けられたチャネルの優先順位に応じて、リクエストを処理します。
3. DMA コントローラは、ペリフェラルのリクエストを許可するとすぐに、ペリフェラルに確認応答信号を送信します。
4. ペリフェラルは、DMA コントローラからの確認応答信号を受け取るとすぐにそのリクエストを解除します。
5. ペリフェラルによってリクエストが無効にされると、DMA コントローラは確認応答信号を解除します。

ペリフェラルは、さらにシングルリクエストを送信し、別のシングル DMA 転送を開始することができます。

ペリフェラルが転送元または転送先である場合は、リクエスト／確認応答プロトコルを使用します。たとえば、メモリからペリフェラル転送の場合、ペリフェラルは DMA コントローラにシングルリクエスト信号を送信することによって転送を開始します。次に、DMA コントローラはメモリ内の単一データを読み出し、このデータをペリフェラルに書き込みます。

任意のチャネル x の DMA ブロック転送では、次の転送が繰り返されます。

- 2つの単一データ AHB 転送をカプセル化する、DMA AHB バスマスタ経由のシングル DMA 転送
  - ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ上の場所からの単一データ読出し（バイト／ハーフワード／ワード）。

最初のシングル転送に使用される開始アドレスは、ペリフェラル／メモリのベースアドレスで、DMA_CPARx または DMA_CMARx レジスタでプログラムされています。

– ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ上の場所からの単一データ書込み（バイト／ハーフワード／ワード）。
最初の転送に使用される開始アドレスは、ペリフェラル／メモリのベースアドレスで、DMA_CPARx または DMA_CMARx レジスタでプログラムされています。

- プログラムされた DMA_CNDTRx レジスタのポストデクリメント
このレジスタは、残りの転送データ項目数（AHB の読出し転送に続く書込み転送の数）を含みます。

DMA_CNDTRx が null になるまで、このシーケンスを繰り返します。

注： **AHB マスタバス転送元／転送先のアドレスは、その転送元／転送先に転送されるプログラムされた単一データのサイズに合わせてください。**

12.4.4 DMA アービトレーション

DMA アービタは、異なるチャネル間の優先順位を管理します。

アービタがアクティブなチャネル x を許可すると（ハードウェアによるリクエスト／ソフトウェアによるトリガ）、シングル DMA 転送（AHB の単一データ読出し転送に続く書込み転送など）が発行されます。次に、アービタはそれらのアクティブなチャネルの中から再び優先順位が一番高いものを選びます。

優先順位は 2 段階で管理されます。

- ソフトウェア：各チャネルの優先順位は DMA_CCRx レジスタで次の 4 段階のいずれかに設定します。
  – 最高
  – 高
  – 中
  – 低
- ハードウェア : 2 つのリクエストのソフトウェア優先順位レベルが同じ場合、小さい番号のチャネルが優先されます。たとえば、チャネル 2 はチャネル 4 よりも優先されます。

チャネル x がメモリ間モードのブロック転送用にプログラムされている場合、このチャネル x のシングル DMA 転送間の再アービトレーションが検討されます。同じタイミングで別のアクティブなチャネルがリクエストされた場合、DMA アービタは自動的にもう一方の最優先リクエストチャネルと交互に許可を与えますが、優先順位はメモリ間転送用のチャネルより低いかもしれません。

12.4.5 DMA チャネル

各チャネルは、ある固定アドレスにあるペリフェラルレジスタとメモリアドレスの間の DMA 転送を処理することができます。転送データ項目数はプログラム可能です。転送データ項目数を格納しているレジスタは、転送終了ごとにデクリメントされます。

DMA チャネルはブロック転送レベルでプログラムされます。

プログラム可能なデータサイズ

ペリフェラルおよびメモリに転送される単一データのサイズ（バイト／ハーフワード／ワード）は、それぞれ DMA_CCRx レジスタの PSIZE[1:0] および MSIZE[1:0] フィールドでプログラムすることができます。

ポインタのインクリメント

ペリフェラルおよびメモリのポインタは、DMA_CCRx レジスタの PINC および MINC ビットに応じて、各転送後に自動的にインクリメントすることができます。

インクリメントモード が有効な場合（PINC または MINC = 1)、次の転送のアドレスは、PSIZE[1:0] または MSIZE[1:0] に定義されているデータサイズに応じて、前のアドレスに 1、2、または 4 をインクリメントしたアドレスになります。最初の転送アドレスは、DMA_CPARx または DMA_CMARx レジスタでプログラムされたアドレスとなります。転送中は、これらのレジスタは最初にプログラムされた値を保持します。現在の転送アドレス（現在の内部ペリフェラル／メモリアドレスレジスタ内）に対するソフトウェアによるアクセスはできません。

チャネル x が **ノンサーキュラモード**に設定されている場合、データ転送終了後（転送すべきデータ項目数がゼロに達したとき）、DMA リクエストは処理されません。新たなデータ項目数を DMA_CNDTRx レジスタに再ロードするには、DMA チャネルを無効にする必要があります。

注： **チャネル x が無効になると、DMA レジスタはリセットされません。DMA チャネルレジスタ（DMA_CCRx、DMA_CPARx、DMA_CMARx）は、チャネル設定フェーズでプログラムされた初期値のままです。**

サーキュラモードでは、データ転送終了後、DMA_CNDTRx レジスタには最初にプログラムした値が自動的に再ロードされます。現在の内部アドレスレジスタには、DMA_CPARx および DMA_CMARx レジスタからのベースアドレス値が再ロードされます。

チャネル設定手順

DMA チャネル x の設定には、次の手順が必要です。

1. DMA_CPARx レジスタに、ペリフェラルレジスタアドレスをセットします。
   データは、ペリフェラルイベント後、またはチャネルがメモリ間モードで有効化された後に、このアドレスからメモリに、またはメモリからこのアドレスに移動されます。
2. DMA_CMARx レジスタにメモリアドレスをセットします。
   データは、ペリフェラルイベント後、またはチャネルがメモリ間モードで有効化された後に、メモリへ書き込まれたり、メモリから読み出されたりします。
3. 転送すべきデータの総数を DMA_CNDTRx レジスタに設定します。
   毎回のデータ転送後に、この値はデクリメントされます。
4. 以下に挙げるパラメータを DMA_CCRx レジスタで設定します。
   - チャネルの優先順位
   - データ転送方向
   - サーキュラモード
   - ペリフェラルとメモリのインクリメントモード
   - ペリフェラルとメモリのデータサイズ
   - 1/2転送時、転送完了時、転送エラー時の割込み有効化
5. DMA_CCRx レジスタの EN ビットをセットすることでチャネルを有効にします。

チャネルは、有効になるとすぐに、そのチャネルに接続されているペリフェラルからの DMAリクエストを処理したり、メモリ間ブロック転送を開始することができます。

注： **チャネル設定手順の最後の 2 ステップを DMA_CCRx レジスタへのシングルアクセスの中に含めて、チャネルを設定および有効化しても構いません。**

チャネルの状態とチャネルの無効化

アクティブ状態のチャネル x は有効チャネルです（DMA_CCRx.EN の読出し = 1）。アクティブチャネル x は、ソフトウェアによって有効化され（DMA_CCRx.EN を 1 にセット）、その後に転送エラーが発生していない（DMA_ISR.TEIFx = 0）必要があります。転送エラーがあると、そのチャネルはハードウェアによって自動的に無効化されます（DMA_CCRx.EN = 0）。

以下の 3 種類のユースケースが発生する可能性があります。

- チャネルのサスペンドと再開

  このユースケースは、次の 2 つのアクションに相当します。

  – アクティブチャネルがソフトウェアによって無効化されます（DMA_CCRx.EN = 1 であるのに DMA_CCRx.EN = 0 を書込み）。

  – 他のチャネル･レジスタ（DMA_CNDTRx、DMA_CPARx、DMA_CMARx など）を再設定することなく、ソフトウェアがそのチャネルを再度有効化します（DMA_CCRx.EN を 1 にセット）。

  このケースには DMA ハードウェアが対応しておらず、残りのデータ転送が正常に実行されるかどうかは保証されません。

- チャネルの停止とアボート

  アプリケーションがもはやそのチャネルを必要としていない場合、このアクティブチャネルはソフトウェアによって無効化できます。チャネルの停止とアボートは行われますが、アボートされた転送元と転送先のバッファ／レジスタに対して、DMA_CNDTRx レジスタの内容は、残りのデータ転送を正しく反映していない可能性があります。

- チャネルのアボートと再開

  このユースケースは、アクティブチャネルを無効化した後にそのチャネルを再設定して再び有効化するというソフトウェアシーケンスに相当します。

  次の条件が満足された場合に、ハードウェアによる対応が行われます。

  – ソフトウェアがチャネルを無効化するときに、そのマスタポートに対して DMA データ転送が同時に発生していないことをアプリケーションが保証します。たとえば、アプリケーションは、DMA モードにあるペリフェラルをまず無効化して、このペリフェラルからの保留中のハードウェア DMAリクエストが存在しないことを保証することができます。

  – ソフトウェアは、以下のようにして、同一の DMA_CCRx レジスタに対して、独立した書込みアクセスを行う必要があります。最初に、チャネルを無効化します。次に、設定の変更が必要であれば、DMA_CCRx を含む次のブロック転送に対してチャネルを再設定します。DMA_CCRx.EN=1 のときには、読出し専用の DMA_CCRx レジスタフィールドが存在します。最後に、チャネルを再び有効化します。

チャネル転送エラーが発生すると、DMA_CCRx レジスタの EN ビットがハードウェアによってクリアされます。DMA_ISR レジスタの TEIFx ビットがセットされるまでは、チャネル x を再有効化するために、この EN ビットをソフトウェアによって再セットすることはできません。

サーキュラモード（メモリからペリフェラル／ペリフェラルからメモリへの転送）

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー（ADC スキャンモードなど）を処理できます。この機能は、DMA_CCRx レジスタの CIRC ビットを使用して有効にできます。

注： **サーキュラモードはメモリ間転送では使用しないでください。サーキュラモードでのチャネル有効化（CIRC = 1）の前に、ソフトウェアによって DMA_CCRx レジスタの MEM2MEM ビットをクリアしてください。サーキュラモードが有効になると、転送データ数にはチャネル設定段階でプログラミングされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。**

サーキュラ転送を停止するには、DMA チャネルを無効化する前に、ソフトウェアによってペリフェラルによる DMA リクエストの生成を止めさせる必要があります（ADC スキャンモードの中止など）。転送を開始／有効化する前、およびサーキュラ転送の停止後、ソフトウェアによって DMA_CNDTRx の値を明示的にプログラムしてください。

メモリ間モード

DMA チャネルは、ペリフェラルからのリクエストによってトリガされなくても動作できます。このモードはメモリ間モードと呼ばれ、ソフトウェアによって開始されます。

DMA_CCRx レジスタの MEM2MEM ビットがセットされると、チャネルは、有効な場合、転送を開始します。DMA_CNDTRx レジスタがゼロに達すると、転送は停止します。

注： **メモリ間モードはサーキュラモードでは使用しないでください。メモリ間モードでのチャネル有効化（MEM2MEM = 1）の前に、ソフトウェアによって DMA_CCRx レジスタの CIRC ビットをクリアしてください。**

ペリフェラル間モード

DMA チャネルは、以下の場合にペリフェラル間モードで動作可能です。

- DMA チャネルをトリガするために、ペリフェラルからのハードウェアリクエストが選択されている場合。

  このペリフェラルは DMA イニシエータであり、このペリフェラルと別のメモリマップペリフェラル（DMA モードに設定されていないもの）に属するレジスタの間のデータ転送のペースを調整します。
- ペリフェラルリクエストが選択されておらず、DMA チャネルに接続されていない場合

  ソフトウェアは、DMA_CCRx レジスタの MEM2MEM ビットをセットすることにより、レジスタ間の転送を設定します。

転送方向のプログラミング、転送元／転送先の割当て

DMA_CCRx レジスタの DIR ビットには転送方向が設定されます。これにより、転送元／転送先の種類（ペリフェラルやメモリ）に関係なく、その値から転送元と転送先が特定されます。

- **DIR = 1** は、一般にメモリからペリフェラルへの転送を定義します。より一般的には、DIR = 1 の場合
  - **転送元** 属性は、DMA_MARx レジスタ、DMA_CCRx レジスタの MSIZE[1:0] フィールドと MINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「メモリ」のレジスタ、フィールド、およびビットは、ペリフェラル間転送における転送元ペリフェラルを定義するために使用します。
  - **転送先** 属性は、DMA_PARx レジスタ、DMA_CCRx レジスタの PSIZE[1:0] フィールドと PINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「ペリフェラル」のレジスタ、フィールド、およびビットは、メモリ間転送における転送先メモリを定義するために使用します。
- **DIR = 0** は、一般にメペリフェラルからメモリへの転送を定義します。より一般的には、DIR = 0 の場合
  - **転送元** 属性は、DMA_PARx レジスタ、DMA_CCRx レジスタの PSIZE[1:0] フィールドと PINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「ペリフェラル」のレジスタ、フィールド、およびビットは、メモリ間転送における転送元メモリを定義するために使用します。
  - **転送先** 属性は、DMA_MARx レジスタ、DMA_CCRx レジスタの MSIZE[1:0] フィールドと MINC ビットで定義します。
    それらの一般的な名称に関係なく、このような「メモリ」のレジスタ、フィールド、およびビットは、ペリフェラル間転送における転送先ペリフェラルを定義するために使用します。

12.4.6 DMA データの幅、アライメント、およびエンディアン

PSIZE[1:0] と MSIZE[1:0] が等しくない場合、DMA コントローラは、次の表のように一部のデータのアライメントを行います。

表 87. プログラム可能なデータ幅およびエンディアンの動作（PINC = MINC = 1 の場合）

転送元ポートの幅（DIR = 1 ならば MSIZE、そうでなければ PSIZE）	転送先ポートの幅（DIR = 1 ならば PSIZE、そうでなければ MSIZE）	転送するデータ項目の数（NDT）	転送元の内容：アドレス／データ（DIR = 1 の場合は DMA_CMARx、それ以外の場合は DMA_CPARx）	DMA 転送	転送先の内容：アドレス／データ（DIR = 1 の場合は DMA_CPARx、それ以外の場合は DMA_CMARx）
8	8	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に B1[7:0] を 0x1 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に B2[7:0] を 0x2 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に B3[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3
8	16	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 00B0[15:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 00B1[15:0] を 0x2 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 00B2[15:0] を 0x4 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 00B3[15:0] を 0x6 に書き込みます。	@0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3
8	32	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1：B0[7:0] を 0x0 で読み出し、次に 000000B0[31:0] を 0x0 に書き込みます。 2：B1[7:0] を 0x1 で読み出し、次に 000000B1[31:0] を 0x4 に書き込みます。 3：B2[7:0] を 0x2 で読み出し、次に 000000B2[31:0] を 0x8 に書き込みます。 4：B3[7:0] を 0x3 で読み出し、次に 000000B3[31:0] を 0xC に書き込みます。	@0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3
16	8	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B2[7:0] を 0x1 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B4[7:0] を 0x2 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B6[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6
16	16	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に B3B2[15:0] を 0x2 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x4 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に B7B6[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6
16	32	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1：B1B0[15:0] を 0x0 で読み出し、次に 0000B1B0[31:0] を 0x0 に書き込みます。 2：B3B2[15:0] を 0x2 で読み出し、次に 0000B3B2[31:0] を 0x4 に書き込みます。 3：B5B4[15:0] を 0x4 で読み出し、次に 0000B5B4[31:0] を 0x8 に書き込みます。 4：B7B6[15:0] を 0x6 で読み出し、次に 0000B7B6[31:0] を 0xC に書き込みます。	@0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6
32	8	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B4[7:0] を 0x1 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B8[7:0] を 0x2 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BC[7:0] を 0x3 に書き込みます。	@0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC
32	16	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x2 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し、次に B9B8[15:0] を 0x4 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し、次に BDBC[15:0] を 0x6 に書き込みます。	@0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC
32	32	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1：B3B2B1B0[31:0] を 0x0 で読み出し､次に B3B2B1B0[31:0] を 0x0 に書き込みます。 2：B7B6B5B4[31:0] を 0x4 で読み出し､次に B7B6B5B4[31:0] を 0x4 に書き込みます。 3：BBBAB9B8[31:0] を 0x8 で読み出し､次に BBBAB9B8[31:0] を 0x8 に書き込みます。 4：BFBEBDBC[31:0] を 0xC で読み出し､次に BFBEBDBC[31:0] を 0xC に書き込みます。	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC

バイト／ハーフワード書込み転送をサポートしていない AHB ペリフェラルのアドレス指定

DMA コントローラが AHB のバイトまたはハーフワード書込み転送を開始すると、データは AHB マスタの 32ビットデータバス（HWDATA[31:0]）の未使用レーンに複製されます。

AHB スレーブペリフェラルがバイト／ハーフワード書込み転送をサポートしておらず、エラーが 1件も出ていない場合は、DMA コントローラは以下の 2つの例に示されるように 32 HWDATA ビットを書き込みます。

- ハーフワード 0xABCD を書き込むには、DMA コントローラは HWDATA バスをデータサイズのハーフワードで 0xABCDABCD に設定します（AHB マスタバスでは HSIZE = ハーフワード）。
- バイト 0xAB を書き込むには、DMA コントローラは HWDATA バスをデータサイズのバイトで 0xABABABAB に設定します（AHB マスタバスでは HSIZE = バイト）。

AHB/APB ブリッジが AHB 32 ビットスレーブペリフェラルである（HSIZE の値は考慮せず）と仮定した場合、以下のようなやり方で、AHB バイトまたはハーフワード転送を 32 ビット APB 転送に変換します。

- データ「0xB0」を アドレス 0x0、0x1、0x2、または 0x3 のいずれかに書き込む AHB バイト書込み転送は、データ「0xB0B0B0B0」をアドレス 0x0 に書き込む APB ワード書込み転送に変換されます。
- データ「0xB1B0」を アドレス 0x0 または 0x2 に書き込む AHB ハーフワード書込み転送は、データ「0xB1B0B1B0」をアドレス 0x0 に書き込む APB ワード書込み転送に変換されます。

12.4.7 DMA エラー管理

DMA 転送エラーは、予約済みアドレス空間に対する読出しや書込み時に生成されます。DMA 読出しまたは書込みアクセス中に DMA 転送エラーが発生した場合、障害のあるチャネル x は、対応する DMA_CCRx レジスタの EN ビットをハードウェアによってクリアすることにより、自動的に無効化されます。

DMA_ISR レジスタの TEIFx ビットがセットされます。次に、DMA_CCRx レジスタの TEIE ビットがセットされている場合、割込みが生成されます。

DMA_ISR レジスタの TEIFx ビットがクリアされるまでは（DMA_IFCR レジスタの CTEIFx ビットをセットすることによって）、DMA_CCRx レジスタの EN ビットをソフトウェアによって再セット（チャネル x の再有効化）することはできません。

ペリフェラルを含むチャネルを介して転送エラーがソフトウェアに通知されると、ペンディング状態のまたはこの先の DMA リクエストを無効にするために、ソフトウェアはまず DMA モードでこのペリフェラルを停止します。ソフトウェアは、通常、DMA モードで DMA とペリフェラルの両方を次の転送用に再設定することができます。

12.5 DMA 割込み

割込みは、DMA チャネル x ごとの 1/2 転送、転送完了、転送エラーに対して生成されます。高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 88. DMA 割込みリクエスト

割込みリクエスト	割込みイベント	イベントフラグ	割込み有効化ビット
チャネル x 割込み	チャネル x の 1/2 転送	HTIFx	HTIEx
	チャネル x の転送完了	TCIFx	TCIEx
	チャネル x の転送エラー	TEIFx	TEIEx
	チャネル x の 1/2 転送、転送完了、または転送エラー	GIFx	-

12.6 DMA レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

DMA レジスタには、ワード（32ビット）単位でアクセスする必要があります。

注 : 機能の実装については、図 31 : DMA のブロック図を参照してください。

12.6.1 DMA 割込みステータスレジスタ（DMA_ISR）

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

ステータス･ビットは、ソフトウェアが DMA_IFCR レジスタの対応するクリア･ビットまたは対応するグローバル･クリア･ビット CGIFx をセットしたときに、ハードウェアによってすべてクリアされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEIF8	HTIF8	TCIF8	GIF8	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31 **TEIF8** : チャネル 8 の転送エラー（TE）フラグ
0 : TE イベントなし
1 : TE イベント発生

ビット 30 **HTIF8** : チャネル 8 の 1/2 転送（HT）フラグ
0 : HT イベントなし
1 : HT イベント発生

ビット 29 **TCIF8** : チャネル 8 の転送完了（TC）フラグ
0 : TC イベントなし
1 : TC イベント発生

ビット 28 **GIF8** : チャネル 8 のグローバル割込みフラグ
0 : TE、HT、または TC イベントなし
1 : TE、HT、または TC イベント発生

ビット 27 **TEIF7**：チャネル 7 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 26 **HTIF7**：チャネル 7 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 25 **TCIF7**：チャネル 7 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 24 **GIF7**：チャネル 7 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 23 **TEIF6**：チャネル 6 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 22 **HTIF6**：チャネル 6 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 21 **TCIF6**：チャネル 6 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 20 **GIF6**：チャネル 6 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 19 **TEIF5**：チャネル 5 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 18 **HTIF5**：チャネル 5 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 17 **TCIF5**：チャネル 5 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 16 **GIF5**：チャネル 5 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 15 **TEIF4**：チャネル 4 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 14 **HTIF4**：チャネル 4 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 13 **TCIF4**：チャネル 4 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 12 **GIF4**：チャネル 4 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 11 **TEIF3**：チャネル 3 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 10 **HTIF3**：チャネル 3 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 9 **TCIF3**：チャネル 3 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 8 **GIF3**：チャネル 3 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 7 **TEIF2**：チャネル 2 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 6 **HTIF2**：チャネル 2 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 5 **TCIF2**：チャネル 2 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 4 **GIF2**：チャネル 2 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

ビット 3 **TEIF1**：チャネル 1 の転送エラー（TE）フラグ
0：TE イベントなし
1：TE イベント発生

ビット 2 **HTIF1**：チャネル 1 の 1/2 転送（HT）フラグ
0：HT イベントなし
1：HT イベント発生

ビット 1 **TCIF1**：チャネル 1 の転送完了（TC）フラグ
0：TC イベントなし
1：TC イベント発生

ビット 0 **GIF1**：チャネル 1 のグローバル割込みフラグ
0：TE、HT、または TC イベントなし
1：TE、HT、または TC イベント発生

12.6.2 DMA 割込みフラグクリアレジスタ（DMA_IFCR）

アドレスオフセット：0x04

リセット値：0x0000 0000

この DMA_IFCR レジスタでチャネル x のグローバル･クリア･ビット CGIFx をセットすると、DMA ハードウェアは、対応する GIFx ビットおよび DMA_ISR レジスタの TEIFx、HTIFx、TCIFx の個々のフラグをクリアします。

この DMA_IFCR レジスタで CTEIFx、CHTIFx、CTCIFx の個々のクリアビッをセットすると、DMA ハードウェアは、対応する個々のフラグ、および DMA_ISR レジスタのグローバルフラグ GIFx をクリアします。ただし、その他の 2 つのフラグがいずれもセットされていないことを条件とします。

フラグクリアビットに 0 を書き込んでも影響はありません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTEIF8	CHTIF8	CTCIF8	CGIF8	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **CTEIF8**：チャネル 8 の転送エラーフラグのクリア

ビット 30 **CHTIF8**：チャネル 8 の 1/2 転送フラグのクリア

ビット 29 **CTCIF8**：チャネル 8 の転送完了フラグのクリア

ビット 28 **CGIF8**：チャネル 8 のグローバル割込みフラグのクリア

ビット 27 **CTEIF7**：チャネル 7 の転送エラーフラグのクリア

ビット 26 **CHTIF7**：チャネル 7 の 1/2 転送フラグのクリア

ビット 25 **CTCIF7**：チャネル 7 の転送完了フラグのクリア

ビット 24 **CGIF7**：チャネル 7 のグローバル割込みフラグのクリア

ビット 23 **CTEIF6**：チャネル 6 の転送エラーフラグのクリア

ビット 22 **CHTIF6**：チャネル 6 の 1/2 転送フラグのクリア

ビット 21 **CTCIF6**：チャネル 6 の転送完了フラグのクリア

ビット 20 **CGIF6**：チャネル 6 のグローバル割込みフラグのクリア

ビット 19 **CTEIF5**：チャネル 5 の転送エラーフラグのクリア

ビット 18 **CHTIF5**：チャネル 5 の 1/2 転送フラグのクリア

ビット 17 **CTCIF5**：チャネル 5 の転送完了フラグのクリア

ビット 16 **CGIF5**：チャネル 5 のグローバル割込みフラグのクリア

ビット 15 **CTEIF4**：チャネル 4 の転送エラーフラグのクリア

ビット 14 **CHTIF4**：チャネル 4 の 1/2 転送フラグのクリア

ビット 13 **CTCIF4**：チャネル 4 の転送完了フラグのクリア

ビット 12 **CGIF4**：チャネル 4 のグローバル割込みフラグのクリア

ビット 11 **CTEIF3**：チャネル 3 の転送エラーフラグのクリア

ビット 10 **CHTIF3**：チャネル 3 の 1/2 転送フラグのクリア

ビット 9 **CTCIF3**：チャネル 3 の転送完了フラグのクリア

ビット 8 **CGIF3**：チャネル 3 のグローバル割込みフラグのクリア

ビット 7 **CTEIF2**：チャネル 2 の転送エラーフラグのクリア

ビット 6 **CHTIF2**：チャネル 2 の 1/2 転送フラグのクリア

ビット 5 **CTCIF2**：チャネル 2 の転送完了フラグのクリア

ビット 4 **CGIF2**：チャネル 2 のグローバル割込みフラグのクリア

ビット 3 **CTEIF1**：チャネル 1 の転送エラーフラグのクリア

ビット 2 **CHTIF1**：チャネル 1 の 1/2 転送フラグのクリア

ビット 1 **CTCIF1**：チャネル 1 の転送完了フラグのクリア

ビット 0 **CGIF1**：チャネル 1 のグローバル割込みフラグのクリア

12.6.3 DMA チャネル x 設定レジスタ（DMA_CCRx）

アドレスオフセット：0x08 + 0x14 * (x - 1)、（x = 1〜8）

リセット値：0x0000 0000

レジスタ･フィールド／ビット MEM2MEM、PL[1:0]、MSIZE[1:0]、PSIZE[1:0]、MINC、PINC、および DIR は EN = 1 のとき、読出し専用です。

MEM2MEM ビットと CIRC ビットの状態が両方とも同時にハイになってはいけません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **MEM2MEM**：メモリ間モード

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 13:12 **PL[1:0]**：優先順位レベル

00：低

01：中

10：高

11：最高

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 11:10 **MSIZE[1:0]**：メモリサイズ

このビットは、指定されたメモリへの各 DMA 転送のデータサイズを定義します。
メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送元を、DIR = 0 の場合はメモリ転送先を示します。
ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送元を、DIR = 0 の場合はペリフェラル転送先を示します。
00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 9:8 **PSIZE[1:0]**：ペリフェラルサイズ

このビットは、指定されたペリフェラルへの各 DMA 転送のデータサイズを定義します。
メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送先を、DIR = 0 の場合はメモリ転送元を示します。
ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送先を、DIR = 0 の場合はペリフェラル転送元を示します。
00：8 ビット
01：16 ビット
10：32 ビット
11：予約済み

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 7 **MINC**：メモリインクリメントモード

このビットは、指定されたメモリへの各 DMA 転送のインクリメントモードを定義します。
メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送元を、DIR = 0 の場合はメモリ転送先を示します。
ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送元を、DIR = 0 の場合はペリフェラル転送先を示します。
0：無効
1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 6 **PINC**：ペリフェラルインクリメントモード

このビットは、指定されたペリフェラルへの各 DMA 転送のインクリメントモードを定義します。
メモリ間モードでは、このフィールドは DIR = 1 の場合はメモリ転送先を、DIR = 0 の場合はメモリ転送元を示します。
ペリフェラル間モードでは、このフィールドは DIR = 1 の場合はペリフェラル転送先を、DIR = 0 の場合はペリフェラル転送元を示します。
0：無効
1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 5 **CIRC**：サーキュラモード

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 4 **DIR**：データ転送方向

このビットは、必ず"メモリからペリフェラル"モードおよび"ペリフェラルからメモリ"モードでセットしてください。

0：ペリフェラルからの読出し

– 転送元属性は、PSIZE と PINC、および DMA_CPARx レジスタで定義します。この条件は、メモリ間モードでも有効です。

– 転送先属性は、MSIZE と MINC、および DMA_CMARx レジスタで定義します。この条件は、ペリフェラル間モードでも有効です。

1：メモリからの読出し

– 転送先属性は、PSIZE と PINC、および DMA_CPARx レジスタで定義します。この条件は、メモリ間モードでも有効です。

– 転送元属性は、MSIZE と MINC、および DMA_CMARx レジスタで定義します。この条件は、ペリフェラル間モードでも有効です。

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

ビット 3 **TEIE**：転送エラー割込みイネーブル

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 2 **HTIE**：1/2 転送割込み有効化

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 1 **TCIE**：転送完了割込み有効化

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

ビット 0 **EN**：チャネル有効化

チャネル転送エラーが発生すると、このビットはハードウェアによってクリアされます。DMA_ISR レジスタの TEIFx ビットがクリアされるまでは（DMA_IFCR レジスタの CTEIFx ビットをセットすることによって）、このビットをソフトウェアによって再セット（チャネル x の再起動）することはできません。

0：無効

1：有効

注： このビットは、ソフトウェアによってセット／クリアされます。

12.6.4 DMA チャネル x 転送データ数レジスタ（DMA_CNDTRx）

アドレスオフセット：0x0C + 0x14 * (x - 1)、（x = 1～8）

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]**：転送データ数（0～2^{16} - 1）

チャネルが有効な場合、このフィールドはハードウェアによって更新されます。

- このフィールドは、書込み転送に続く各シングル DMA 読出し後にデクリメントされ、残りの転送データ項目数を示します。
- チャネルがサーキュラモードでない場合（DMA_CCRx レジスタの CIRC = 0）、プログラムされた転送データ数に達すると、このフィールドはゼロに保持されます。
- チャネルがサーキュラモードの場合（CIRC = 1）、転送完了時に、このフィールドには前もってプログラムされた値が自動的に再ロードされます。

このフィールドがゼロの場合、チャネルの状態（有効／無効）にかかわらず、転送は処理できません。

注： このフィールドは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用です。

12.6.5 DMA チャネル x ペリフェラルアドレスレジスタ（DMA_CPARx）

アドレスオフセット：0x10 + 0x14 * (x - 1)、（x = 1～8）

リセット値：0x0000 0000

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

ビット 31:0 **PA[31:0]**：ペリフェラルアドレス

このビットは、データの読出し／書込みが行われるペリフェラルデータレジスタのベースアドレスを含みます。

PSIZE[1:0] = 01（16 ビット）の場合、PA[31:0] のビット 0 は無視されます。アクセスは、自動的にハーフワードアドレスにアライメントされます。

PSIZE = 10（32 ビット）の場合、PA[31:0] のビット 1 および 0 は無視されます。アクセスは、自動的にワードアドレスにアライメントされます。

メモリ間モードでは、このレジスタは DIR = 1 の場合はメモリ転送先アドレスを、DIR = 0 の場合はメモリ転送元アドレスを示します。

ペリフェラル間モードでは、このレジスタは DIR = 1 の場合はペリフェラル転送先アドレスを、DIR = 0 の場合はペリフェラル転送元アドレスを示します。

注： このレジスタは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

12.6.6 DMA チャネル x メモリアドレスレジスタ（DMA_CMARx）

アドレスオフセット：0x14 + 0x14 * (x - 1)、（x = 1〜8）

リセット値：0x0000 0000

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

ビット 31:0 **MA[31:0]**：ペリフェラルアドレス

このビットは、データの読出し／書込みが行われるメモリのベースアドレスを含みます。

MSIZE[1:0] = 01（16 ビット）の場合、MA[31:0] のビット 0 は無視されます。アクセスは、自動的にハーフワードアドレスにアライメントされます。

MSIZE = 10（32 ビット）の場合、MA[31:0] のビット 1 および 0 は無視されます。アクセスは、自動的にワードアドレスにアライメントされます。

メモリ間モードでは、このレジスタは DIR = 1 の場合はメモリ転送元アドレスを、DIR = 0 の場合はメモリ転送先アドレスを示します。

ペリフェラル間モードでは、このレジスタは DIR = 1 の場合はペリフェラル転送元アドレスを、DIR = 0 の場合はペリフェラル転送先アドレスを示します。

注：　このレジスタは、ソフトウェアによってセット／クリアされます。
チャネルが有効な場合（EN = 1）は書き込まないでください。
チャネルが有効な場合（EN = 1）は読出し専用ではありません。

12.6.7 DMA レジスタ･マップ

次の表に、DMA のレジスタマップとリセット値を示します。

表 89. DMA レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMA_ISR	TEIF8	HTIF8	TCIF8	GIF8	TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	DMA_IFCR	CTEIF8	CHTIF8	CTCIF8	CGIF8	CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x008	DMA_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	DMA_CNDTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	DMA_CPAR1	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	DMA_CMAR1	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	予約済みです。	予約済みです。																															
0x01C	DMA_CCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	DMA_CNDTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	DMA_CPAR2	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	DMA_CMAR2	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	予約済み	予約済み																															
0x030	DMA_CCR3	Res.	Res.	Res.	Res.	Res	Res	Res	Res	Res.	Res.	Res	Res	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	DMA_CNDTR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	DMA_CPAR3	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	DMA_CMAR3	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	予約済み	予約済み																															
0x044	DMA_CCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	DMA_CNDTR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	DMA_CPAR4	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	DMA_CMAR4	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	予約済み	予約済み																															

表 89. DMA レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x058	DMA_CCR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	DMA_CNDTR5	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060	DMA_CPAR5	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x064	DMA_CMAR5	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x068	予約済み	予約済み。																															
0x06C	DMA_CCR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x070	DMA_CNDTR6	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x074	DMA_CPAR6	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x078	DMA_CMAR6	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x07C	予約済み	予約済み。																															
0x080	DMA_CCR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	DMA_CNDTR7	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x088	DMA_CPAR7	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08C	DMA_CMAR7	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x090	予約済み	予約済み。																															
0x094	DMA_CCR8	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MEM2MEM	PL[1:0]		MSIZE[1:0]		PSIZE[1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x098	DMA_CNDTR8	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NDTR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x09C	DMA_CPAR8	PA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0	DMA_CMAR8	MA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2 : を参照してください。

13 DMA リクエスト･マルチプレクサ（DMAMUX）

13.1 概要

ペリフェラルは DMA リクエスト信号の設定による DMA 転送のリクエストを示します。DMA リクエストは､DMA 確認応答信号を生成する DMA コントローラで処理されるまで保留され､対応する DMA リクエスト信号はデアサートされます。

本書では、DMA リクエスト／確認応答プロトコルに必要な一連の制御信号を明示的に示したり説明したりしていませんが、これは DMA リクエストラインとして参照します。

DMAMUX リクエスト･マルチプレクサにより、製品のペリフェラルと DMA コントローラ間の DMA リクエスト･ラインをルーティングできます。ルーティング機能は、プログラム可能なマルチチャネル DMA リクエスト･ライン･マルチプレクサにより確保されます。各チャネルは、DMAMUX 同期入力のイベントで無条件に、または同期的に一意の DMA リクエストラインを選択します。DMAMUX は、入力トリガ信号のプログラム可能なイベントからの DMA リクエストジェネレータとしても使用できます。

DMAMUX 構成の数およびそれらの主な特性は、セクション 13.3.1 : で指定されています。

ペリフェラルからの DMA リクエスト･ラインおよび DMAMUX リクエスト･ジェネレータ出力への DMAMUX リクエスト･マルチプレクサ入力の割り当て、DMA コントローラ･チャネルへの DMAMUX リクエスト･マルチプレクサ出力の割り当て、および内部および外部信号への DMAMUX 同期およびトリガ入力の割り当ては、製品の実装に依存しています。詳細については、セクション 13.3.2 : を参照してください。

13.2 DMAMUX の主な機能

- 最大 16 チャネルのプログラム可能な DMA リクエスト･ライン･マルチプレクサ出力
- 4 チャネルの DMA リクエストジェネレータ
- 21 の DMA リクエストジェネレータへのトリガ入力
- 21 の同期入力
- DMA リクエストジェネレータチャネル：
  - DMA リクエストトリガ入力セレクタ
  - DMA リクエストカウンタ
  - 選択された DMA リクエストトリガ入力のイベントオーバーランフラグ
- DMA リクエストラインマルチプレクサチャネル出力：
  - 115 のペリフェラルからの DMA リクエストラインの入力
  - 1 つの DMA リクエストライン出力
  - 同期入力セレクタ
  - DMA リクエストカウンタ
  - 選択された同期入力のイベントオーバーランフラグ
  - DMA リクエスト連鎖のための 1 つのイベント出力

13.3 DMAMUX の実装

13.3.1 DMAMUX の構成

DMAMUX は、次の表に記載されているハードウェア設定パラメータで構成されています。

表 90. DMAMUX の構成

機能		DMAMUX
DMAMUX 出力リクエスト･チャネルの数	カテゴリ 2 デバイス(1) カテゴリ 3 デバイス (1) カテゴリ 4 デバイス (1)	12 16 16
DMAMUX リクエストジェネレータチャネルの数		4
DMAMUX リクエストトリガ入力の数		21
DMAMUX 同期入力の数		21
DMAMUX ペリフェラルリクエスト入力の数		115

1. 表 1：STM32G4 シリーズメモリ容量を参照してください。

13.3.2 DMAMUX のマッピング

DMAMUX へのリソースの配置は配線接続されています。

DMAMUX は、DMA1 および DMA2 で使用されます。

カテゴリ 3 および カテゴリ 4 デバイスの場合

- DMAMUX のチャネル 0 から 7 は、DMA1 のチャネル 1 から 8 に接続されます。
- DMAMUX のチャネル 8 から 15 は、DMA2 のチャネル 1 から 8 に接続されます。

カテゴリ 2 デバイスの場合

- DMAMUX のチャネル 0 から 5 は、DMA1 のチャネル 1 から 6 に接続されます。
- DMAMUX のチャネル 6 から 11 は、DMA2 のチャネル 1 から 6 に接続されます。

表 91. DMAMUX：リソースへのマルチプレクサ入力の割り当て(1)

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
1	DMAMUX_Req G0	44	TIM1_CH3	87	TIM20_CH2
2	DMAMUX_Req G1	45	TIM1_CH4	88	TIM20_CH3
3	DMAMUX_Req G2	46	TIM1_UP	89	TIM20_CH4
4	DMAMUX_Req G3	47	TIM1_TRIG	90	TIM20_UP
5	ADC1	48	TIM1_COM	91	AES_IN
6	DAC1_CH1	49	TIM8_CH1	92	AES_OUT
7	DAC1_CH2	50	TIM8_CH2	93	TIM20_TRIG
8	TIM6_UP	51	TIM8_CH3	94	TIM20_COM
9	TIM7_UP	52	TIM8_CH4	95	HRTIM_MASTER（hrtim_dma1）
10	SPI1_RX	53	TIM8_UP	96	HRTIM_TIMA（hrtim_dma2）
11	SPI1_TX	54	TIM8_TRIG	97	HRTIM_TIMB（hrtim_dma3）
12	SPI2_RX	55	TIM8_COM	98	HRTIM_TIMC（hrtim_dma4）
13	SPI2_TX	56	TIM2_CH1	99	HRTIM_TIMD（hrtim_dma5）
14	SPI3_RX	57	TIM2_CH2	100	HRTIM_TIME（hrtim_dma6）
15	SPI3_TX	58	TIM2_CH3	101	HRTIM_TIMF（hrtim_dma7）
16	I2C1_RX	59	TIM2_CH4	102	DAC3_CH1
17	I2C1_TX	60	TIM2_UP	103	DAC3_CH2
18	I2C2_RX	61	TIM3_CH1	104	DAC4_CH1
19	I2C2_TX	62	TIM3_CH2	105	DAC4_CH2
20	I2C3_RX	63	TIM3_CH3	106	SPI4_RX
21	I2C3_TX	64	TIM3_CH4	107	SPI4_TX
22	I2C4_RX	65	TIM3_UP	108	SAI1_A
23	I2C4_TX	66	TIM3_TRIG	109	SAI1_B
24	USART1_RX	67	TIM4_CH1	110	FMAC_Read
25	USART1_TX	68	TIM4_CH2	111	FMAC_Write
26	USART2_RX	69	TIM4_CH3	112	Cordic_Read

表 91. DMAMUX：リソースへのマルチプレクサ入力の割り当て(1)（続き）

DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース	DMA リクエスト MUX 入力	リソース
27	USART2_TX	70	TIM4_CH4	113	Cordic_Write
28	USART3_RX	71	TIM4_UP	114	UCPD1_RX
29	USART3_TX	72	TIM5_CH1	115	UCPD1_TX
30	UART4_RX	73	TIM5_CH2	116	予約済みです。
31	UART4_TX	74	TIM5_CH3	117	予約済みです。
32	UART5_RX	75	TIM5_CH4	118	予約済みです。
33	UART5_TX	76	TIM5_UP	119	予約済みです。
34	LPUART1_RX	77	TIM5_TRIG	120	予約済みです。
35	LPUART1_TX	78	TIM15_CH1	121	予約済みです。
36	ADC2	79	TIM15_UP	122	予約済みです。
37	ADC3	80	TIM15_TRIG	123	予約済みです。
38	ADC4	81	TIM15_COM	124	予約済みです。
39	ADC5	82	TIM16_CH1	125	予約済みです。
40	QUADSPI	83	TIM16_UP	126	予約済みです。
41	DAC2_CH1	84	TIM17_CH1	127	予約済みです。
42	TIM1_CH1	85	TIM17_UP	-	-
43	TIM1_CH2	86	TIM20_CH1	-	-

1. 使用可能なリソースについては、表 2：製品固有の機能を参照してください。

表 92. DMAMUX：リソースへのトリガ入力の割り当て

トリガ入力	リソース	トリガ入力	リソース
0	EXTI ライン 0	16	DMAMUX1_ch0_event
1	EXTI ライン 1	17	DMAMUX1_ch1_event
2	EXTI ライン 2	18	DMAMUX1_ch2_event
3	EXTI ライン 3	19	DMAMUX1_ch3_event
4	EXTI ライン 4	20	LPTIM1_OUT
5	EXTI ライン 5	21	予約済みです。
6	EXTI ライン 6	22	予約済みです。
7	EXTI ライン 7	23	予約済みです。
8	EXTI ライン 8	24	予約済みです。
9	EXTI ライン 9	25	予約済みです。
10	EXTI ライン 10	26	予約済みです。
11	EXTI ライン 11	27	予約済みです。
12	EXTI ライン 12	28	予約済みです。
13	EXTI ライン 13	29	予約済みです。

表 92. DMAMUX：リソースへのトリガ入力の割り当て （続き）

トリガ入力	リソース	トリガ入力	リソース
14	EXTI ライン 14	30	予約済みです。
15	EXTI ライン 15	31	予約済みです。

表 93. DMAMUX：リソースへの同期入力の割り当て

同期入力	リソース	同期入力	リソース
0	EXTI ライン 0	16	DMAMUX1_ch0_event
1	EXTI ライン 1	17	DMAMUX1_ch1_event
2	EXTI ライン 2	18	DMAMUX1_ch2_event
3	EXTI ライン 3	19	DMAMUX1_ch3_event
4	EXTI ライン 4	20	LPTIM1_OUT
5	EXTI ライン 5	21	予約済みです。
6	EXTI ライン 6	22	予約済みです。
7	EXTI ライン 7	23	予約済みです。
8	EXTI ライン 8	24	予約済みです。
9	EXTI ライン 9	25	予約済みです。
10	EXTI ライン 10	26	予約済みです。
11	EXTI ライン 11	27	予約済みです。
12	EXTI ライン 12	28	予約済みです。
13	EXTI ライン 13	29	予約済みです。
14	EXTI ライン 14	30	予約済みです。
15	EXTI ライン 15	31	予約済みです。

13.4 DMAMUX の機能詳細

13.4.1 DMAMUX のブロック図

図 32 に DMAMUX のブロック図を示します。

図 32. DMAMUX のブロック図

DMAMUX には、リクエスト･ライン･マルチプレクサとリクエスト･ライン･ジェネレータの 2 つの主なサブブロックがあります。

実装時の割当ては次のとおりです。

- ペリフェラル（dmamux_req_inx）および DMAMUX リクエスト･ジェネレータ･サブブロックのチャネル（dmamux_req_genx）からの DMAMUX リクエスト･マルチプレクサ･サブブロック入力（dmamux_reqx）
- DMA コントローラのチャネルへの DMAMUX リクエスト出力（dmamux_req_outx）から
- DMA リクエスト･トリガ入力への内部または外部信号（dmamux_trgx）
- 同期入力への内部または外部信号（dmamux_syncx）

13.4.2 DMAMUX 信号

表 94 に DMAMUX 信号をリストします。

表 94. DMAMUX 信号

信号名	説明
dmamux_hclk	DMAMUX AHB クロック
dmamux_req_inx	DMAMUX DMA リクエストライン入力（ペリフェラルから）
dmamux_trgx	DMAMUX DMA リクエストトリガ入力（リクエストジェネレータサブブロックへ）
dmamux_req_genx	DMAMUX リクエストジェネレータサブブロックチャネル出力
dmamux_reqx	DMAMUX リクエスト･マルチプレクサ･サブブロック入力（ペリフェラル･リクエストおよびリクエスト･ジェネレータ･チャネルから）
dmamux_syncx	DMAMUX 同期入力（リクエストマルチプレクササブブロックへ）
dmamux_req_outx	DMAMUX リクエスト出力（DMA コントローラへ）
dmamux_evtx	DMAMUX イベント出力
dmamux_ovr_it	DMAMUX オーバーラン割込み

13.4.3 DMAMUX チャネル

DMAMUX チャネルは、リクエストマルチプレクサで選択された入力に応じて、追加の DMAMUX リクエストジェネレータチャネルを含む、DMAMUX リクエストマルチプレクサチャネルです。

DMAMUX リクエスト･マルチプレクサ･チャネルは、DMA コントローラの単一チャネル専用の接続です。

チャネル設定手順

DMAMUX の x チャネルと関連する DMA チャネル y の両方を設定するには、次のシーケンスに従ってください。

1. DMA チャネル y をセットし、チャネル y の有効化を除くすべての設定を行います。
2. 関連する DMAMUX y チャネルをセットし、すべての設定を行います。
3. 最後に、DMA y チャネルレジスタの EN ビットをセットして DMA チャネル y を有効にします。

13.4.4 DMAMUX リクエストラインマルチプレクサ

DMAMUX リクエストマルチプレクサには複数のチャネルがあるため、DMA リクエストラインと呼ばれる DMA リクエスト／確認応答制御信号の実際のルーティングが確保されます。

各DMA リクエストラインは、DMAMUX リクエストラインマルチプレクサのすべてのチャネルにそれぞれ並列に接続されます。

DMA リクエストは、ペリフェラルまたは DMAMUX リクエストジェネレータのどちらかから供給されます。

DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの DMAREQ_ID フィールドの設定どおり、DMA リクエストライン番号を選択します。

注： **DMAREQ_IDフィールド が null 値の場合、いずれの DMA リクエストラインも選択されていません。**

注意： **Null ではない同じ DMAREQ_ID を（DMAMUX_CxCR と DMAMUX_CyCR を使用して）異なる x と y の DMAMUX リクエスト･マルチプレクサ･チャネルにプログラムしてはなりません。ただし、接続された 2 つの DMA チャネルが同時にアクティブにならないことをアプリケーション側で保証している場合を除きます。**

DMA リクエストの選択に加えて、必要ならば同期モードまたはイベント生成（あるいはその両方）も設定および有効化できます。

同期モードおよびチャネルイベント生成

各 DMAMUX リクエストラインマルチプレクサチャネル x は、DMAMUX_CxCR レジスタの同期有効化（SE）ビットを設定することで、個別に同期することができます。

DMAMUX には複数の同期入力があります。同期入力は、リクエストマルチプレクサのすべてのチャネルに並列に接続されます。

同期入力は、特定のチャネル x の DMAMUX_CxCR レジスタの SYNC_ID フィールドを介して選択されます。

チャネルが同期モードの場合、DMAMUX_CxCR レジスタの SPOL[1:0] フィールドを介して、選択された入力同期信号でプログラム可能な立ち上がり／立ち下がりエッジを検出すると、選択された DMA リクエストラインの入力はマルチプレクサチャネル出力に伝播されます。

さらに、内部的に DMAMUX リクエストマルチプレクサにはプログラム可能な DMA リクエストカウンタもあります。これは、チャネルリクエスト出力生成あるいはイベント生成にも使用されます。チャネル x 出力でのイベント生成は、DMAMUX_CxCR レジスタの EGE ビット（イベント生成イネーブル）を使用して有効化されます。

図 34 に示すように、同期入力の検出されたエッジで、保留中の選択された DMA リクエストラインの入力は DMAMUX マルチプレクサチャネル x 出力に接続されます。

注： **保留中の選択された DMA リクエストラインがないときに同期イベントが発生した場合、そのイベントは破棄されます。続いてアサートされた入力リクエストラインは、同期イベントが再度発生するまで DMAMUX マルチプレクサチャネル出力に接続されません。**

これ以降、接続されている DMAMUX リクエストが DMA コントローラによって処理されるたびに（処理されたリクエストはデアサートされます）、DMAMUX リクエストカウンタがデクリメントされます。アンダーランの際、DMA リクエストカウンタには DMAMUX_CxCR レジスタの NBREQ フィールドの値が自動的にロードされ、DMA リクエストラインの入力はマルチプレクサチャネル x 出力から切断されます。

このため、検出された同期イベントに続いてマルチプレクサチャネル x 出力に転送される DMA リクエストの数は、NBREQ フィールドの値に 1 を加えた数と等しくなります。

注： **NBREQ フィールド値は、対応するマルチプレクサチャネル x の SE（同期イネーブル）および EGE（イベント生成イネーブル）ビットの両方が無効である場合にのみ、ソフトウェアによって書き込むことができます。**

図 33. DMAMUX リクエスト･ライン･マルチプレクサ･チャネルの同期モード

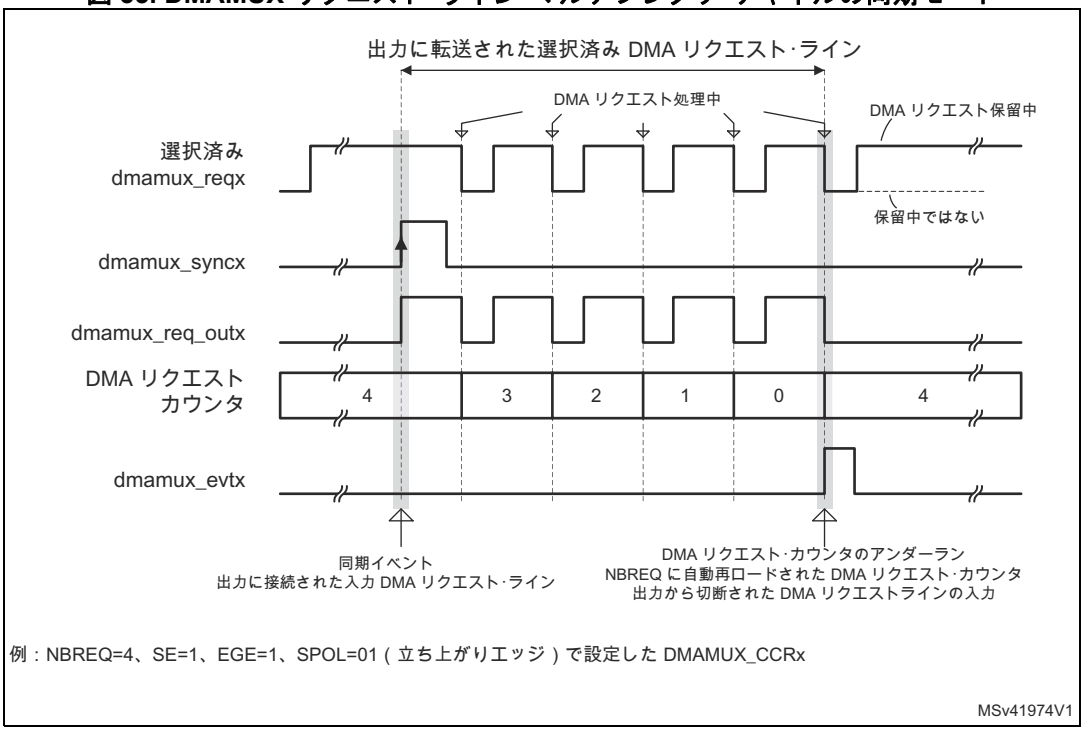


図 34. DMA リクエスト･ライン･マルチプレクサ･チャネルのイベント生成

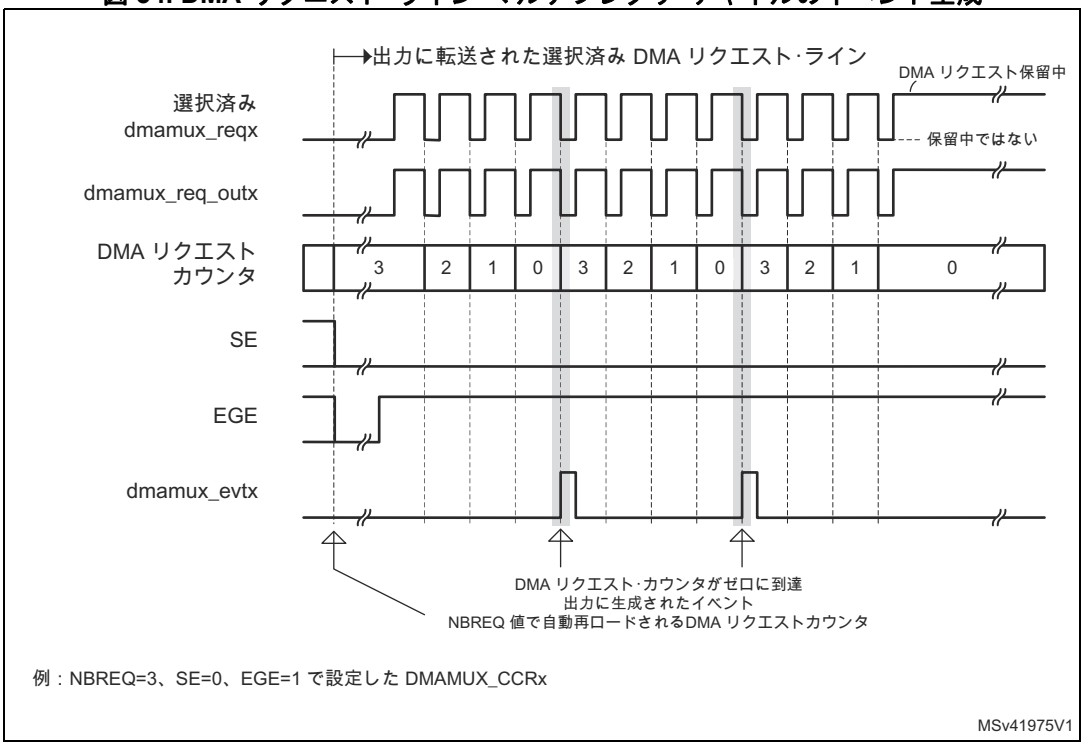


図 33 および図 34 に示したように、EGE が有効である場合、DMA リクエストカウンタにプログラムされた NBREQ フィールドの値が自動的に再ロードされると、マルチプレクサチャネルは 1 つの AHB クロックサイクルのパルスとしてチャネルイベントを生成します。

注： **EGE が有効で NBREQ = 0 の場合、各 DMA リクエストが処理されるたびにイベントが発生します。**

注： **同期イベント（エッジ）は、エッジ後の状態が 2 AHB クロックサイクルよりも長く安定している場合に検出されます。**

DMAMUX_CxCR レジスタに書き込む際、同期イベントは 3 AHB クロックサイクルの間、マスクされます。

同期オーバーランおよび割込み

リクエストカウンタアンダーラン（DMAMUX_CxCR レジスタの NBREQ フィールドを介してプログラムされる内部リクエストカウンタ）の前に新しい同期イベントが発生すると、DMAMUX_CSR ステータスレジスタの同期オーバーランフラグビット SOFx がセットされます。

注： **リクエストマルチプレクサチャネル x 同期は、DMA コントローラの関連するチャネルの使用完了時に無効にする必要があります（DMAMUX_CxCR.SE=0）。そうでない場合、新しく同期イベントを検出したときに、DMA コントローラから受信する DMA 確認応答の欠落（すなわち、未処理のリクエスト）によって同期オーバーランが発生します。**

オーバーラン･フラグ SOFx は、DMAMUX_CFR レジスタの関連する同期オーバーラン･フラグ･クリア･ビット CSOFx をセットすることでリセットされます。

DMAMUX_CxCR レジスタの同期オーバーラン割込みイネーブル･ビット SOIE がセットされている場合、同期オーバーラン･フラグがセットされると、割込みが生成されます。

13.4.5 DMAMUX リクエストジェネレータ

DMAMUX リクエストジェネレータでは、DMA リクエストトリガ入力のトリガイベントに続いて、DMA リクエストを生成します。

DMAMUX リクエストジェネレータには複数のチャネルがあります。DMA リクエストトリガ入力は、すべてのチャネルに並列に接続されます。

DMAMUX リクエストジェネレータチャネルの出力は、DMAMUX リクエストラインマルチプレクサへの入力です。

各 DMAMUX リクエストジェネレータチャネル x には、対応する DMAMUX_RGxCR レジスタにイネーブルビット GE （ジェネレータイネーブル）があります。

DMAMUX リクエストジェネレータチャネル x の DMA リクエストトリガ入力は、対応する DMAMUX_RGxCR レジスタの SIG_ID （トリガ信号 ID）フィールドによって選択されます。

DMA リクエストトリガ入力のトリガイベントは立ち上がりエッジと立ち下がりエッジのいずれかです。アクティブなエッジは対応する DMAMUX_RGxCR レジスタの GPOL（ジェネレータ極性）フィールドによって選択されます。

トリガイベント時、対応するジェネレータチャネルでは出力時に DMA リクエストの生成を開始します。DMAMUX で生成されたリクエストが接続されている DMA コントローラによって処理されるたびに（処理されたリクエストはデアサートされます）、内蔵（DMAMUX リクエストジェネレータ内の）DMA リクエストカウンタがデクリメントされます。アンダーランの際、リクエスト･ジェネレータ･チャネルは DMA リクエストの生成を停止し、次のトリガ･イベント時に DMA リクエスト･カウンタはプログラムされた値に自動的に再ロードされます。

このため、トリガ･イベント後に生成される DMA リクエストの数は GNBREQ + 1 です。

注： **GNBREQ フィールド値は、対応するジェネレータチャネル x のイネーブル GE ビットが無効である場合にのみ、ソフトウェアによって書き込まれなければなりません。**

ハードウェア書込み保護はありません。

トリガ･イベント（エッジ）は、エッジ後の状態が 2 AHB クロック･サイクルよりも長く安定している場合に検出されます。

DMAMUX_RGxCR レジスタに書き込む際、トリガイベントは 3 AHB クロックサイクルの間、マスクされます。

トリガオーバーランおよび割込み

DMAMUX リクエストジェネレータカウンタアンダーラン（DMAMUX_RGxCR レジスタの GNBREQ フィールドを介してプログラムされる内部カウンタ）の前に新しい DMA リクエストトリガイベントが発生し、かつ、リクエストジェネレータチャネル x が GE によって有効化されている場合、リクエストトリガイベントオーバーランフラグビット OFx は、ハードウェアによってステータス DMAMUX_RGSR レジスタにアサートされます。

注： **リクエストジェネレータチャネル x は、DMA コントローラの関連するチャネルの使用完了時に無効にする必要があります（DMAMUX_RGxCR.GE = 0）。そうでない場合、新しくトリガイベントを検出したときに、DMA から受信する確認応答の欠落（すなわち、未処理のリクエスト）によってトリガオーバーランが発生します。**

オーバーランフラグ OFx は、DMAMUX_RGCFR レジスタの関連するクリアオーバーランフラグビット COFx をセットすることでリセットされます。

DMAMUX_RGxCR レジスタの DMA リクエストトリガイベントオーバーラン割込みイネーブルビット OIE がセットされている場合、DMAMUX リクエストトリガオーバーランフラグによって、割込みが生成されます。

13.5 DMAMUX 割込み

次の場合に割込みを生成することができます。

- 各 DMA リクエストラインマルチプレクサチャネルの同期イベントオーバーラン
- 各 DMA リクエストジェネレータチャネルのトリガイベントオーバーラン

それぞれのケースで、チャネルごとの個別の割込みイネーブル、ステータス、およびクリアフラグレジスタビットを使用できます。

表 95. DMAMUX 割込み

割込み信号	割込みイベント	イベントフラグ	クリアビット	イネーブルビット
dmamuxovr_it	DMAMUX リクエスト･ライン･マルチプレクサのチャネル x での同期イベント･オーバーラン	SOFx	CSOFx	SOIE
	DMAMUX リクエスト･ジェネレータのチャネル x でのトリガ･イベント･オーバーラン	OFx	COFx	OIE

13.6 DMAMUX レジスタ

DMAMUX のベース･アドレスについては、レジスタ境界アドレスを記載した表を参照してください。

DMAMUX レジスタには、バイト（8 ビット）、ハーフワード（16 ビット）、またはワード（32 ビット）でアクセスできます。アドレスはデータサイズに合わせる必要があります。

13.6.1 DMAMUX リクエスト･ライン･マルチプレクサ･チャネル x 設定レジスタ（DMAMUX_CxCR）

アドレスオフセット：0x000 + 0x04 * x（x = 0～ 15）

リセット値：0x0000 0000

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

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SYNC_ID[4:0]**：同期識別

同期入力を選択します（表 93：DMAMUX：リソースへの同期入力の割り当てを参照）。

ビット 23:19 **NBREQ[4:0]**：転送する DMA リクエストの数から 1 を引いた数

同期イベントの後で DMA コントローラに転送する DMA リクエストの数、および／または出力イベントが生成される前の DMA リクエストの数を定義します。

このフィールドは、SE と EGE ビットの両方がローの場合にのみ書き込まれます。

ビット 18:17 **SPOL[1:0]**：同期極性

選択された同期入力のエッジ極性を定義します。

00：イベントなし、つまり同期も検出もなし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がりおよび立ち下がりエッジ

ビット 16 **SE**：同期の有効化

0：同期は無効です。

1：同期は有効です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EGE**：イベント生成の有効化

0：イベント生成は無効です。

1：イベント生成は有効です。

ビット 8 **SOIE**：同期オーバーラン割込みの有効化

0：割込みは無効です。

1：割込みは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **DMAREQ_ID[6:0]**：DMA リクエストID

DMA リクエストの入力を選択しますリソースへのマルチプレクサ入力の割当てについては、DMAMUX の表を参照してください。

13.6.2 DMAMUX リクエストラインマルチプレクサ割込みチャネルステータスレジスタ（DMAMUX_CSR）

アドレスオフセット：0x080

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SOF[15:0]**：同期オーバーランイベントフラグ

このフラグは、DMA リクエストカウンタの値が NBREQ よりも低い場合に、DMA リクエストラインマルチプレクサチャネル x で同期イベントが発生したときにセットされます。

このフラグは、対応する DMAMUX_CFR レジスタの CSOFx ビットに 1 を書き込むことでクリアされます。

13.6.3 DMAMUX リクエストラインマルチプレクサ割込みクリアフラグレジスタ（DMAMUX_CFR）

アドレス･オフセット：0x084

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CSOF[15:0]**：同期オーバーランイベントフラグをクリアします。

各ビットに 1 を書き込むと DMAMUX_CSR レジスタの対応するオーバーラン･フラグ SOFx がクリアされます。

13.6.4 DMAMUX リクエスト･ジェネレータ･チャネル x 設定レジスタ（DMAMUX_RGxCR）

アドレスオフセット：0x100 + 0x04 * x（x = 0～ 3）

リセット値：0x0000 0000

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

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:19 **GNBREQ[4:0]**：生成される DMA リクエストの数（1 を引く）

トリガイベント後に生成される DMA リクエストの数を定義します。実際に生成される DMA リクエストの数は GNBREQ + 1 です。

注： このフィールドに書き込めるのは、GE が無効なときだけです。

ビット 18:17 **GPOL[1:0]**：DMA リクエストジェネレータトリガ極性

選択されたトリガ入力のエッジ極性を定義します。

00：イベントなし、つまりトリガの検出も生成もなし

01：立ち上がりエッジ

10：立ち下がりエッジ

11：立ち上がりおよび立ち下がりエッジ

ビット 16 **GE**：DMA リクエスト･ジェネレータ･チャネル x の有効化

0：DMA リクエストジェネレータチャネル x は無効です。

1：DMA リクエストジェネレータチャネル x は有効です。

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OIE**：トリガ･オーバーラン割込みの有効化

0：トリガ･オーバーラン･イベント発生時の割込みは無効です。

1：トリガ･オーバーラン･イベント発生時の割込みは有効です。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SIG_ID[4:0]**：信号 ID

DMA リクエストジェネレータのチャネル x で使用する DMA リクエストトリガ入力を選択します。

13.6.5 DMAMUX リクエストジェネレータ割込みステータスレジスタ（DMAMUX_RGSR）

アドレスオフセット：0x140

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF3	OF2	OF1	OF0
												r	r	r	r

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **OF[3:0]**：トリガオーバーランイベントフラグ

このフラグは、リクエストカウンタアンダーラン（DMAMUX_RGxCR レジスタの GNBREQ フィールドを介してプログラムされる内部リクエストカウンタ）の前に、DMA リクエストジェネレータチャネル x で新しいトリガイベントが発生したときにセットされます。

このフラグは、DMAMUX_RGCFR レジスタの対応する COFx ビットに 1 を書き込むことでクリアされます。

13.6.6 DMAMUX リクエスト･ジェネレータ割込みクリア･フラグ･レジスタ（DMAMUX_RGCFR）

アドレスオフセット：0x144

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF3	COF2	COF1	COF0
												w	w	w	w

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **COF[3:0]**：トリガオーバーランイベントフラグのクリア

各ビットに 1 を書き込むと DMAMUX_RGSR レジスタの対応するオーバーランフラグ OFx をクリアします。

13.6.7 DMAMUX レジスタ･マップ

次の表に、DMAMUX レジスタとリセット値の一覧を示します。DMAMUX レジスタベースアドレスについては、レジスタ境界アドレス表を参照してください。

表 96. DMAMUX レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DMAMUX_C0CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x004	DMAMUX_C1CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x008	DMAMUX_C2CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x00C	DMAMUX_C3CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x010	DMAMUX_C4CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x014	DMAMUX_C5CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x018	DMAMUX_C6CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x01C	DMAMUX_C7CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x020	DMAMUX_C8CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x024	DMAMUX_C9CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x028	DMAMUX_C10CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x02C	DMAMUX_C11CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x030	DMAMUX_C12CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x034	DMAMUX_C13CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x038	DMAMUX_C14CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x03C	DMAMUX_C15CR	Res.	Res.	Res.	SYNC_ID[4:0]					NBREQ[4:0]					SPOL[1:0]		SE	Res.	Res.	Res.	Res.	Res.	Res.	EGE	SOIE	Res.	DMAREQ_ID[6:0]						
	リセット値				0	0	0	0	0	0	0	0	0	0	0	0	0							0	0		0	0	0	0	0	0	0
0x040～0x07C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x080	DMAMUX_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SOF15	SOF14	SOF13	SOF12	SOF11	SOF10	SOF9	SOF8	SOF7	SOF6	SOF5	SOF4	SOF3	SOF2	SOF1	SOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084	DMAMUX_CCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CSOF15	CSOF14	CSOF13	CSOF12	CSOF11	CSOF10	CSOF9	CSOF8	CSOF7	CSOF6	CSOF5	CSOF4	CSOF3	CSOF2	CSOF1	CSOF0
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 96. DMAMUX レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x088 - 0x0FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x100	DMAMUX_RG0CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x104	DMAMUX_RG1CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x108	DMAMUX_RG2CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x10C	DMAMUX_RG3CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GNBREQ[4:0]					GPOL[1:0]		GE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIE	Res.	Res.	Res.	SIG_ID[4:0]				
	リセット値									0	0	0	0	0	0	0	0								0				0	0	0	0	0
0x110 - 0x13C	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x140	DMAMUX_RGSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OF3	OF2	OF1	OF0
	リセット値																													0	0	0	0
0x144	DMAMUX_RGCFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COF3	COF2	COF1	COF0
	リセット値																													0	0	0	0
0x148 - 0x3FC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

14 ネスト化されたベクタ割込みコントローラ（NVIC）

14.1 NVIC の主な機能

- マスク可能な 120 の割込みチャネル（FPU 搭載 Cortex®-M4 の 16 本の割込みラインはこの数に含まれていない）
- 16 のプログラム可能な優先レベル（4 ビットの割込み優先順位を使用）
- 遅延時間の少ない例外および割込み処理
- 電源管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインタフェースは密に結合され、割込み処理の遅延時間を小さくし、遅れて到着した割込みを効率的に処理できます。

コア例外を含むすべての割込みは、NVIC によって管理されます。例外と NVIC プログラミングの詳細については、Cortex™-M4 製品のプログラミングマニュアル PM0214 を参照してください。

14.2 SysTick 較正値レジスタ

SysTick 較正値は 0x3E8 にセットされますので、SysTick クロックを 125 KHz（1 MHz/8）に設定した状態での基準タイムベースは 1 ms になります。

14.3 割込みベクタと例外ベクタ

表 97 の灰色の行は、特定の位置を持たないベクタを示しています。

表 97. STM32G4 シリーズ ベクタテーブル

位置	優先順位	優先種別	項目（略称）	説明	アドレス
-	-	-	-	予約済みです。	0x0000 0000
-	-3	固定	リセット	リセット	0x0000 0004
-	-2	固定	NMI	ノンマスカブル割込み。SRAM パリティエラー + Flash ECC エラー + HSE CSS	0x0000 0008
-	-1	固定	HardFault	あらゆる種類の異常	0x0000 000C
-	0	設定可能	MemManage	メモリ管理	0x0000 0010
-	1	設定可能	BusFault	プリフェッチ異常、メモリアクセス異常	0x0000 0014
-	2	設定可能	UsageFault	未定義命令または無効状態	0x0000 0018
-	-	-	-	予約済みです。	0x0000 001C ~ 0x0000 0028
-	3	-	-	SWI 命令によるシステムサービスコール	0x0000 002C
-	4	-	-	監視	0x0000 0030
-	-	-	-	予約済みです。	0x0000 0034
-	5	設定可能	PendSV	ペンディング可能なシステムサービスリクエスト	0x0000 0038
-	6	設定可能	SysTick	システムティックタイマ	0x0000 003C
0	7	設定可能	WWDG	ウィンドウ型ウォッチドッグ割込み	0x0000 0040
1	8	設定可能	PVD_PVM	PVD 割込み（EXTI ライン 16 経由）	0x0000 0044
2	9	設定可能	RTC/TAMP/CSS_LSE	RTC/TAMP/LSE の CSS（EXTI ライン 19 経由）割込み	0x0000 0048
3	10	設定可能	RTC_WKUP	RTC ウェイクアップタイマ（EXTI ライン 20 経由）割込み	0x0000 004C
4	11	設定可能	FLASH	Flash グローバル割込み	0x0000 0050
5	12	設定可能	RCC	RCC グローバル割込み	0x0000 0054
6	13	設定可能	EXTI0	EXTI ライン 0 割込み	0x0000 0058
7	14	設定可能	EXTI1	EXTI ライン 1 割込み	0x0000 005C
8	15	設定可能	EXTI2	EXTI ライン 2 割込み	0x0000 0060
9	16	設定可能	EXTI3	EXTI ライン 3 割込み	0x0000 0064
10	17	設定可能	EXTI4	EXTI ライン 4 割込み	0x0000 0068
11	18	設定可能	DMA1_CH1	DMA1 チャネル 1 割込み	0x0000 006C
12	19	設定可能	DMA1_CH2	DMA1 チャネル 2 割込み	0x0000 0070
13	20	設定可能	DMA1_CH3	DMA1 チャネル 3 割込み	0x0000 0074
14	21	設定可能	DMA1_CH4	DMA1 チャネル 4 割込み	0x0000 0078
15	22	設定可能	DMA1_CH5	DMA1 チャネル 5 割込み	0x0000 007C
16	23	設定可能	DMA1_CH6	DMA1 チャネル 6 割込み	0x0000 0080

表 97. STM32G4 シリーズ ベクタテーブル （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
17	24	設定可能	DMA1_CH7	DMA1 チャネル 7 割込み	0x0000 0084
18	25	設定可能	ADC1_2	ADC1 および ADC2 グローバル割込み	0x0000 0088
19	26	設定可能	USB_HP	USB 高優先割込み	0x0000 008C
20	27	設定可能	USB_LP	USB 低優先割込み	0x0000 0090
21	28	設定可能	FDCAN1_IT0	FDCAN1 割込み 0	0x0000 0094
22	29	設定可能	FDCAN1_IT1	FDCAN1 割込み 1	0x0000 0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割込み	0x0000 009C
24	31	設定可能	TIM1_BRK/TIM15	TIM1 ブレーク／TIM15 グローバル割込み	0x0000 00A0
25	32	設定可能	TIM1_UP/TIM16	TIM1 更新／TIM16 グローバル割込み	0x0000 00A4
26	33	設定可能	TIM1_TRG_COM/ TIM17/ TIM1_DIR/TIM1_IDX	TIM1 トリガおよび転流／TIM17 割込み／ TIM1 方向変更割込み／TIM1 インデックス	0x0000 00A8
27	34	設定可能	TIM1_CC	TIM1 キャプチャ / 比較割込み	0x0000 00AC
28	35	設定可能	TIM2	TIM2 グローバル割込み	0x0000 00B0
29	36	設定可能	TIM3	TIM3 グローバル割込み	0x0000 00B4
30	37	設定可能	TIM4	TIM4 グローバル割込み	0x0000 00B8
31	38	設定可能	I2C1_EV	I2C1 イベント割込みおよび EXTI ライン 23 割込み	0x0000 00BC
32	39	設定可能	I2C1_ER	I2C1 エラー割込み	0x0000 00C0
33	40	設定可能	I2C2_EV	I2C2 イベント割込みおよび EXTI ライン 24 割込み	0x0000 00C4
34	41	設定可能	I2C2_ER	I2C2 エラー割込み	0x0000 00C8
35	42	設定可能	SPI1	SPI1 グローバル割込み	0x0000 00CC
36	43	設定可能	SPI2	SPI2 グローバル割込み	0x0000 00D0
37	44	設定可能	USART1	USART1 グローバル割込みおよび EXTI ライン 25	0x0000 00D4
38	45	設定可能	USART2	USART2 グローバル割込みおよび EXTI ライン 26	0x0000 00D8
39	46	設定可能	USART3	USART3 グローバル割込みおよび EXTI ライン 28	0x0000 00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割込み	0x0000 00E0
41	48	設定可能	RTC_ALARM	RTC アラーム割込み	0x0000 00E4
42	49	設定可能	USBWakeUp	サスペンドからの USB ウェイクアップ （EXTI ライン 18）	0x0000 00E8
43	50	設定可能	TIM8_BRK/TIM8_TERR/TIM8_IERR	TIM8 ブレーク割込み／TIM8 遷移エラー／ TIM8 インデックスエラー	0x0000 00EC
44	51	設定可能	TIM8_UP	TIM8 更新割込み	0x0000 00F0
45	52	設定可能	TIM8_TRG_COM/TIM8_DIR/TIM8_IDX	TIM8 トリガおよび転流割込み／ TIM8 方向変更割込み／TIM8 インデックス	0x0000 00F4
46	53	設定可能	TIM1_CC	TIM8 キャプチャ /比較割込み	0x0000 00F8
47	54	設定可能	ADC3	ADC3 グローバル割込み	0x0000 00FC
48	55	設定可能	FSMC	FSMC グローバル割込み	0x0000 0100

表 97. STM32G4 シリーズ ベクタテーブル （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
49	56	設定可能	LPTIM1	LPTIM1 グローバル割込み	0x0000 0104
50	57	設定可能	TIM5	TIM5 グローバル割込み	0x0000 0108
51	58	設定可能	SPI3	SPI3 グローバル割込み	0x0000 010C
52	59	設定可能	UART4	UART4 グローバル割込みおよび EXTI ライン 34 割込み	0x0000 0110
53	60	設定可能	UART5	UART5 グローバル割込みおよび EXTI ライン 35 割込み	0x0000 0114
54	61	設定可能	TIM6_DACUNDER	TIM6 および DAC1/3 アンダーラン･グローバル割込み	0x0000 0118
55	62	設定可能	TIM7_DACUNDER	TIM7 および DAC2/4 アンダーラン･グローバル割込み	0x0000 011C
56	63	設定可能	DMA2_CH1	DMA2 チャネル 1 割込み	0x0000 0120
57	64	設定可能	DMA2_CH2	DMA2 チャネル 2 割込み	0x0000 0124
58	65	設定可能	DMA2_CH3	DMA2 チャネル 3 割込み	0x0000 0128
59	66	設定可能	DMA2_CH4	DMA2 チャネル 4 割込み	0x0000 012C
60	67	設定可能	DMA2_CH5	DMA2 チャネル 5 割込み	0x0000 0130
61	68	設定可能	ADC4	ADC4 グローバル割込み	0x0000 0134
62	69	設定可能	ADC5	ADC5 グローバル割込み	0x0000 0138
63	70	設定可能	UCPD1 グローバル割込み	UCPD1 グローバル割込みおよび EXTI ライン 43	0x0000 013C
64	71	設定可能	COMP1_2_3	COMP1/COMP2/COMP3（EXTI ライン 21/22/29 経由）割込み	0x0000 0140
65	72	設定可能	COMP4_5_6	COMP4/COMP5/COMP6（EXTI ライン 30/31/32 経由）割込み	0x0000 0144
66	73	設定可能	COMP7	グローバル割込み COMP7（EXTI ライン 33 経由）	0x0000 0148
67	74	設定可能	HRTIM_Master_IRQn	HRTIM マスタタイマ割込み（hrtim_it1）	0x0000 014C
68	75	設定可能	HRTIM_TIMA_IRQn	HRTIM タイマ A 割込み（hrtim_it2）	0x0000 0150
69	76	設定可能	HRTIM_TIMB_IRQn	HRTIM タイマ B 割込み（hrtim_it3）	0x0000 0154
70	77	設定可能	HRTIM_TIMC_IRQn	HRTIM タイマ C 割込み（hrtim_it4）	0x0000 0158
71	78	設定可能	HRTIM_TIMD_IRQn	HRTIM タイマ D 割込み（hrtim_it5）	0x0000 015C
72	79	設定可能	HRTIM_TIME_IRQn	HRTIM タイマ E 割込み（hrtim_it6）	0x0000 0160
73	80	設定可能	HRTIM_TIM_FLT_IRQn	HRTIM 障害割込み（hrtim_it8）	0x0000 0164
74	81	設定可能	HRTIM_TIMF_IRQn	hrtim_it7 / HRTIM タイマ F 割込み	0x0000 0168
75	82	設定可能	CRS	CRS 割込み	0x0000 016C
76	83	設定可能	SAI	SAI	0x0000 0170
77	84	設定可能	TIM20_BRK/ TIM20_TERR/ TIM20_IERR	TIM20 ブレーク割込み／TIM20 遷移エラー／TIM20 インデックス･エラー	0x0000 0174
78	85	設定可能	TIM20_UP	TIM20 更新割込み	0x0000 0178

表 97. STM32G4 シリーズ ベクタテーブル （続き）

位置	優先順位	優先種別	項目（略称）	説明	アドレス
79	86	設定可能	TIM20_TRG_COM/ TIM20_DIR/TIM20_IDX	TIM20 トリガおよび転流割込み／ TIM20 方向変更割込み／TIM20 インデックス	0x0000 017C
80	87	設定可能	TIM20_CC	TIM20 キャプチャ /比較割込み	0x0000 0180
81	88	設定可能	FPU	浮動小数点割込み	0x0000 0184
82	89	設定可能	I2C4_EV	I2C4 イベント割込みおよび EXTI ライン 42	0x0000_0188
83	90	設定可能	I2C4_ER	I2C4 エラー割込み	0x0000_018C
84	91	設定可能	SPI4	SPI4 グローバル割込み	0x0000_0190
85	92	設定可能	AES	AES グローバル割込み	0x0000_0194
86	93	設定可能	FDCAN2_IT0	FDCAN2 割込み 0	0x0000_0198
87	94	設定可能	FDCAN2_IT1	FDCAN2 割込み 1	0x0000_019C
88	95	設定可能	FDCAN3_IT0	FDCAN3 割込み 0	0x0000_01A0
89	96	設定可能	FDCAN3_IT1	FDCAN3 割込み 0	0x0000_01A4
90	97	設定可能	RNG	RNG グローバル割込み	0x0000_01A8
91	98	設定可能	LPUART	LPUART グローバル割込み	0x0000_01AC
92	99	設定可能	I2C3_EV	I2C3 イベントおよび EXTI ライン 27 割込み	0x0000_01B0
93	100	設定可能	I2C3_ER	I2C3 エラー割込み	0x0000_01B4
94	101	設定可能	DMAMUX_OVR	DMAMUX オーバーラン割込み	0x0000_01B8
95	102	設定可能	QUADSPI	QUADSPI グローバル割込み	0x0000_01BC
96	103	設定可能	DMA1_CH8	DMA1 チャネル 8 割込み	0x0000_01C0
97	104	設定可能	DMA2_CH6	DMA2 チャネル 6 割込み	0x0000_01C4
98	105	設定可能	DMA2_CH7	DMA2 チャネル 7 割込み	0x0000_01C8
99	106	設定可能	DMA2_CH8	DMA2 チャネル 8 割込み	0x0000_01CC
100	107	設定可能	Cordic	Cordic 割込み	0x0000_01D0
101	108	設定可能	FMAC	FMAC 割込み	0x0000_01D4

15 拡張割込み／イベント･コントローラ（EXTI）

15.1 概要

EXTI の主な機能は以下のとおりです。

- 最大 42 のイベント／割込みリクエストの生成
  - 28 本の設定可能なライン
  - 14 本のダイレクトライン
- 各イベント／割込みラインの個別マスク
- 設定可能な立ち上がりまたは立ち下がりエッジ（設定可能なラインのみ）
- 専用のステータスビット（設定可能なラインのみ）
- イベント／割込みリクエストのエミュレーション（設定可能なラインのみ）

15.2 EXTI の主な機能

拡張割込み／イベント･コントローラ（EXTI）は、外部および内部で非同期的に発生するイベント／割込みを管理し、CPU／割込みコントローラへのイベント･リクエスト、およびパワー･コントローラへのウェイクアップ･リクエストを生成します。

EXTI を使用することで、STOP モードからウェイクアップできるイベントラインを最大 42 本まで管理することが可能です。

設定可能なラインとダイレクトラインがあります。

- ラインが設定可能な場合、アクティブエッジは個別に選択することができ、ステータスフラグは割込みのソースを示します。設定可能なラインを使用するのは、外部 I/O 割込みと、いくつかのペリフェラルです。
- ダイレクトなラインの場合、それらを使用するのは、STOP イベントや割込みからのウェイクアップを生成するいくつかのペリフェラルです。ステータスフラグはペリフェラルから供給されます。

割込みやイベントの生成のために、各ラインを個別にマスクすることができます。

このコントローラは、ソフトウェアによる多重化されたハードウェアイベントラインへの専用レジスタの書込みで、割込みやイベントのエミュレーションもまた可能です。

15.3 EXTI の機能詳細

設定可能な割込みラインの場合、割込みを生成するためには、それらを設定し、有効化する必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、割込みマスクレジスタの対応するビットに"1"を書き込むことによって割込みリクエストを有効にします。選択されたエッジが外部割込みラインで発生すると、割込みリクエストが生成されます。割込みラインに対応するペンディングビットもセットされます。このリクエストは、ペンディングレジスタに"1"を書き込むことによってリセットされます。

ダイレクト割込みラインの場合、割込みは割込みマスクレジスタのデフォルト値によって有効化され、ペンディングレジスタには対応するペンディングビットはセットされていません。

イベントを生成するには、イベントラインを設定し、有効にする必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、イベントマスクレジスタの対応するビットに"1"を書き込むことによってイベントリクエストを有効にします。選択されたエッジがイベントライン

で発生すると、イベントパルスが生成されます。イベントラインに対応するペンディングビットはセットされません。

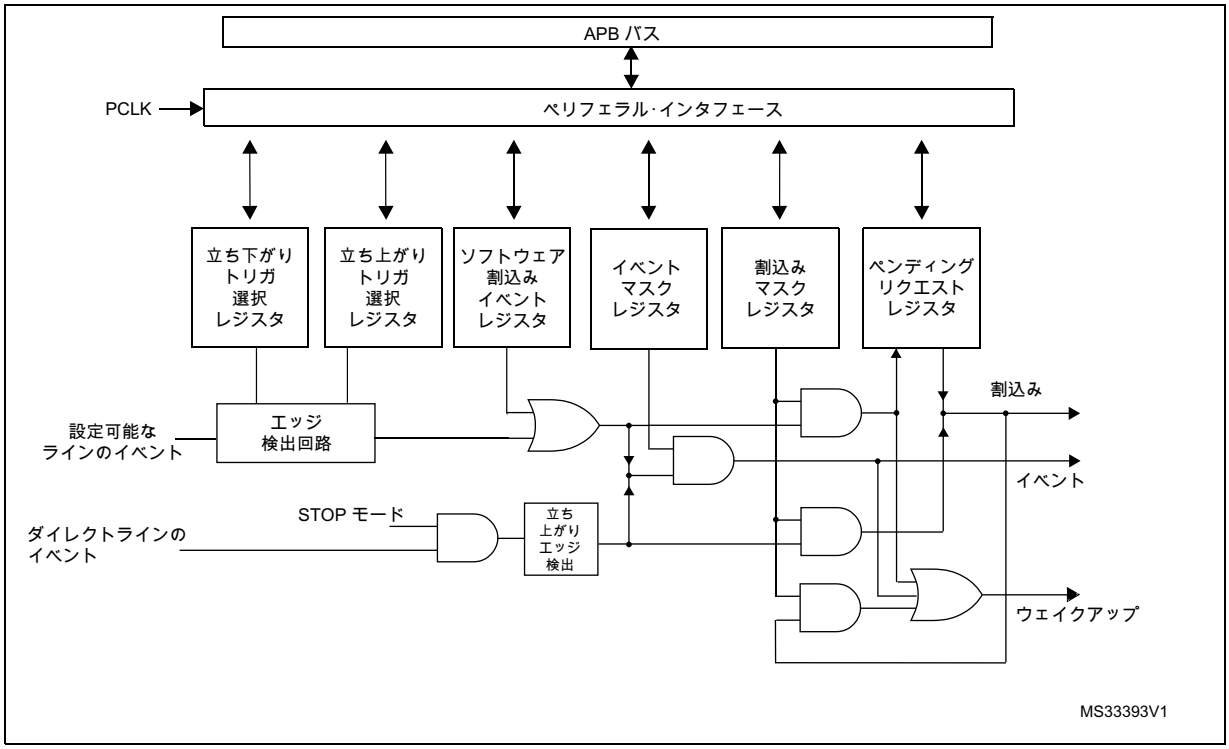
設定可能なラインの場合、割込み／イベントリクエストは、ソフトウェア割込み／イベントレジスタに"1"を書き込むことによって、ソフトウェアによって生成することもできます。

注：　ダイレクトラインに関連する割込みやイベントは、システムが STOP モードのときのみトリガされます。システムがまだ動作している場合は、EXTI は割込み／イベントを生成しません。

15.3.1 EXTI ブロック図

図 35 に拡張割込み／イベント･ブロック図を示します。

図 35. 設定可能な割込み／イベントブロック図



15.3.2 ウェイクアップイベント管理

STM32G4 シリーズ は、コアをウェイクアップするために、外部イベントや内部イベントを処理することができます（WFE）。ウェイクアップイベントは、次のいずれかによって生成できます。

- ペリフェラル制御レジスタで割込みを有効にし、NVIC では有効にせず、Cortex™-M4 システム制御レジスタの SEVONPEND ビットを有効にします。MCU が WFE からリスタートするときには、EXTI ペリフェラル割込みペンディングビットと（NVIC 割込みクリアペンディングレジスタの）ペリフェラル NVIC IRQ チャネルペンディングビットをクリアする必要があります。
- あるいは、EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、ペリフェラル割込みペンディングビットや NVIC IRQ チャネルペンディングビットをクリアする必要はありません。

15.3.3 ペリフェラル非同期割込み

一部のペリフェラルは、システムが RUN モードにあるときにイベントを生成することができ、STOP モードにあるときもシステムを STOP モードからウェイクアップさせることが可能になります。

そのためには、ペリフェラルは、（APB クロックなどのシステムクロックに）同期されたイベントと非同期のイベントの両方を生成します。この非同期イベントは、EXTI のダイレクトラインに接続されます。

注： **STOP モードからのウェイクアップ機能を持つ数少ないペリフェラルは、EXTI の設定可能なラインに接続されます。この場合、STOP モードからのウェイクアップを可能にするには EXTI の設定が必要です。**

15.3.4 ハードウェア割込みの選択

ラインを割込みソースとして設定するには、次の手順を使用します。

1. EXTI_IMR レジスタの対応するマスクビットを設定します。
2. 割込みラインのトリガ選択ビットを設定します（EXTI_RTSR および EXTI_FTSR）。
3. いずれかの EXTI ラインからの割込みを正しく処理できるように、EXTI にマップされた NVIC IRQ チャネルを制御するイネーブルビットとマスクビットを設定します。

注： **ダイレクトラインには EXTI の設定は必要ありません。**

15.3.5 ハードウェアイベントの選択

ラインを割込みソースとして設定するには、次の手順を使用します。

1. EXTI_EMR レジスタの対応するマスクビットを設定します。
2. イベントラインのトリガ選択ビットを設定します（EXTI_RTSR および EXTI_FTSR）。

15.3.6 ソフトウェア割込み／イベントの選択

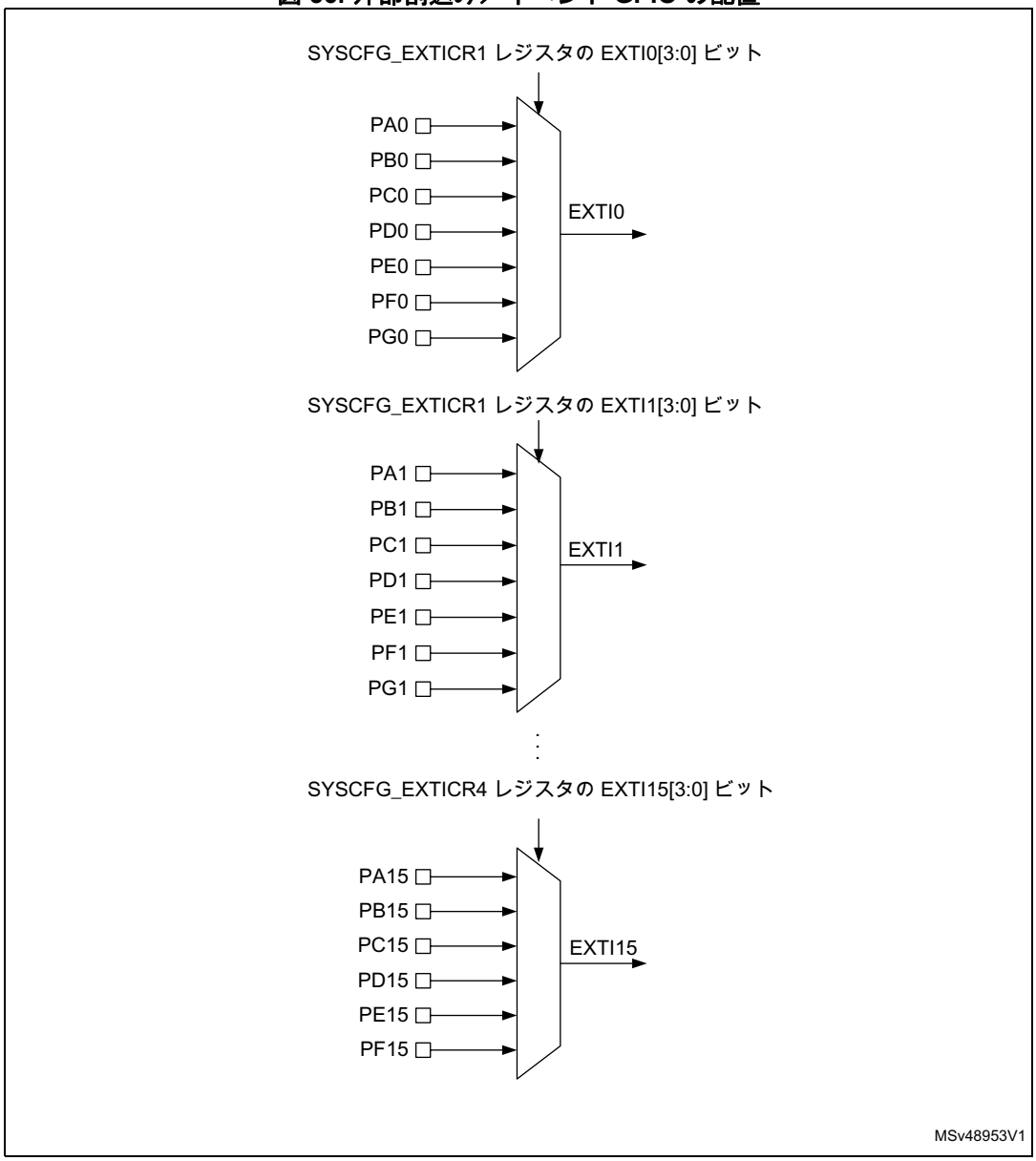
設定可能なラインのいずれかを、ソフトウェア割込み／イベントラインとして設定できます。ソフトウェア割込みを生成する手順は次のとおりです。

1. 対応するマスクビット（EXTI_IMR、EXTI_EMR）を設定します。
2. ソフトウェア割込みレジスタの必要なビットをセットします（EXTI_SWIER）。

15.4 EXTI 割込み／イベントラインの配置

STM32G4 シリーズでは、42 本の割込み／イベントラインが使用できます。GPIO は、16 本の設定可能な割込み／イベントラインに接続されます（図 36を参照してください）。

図 36. 外部割込み／イベント GPIO の配置



EXTI ラインは表 98：EXTI のラインの接続に示すように接続されています。

表 98. EXTI のラインの接続

EXTI ライン	ラインソース	ラインタイプ
0～15	GPIO	設定可能
16	PVD	設定可能
17	RTC アラームイベント	設定可能
18	USB デバイス FS ウェイクアップイベント	ダイレクト
19	タイムスタンプまたはCSS_LSE	設定可能
20	RTC ウェイクアップタイマ	設定可能
21	COMP1 出力	設定可能
22	COMP2 出力	設定可能
23	I2C1 ウェイクアップ	ダイレクト
24	I2C2 ウェイクアップ	ダイレクト
25	USART1 ウェイクアップ	ダイレクト
26	USART2 ウェイクアップ	ダイレクト
27	I2C3 ウェイクアップ	ダイレクト
28	USART3 ウェイクアップ	ダイレクト
29	COMP3 出力	設定可能
30	COMP4 出力	設定可能
31	COMP5 出力	設定可能
32	COMP6 出力	設定可能
33	COMP7 出力	設定可能
34	UART4 ウェイクアップ	ダイレクト
35	UART5 ウェイクアップ	ダイレクト
36	LPUART1 ウェイクアップ	ダイレクト
37	LPTIM1 ウェイクアップ	ダイレクト
40	PVM1 ウェイクアップ	設定可能
41	PVM2 ウェイクアップ	設定可能
42	I2C4 ウェイクアップ	ダイレクト
43	UCPD1	ダイレクト

15.5 EXTI レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル･レジスタには、ワード（32 ビット）単位でアクセスする必要があります。

15.5.1 割込みマスクレジスタ 1（EXTI_IMR1）

アドレス･オフセット：0x00

リセット値：ダイレクト･ラインは '1'、その他のラインは '0' に設定されます。表 98 : EXTI のラインの接続を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM31	IM30	IM29	IM28	IM27	IM26	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IMx**：ライン x の割込みマスク（x = 31～0）
0：ライン x からの割込みリクエストはマスクされます。
1：ライン x からの割込みリクエストはマスクされません。

注： **デフォルトで割込みを有効にするために、ダイレクトラインのリセット値は「1」にセットされます。**

15.5.2 イベントマスクレジスタ 1（EXTI_EMR1）

アドレス･オフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EM31	EM30	EM29	EM28	EM27	EM26	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **EMx**：ライン x のイベントマスク（x = 31～0）
0：ライン x からのイベント･リクエストはマスクされます。
1：ライン x からのイベント･リクエストはマスクされません。

15.5.3 立ち上がりトリガ選択レジスタ 1（EXTI_RTSR1）

アドレス･オフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RT31	RT30	RT29	Res.	Res.	Res.	Res.	Res.	Res.	RT22	RT21	RT20	RT19	Res.	RT17	RT16
rw	rw	rw							rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **RTx：**ライン x の立ち上がりトリガイベント設定ビット（x = 31～29）
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

ビット 28:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **RTx：**ライン x の立ち上がりトリガイベント設定ビット（x = 22～19）
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **RTx：**ライン x の立ち上がりトリガイベント設定ビット（x = 17～0）
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： **設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_RTSR レジスタへの書込み中に設定可能な割込みラインで立ち上がりエッジが発生した場合、ペンディングビットはセットされません。**

同じ割込みラインに対して、立ち上がりエッジ･トリガと立ち下がりエッジ･トリガを設定できます。この場合、両方のエッジでトリガ条件が生成されます。

15.5.4 立ち下がりトリガ選択レジスタ 1（EXTI_FTSR1）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FT31	FT30	FT29	Res.	Res.	Res.	Res.	Res.	Res.	FT22	FT21	FT20	FT19	Res.	FT17	FT16
rw	rw	rw							rw	rw	rw	rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 31～29）
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です

ビット 28:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 22～19）
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 17～0）
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です

注： 設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_FTSR レジスタへの書込み中に設定可能な割込みラインで立ち下がりエッジが発生した場合、ペンディングビットはセットされません。

同じ割込みラインに対して、立ち上がりエッジ･トリガと立ち下がりエッジ･トリガを設定できます。この場合、両方のエッジでトリガ条件が生成されます。

15.5.5 ソフトウェア割込みイベントレジスタ 1（EXTI_SWIER1）

アドレス･オフセット：0x10

リセット値：0x0000 0000

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

ビット 31:29 **SWIx：**ライン x のソフトウェア割込み（x = 31～29）
割込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割込みリクエストが生成されます。
EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

ビット 28:23 予約済みであり、リセット値に保持する必要があります。

ビット 22：19 **SWIx：**ライン x のソフトウェア割込み（x = 22～19）
割込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割込みリクエストが生成されます。
EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **SWIx：**ライン x のソフトウェア割込み（x = 17～0）
割込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割込みリクエストが生成されます。
EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

15.5.6 ペンディングレジスタ 1（EXTI_PR1）

アドレス･オフセット：0x14

リセット値：定義されていません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PIF31	PIF30	PIF29	Res.	Res.	Res.	Res.	Res.	Res.	PIF22	PIF21	PIF20	PIF19	Res.	PIF17	PIF16
rc_w1	rc_w1	rc_w1							rc_w1	rc_w1	rc_w1	rc_w1		rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:29 **PIFx**：ライン x のペンディング割込みフラグ（x = 31～29）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

ビット 28:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **PIFx**：ライン x のペンディング割込みフラグ（x = 22～19）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **PIFx**：ライン x のペンディング割込みフラグ（x = 17～0）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

15.5.7 割込みマスクレジスタ 2（EXTI_IMR2）

アドレス･オフセット：0x20

リセット値：ダイレクト･ラインは '1'、その他のラインは '0' に設定されます。表 98 : EXTI のラインの接続を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	IM43	IM42	IM41	IM40	Res.	Res.	IM37	IM36	IM35	IM34	IM33	IM32
				rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **IMx**：ライン x の割込みマスク（x = 43～40）
0：ライン x からの割込みリクエストはマスクされます。
1：ライン x からの割込みリクエストはマスクされません。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **IMx**：ライン x の割込みマスク（x = 37～32）
0：ライン x からの割込みリクエストはマスクされます。
1：ライン x からの割込みリクエストはマスクされません。

注： **デフォルトで割込みを有効にするために、ダイレクトラインのリセット値は「1」にセットされます。**

15.5.8 イベントマスクレジスタ 2（EXTI_EMR2）

アドレス･オフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	EM43	EM42	EM41	EM40	Res.	Res.	EM37	EM36	EM35	EM34	EM33	EM32
				rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **EMx**：ライン x のイベントマスク（x = 43～40）
0：ライン x からのイベント･リクエストはマスクされます。
1：ライン x からのイベント･リクエストはマスクされません。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **EMx**：ライン x のイベントマスク（x = 37～32）
0：ライン x からのイベント･リクエストはマスクされます。
1：ライン x からのイベント･リクエストはマスクされません。

15.5.9 立ち上がりトリガ選択レジスタ 2（EXTI_RTSR2）

アドレス･オフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	RT41	RT40	Res.	Res.	Res.	Res.	Res.	Res.	RT33	RT32
						rw	rw							rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **RTx**：ライン x の立ち上がりトリガイベント設定ビット（x = 40～41）
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **RTx**：ライン x の立ち上がりトリガイベント設定ビット（x = 32～33）
0：入力ラインの（イベントと割込みについて）立ち上がりトリガは無効です。
1：入力ラインの（イベントと割込みについて）立ち上がりトリガは有効です。

注： **設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_RTSR レジスタへの書込み中に設定可能な割込みラインで立ち上がりエッジが発生した場合、ペンディングビットはセットされません。**

同じ割込みラインに対して、立ち上がりエッジ･トリガと立ち下がりエッジ･トリガを設定できます。この場合、両方のエッジでトリガ条件が生成されます。

15.5.10 立ち下がりトリガ選択レジスタ 2（EXTI_FTSR2）

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	FT41	FT40	Res.	Res.	Res.	Res.	Res.	Res.	FT33	FT32
						rw	rw							rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 40～41）
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **FTx：**ライン x の立ち下がりトリガイベント設定ビット（x = 32～33）
0：入力ラインの（イベントと割込みについて）立ち下がりトリガは無効です
1：入力ラインの（イベントと割込みについて）立ち下がりトリガは有効です

注： 設定可能なウェイクアップラインはエッジトリガされています。これらのライン上でグリッチが生成されないようにする必要があります。EXTI_FTSR レジスタへの書込み中に設定可能な割込みラインで立ち下がりエッジが発生した場合、ペンディングビットはセットされません。

同じ割込みラインに対して、立ち上がりエッジ･トリガと立ち下がりエッジ･トリガを設定できます。この場合、両方のエッジでトリガ条件が生成されます。

15.5.11 ソフトウェア割込みイベントレジスタ 2（EXTI_SWIER2）

アドレス･オフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	SWI 41	SWI 40	Res.	Res.	Res.	Res.	Res.	Res.	SWI 33	SWI 32
						rw	rw							rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **SWIx：**ライン x のソフトウェア割込み（x = 40～41）
割込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割込みリクエストが生成されます。
EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **SWIx：**ライン x のソフトウェア割込み（x = 32～33）

割込みが EXTI_IMR のこのライン上で有効になっている場合、このビットが“0”のときに“1”を書き込むと EXTI_PR の対応するペンディングビットがセットされるため、割込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする（対応ビットに“1”を書き込む）と、このビットはクリアされます。

15.5.12 ペンディングレジスタ 2（EXTI_PR2）

アドレス･オフセット：0x34

リセット値：定義されていません。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	PIF41	PIF40	Res.	Res.	Res.	Res.	Res.	Res.	PIF33	PIF32
						rc_w1	rc_w1							rc_w1	rc_w1

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **PIFx：**ライン x のペンディング割込みフラグ（x = 40～41）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **PIFx：**ライン x のペンディング割込みフラグ（x = 32～33）

0：トリガリクエストは発生していません。
1：選択されたトリガリクエストが発生しました。
このビットは、選択されたエッジイベントが割込みラインに到着したときにセットされます。このビットは、ビットに“1”を書き込むことによってクリアされます。

15.5.13 EXTI レジスタマップ

表 99 に、EXTI レジスタ･マップとリセット値を示します。

表 99. 拡張割込み／イベントコントローラのレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	EXTI_IMR1	IM31	IM30	IM29	IM28	IM27	IM26	IM25	IM24	IM23	IM22	IM21	IM20	IM19	IM18	IM17	IM16	IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
	リセット値	1	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	EXTI_EMR1	EM31	EM30	EM29	EM28	EM27	EM26	EM25	EM24	EM23	EM22	EM21	EM20	EM19	EM18	EM17	EM16	EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	EXTI_RTSR1	RT31	RT30	RT29	Res.	Res.	Res.	Res.	Res.	Res.	RT22	RT21	RT20	RT19	Res.	RT17	RT16	RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
	リセット値	0	0	0							0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	EXTI_FTSR1	FT31	FT30	FT29	Res.	Res.	Res.	Res.	Res.	Res.	FT22	FT21	FT20	FT19	Res.	FT17	FT16	FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
	リセット値	0	0	0							0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	EXTI_SWIER1	SW312	SWI30	SWI29	Res.	Res.	Res.	Res.	Res.	Res.	SWI22	SWI21	SWI20	SWI19	Res.	SWI17	SWI16	SWI15	SWI14	SWI13	SWI12	SWI11	SWI10	SWI9	SWI8	SWI7	SWI6	SWI5	SWI4	SWI3	SWI2	SWI1	SWI0
	リセット値	0	0	0							0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	EXTI_PR1	PIF31	PIF30	PIF29	Res.	Res.	Res.	Res.	Res.	Res.	PIF22	PIF21	PIF20	PIF19	Res.	PIF17	PIF16	PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
	リセット値	0	0	0							0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	EXTI_IMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IM43	IM42	IM41	IM40	Res.	Res.	IM37	IM36	IM35	IM34	IM33	IM32
	リセット値																					1	1	0	1			0	0	0	1	1	1
0x24	EXTI_EMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EM43	EM42	EM41	EM40	Res.	Res.	EM37	EM36	EM35	EM34	EM33	EM32
	リセット値																					0	0	0	0			0	0	0	0	0	0
0x28	EXTI_RTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RT41	RT40	Res.	Res.	Res.	Res.	Res.	Res.	RT33	RT32
	リセット値																							0	0							0	0
0x2C	EXTI_FTSR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FT41	FT40	Res.	Res.	Res.	Res.	Res.	Res.	FT33	FT32
	リセット値																							0	0							0	0
0x30	EXTI_SWIER2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWI41	SWI40	Res.	Res.	Res.	Res.	Res.	Res.	SWI33	SWI32
	リセット値																							0	0							0	0
0x34	EXTI_PR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PIF41	PIF40	Res.	Res.	Res.	Res.	Res.	Res.	PIF33	PIF32
	リセット値																							0	0							0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

16 巡回冗長検査計算ユニット（CRC）

16.1 概要

CRC（Cyclic Redundancy Check：巡回冗長検査）計算ユニットは、8、16、または 32 ビットデータワードと、ある生成多項式から、CRC コードを得るために使用されます。

他のアプリケーションの中でも、CRC ベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがFlashメモリの整合性を確認するひとつの手段となっています。CRC 計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたリファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

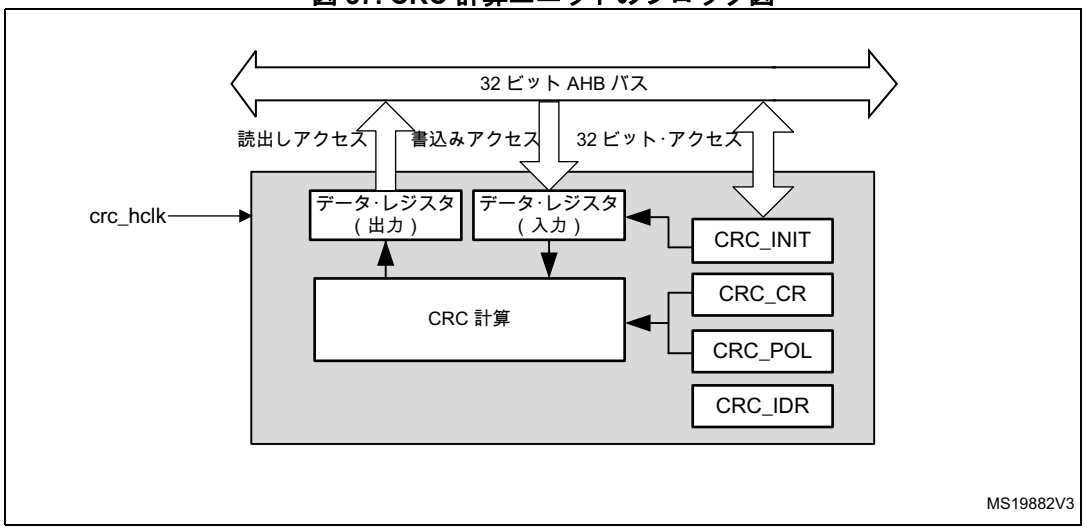
16.2 CRC の主な機能

- 以下の CRC-32（Ethernet）多項式を使用します。0x4C11DB7
  $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- あるいは、完全にプログラム可能な多項式をプログラム可能なサイズ（7、8、16、32 ビット）で使用します。
- サイズが 8、16、32 ビットのデータを取り扱います。
- プログラム可能な CRC の初期値
- シングル入力／出力 32 ビットデータレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが 32 ビットの場合、CRC の計算は AHB クロック 4 サイクル（HCLK）以内に行われます。
- 汎用 8 ビットレジスタ（一時ストレージとして使用可能）
- I/O データの可逆性のオプション
- AHB スレーブ･ペリフェラルからは 32 ビット･ワード単位のアクセスのみが可能。ただし、CRC_DR レジスタに対しては、ワード単位、右詰めのハーフ･ワード単位、右詰めのバイト単位でアクセスできます。

16.3 CRC の機能説明

16.3.1 CRC ブロック図

図 37. CRC 計算ユニットのブロック図



16.3.2 CRC 内部信号

表 100. CRC 内部入力／出力信号

信号名	信号タイプ	説明
crc_hclk	デジタル入力	AHB クロック

16.3.3 CRC 操作

CRC 計算ユニットは、1 つの 32 ビット·リード／ライト·データ·レジスタ（CRC_DR）を持っています。このレジスタを使用して、新しいデータを入力し（書込みアクセス）、前回の CRC 計算結果を保持します（読出しアクセス）。

データレジスタへの書込み操作のたびに、前回の CRC 値（CRC_DR に格納）と新しい値の組み合わせが作成されます。CRC 計算は、書き込まれるデータのフォーマットに応じ、32 ビットデータワード全体に対して、またはバイト単位で行われます。

CRC_DR レジスタは、ワード、右詰めのハーフワード、右詰めのバイトによってアクセスできます。他のレジスタについては、32 ビットアクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32 ビットの場合、4 AHB クロック·サイクル
- 16 ビットの場合、2 AHB クロック·サイクル
- 8 ビットの場合、1 AHB クロック·サイクル

入力バッファを使うと、前回の CRC 計算によるウェイトステートを待つことなく、すぐに第 2 のデータを書き込むことができます。

特定のバイト数に対する書込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5 バイトの CRC は、1 ワードの書込みと、それに続く 1 バイトの書込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CR レジスタの REV_IN[1:0] ビットに応じて、8、16、および 32 ビットで行うことができます。

たとえば、入力データ 0x1A2B3C4D は、CRC 計算では以下のように使用されます。

- 0x58D43CB2（バイト単位でビットが逆転）
- 0xD458B23C（ハーフワード単位でビットが逆転）
- 0xB23CD458（フルワードでビットが逆転）

また、出力データも、CRC_CR レジスタの REV_OUT ビットをセットすることによって逆にすることができます。

操作はビットレベルで行われます。たとえば、出力データ 0x11223344 は 0x22CC4488 に変換されます。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は 0xFFFFFFFF）。

CRC の初期値は、CRC_INIT レジスタでプログラムすることができます。CRC_DR レジスタは、CRC_INIT レジスタの書込みアクセス時に自動的に初期化されます。

CRC_IDR レジスタを使用して、CRC 計算に関する一時的な値を保持することができます。このレジスタは、CRC_CR レジスタの RESET ビットによる影響を受けません。

多項式のプログラマビリティ

多項式係数は CRC_POL レジスタを介して完全にプログラム可能であり、多項式のサイズは CRC_CR レジスタの POLYSIZE[1:0] ビットをプログラムすることにより、7、8、16、または 32 ビットに設定できます。偶多項式はサポートされていません。

注： **偶多項式とは $X+X^2+..+X^n$、奇多項式とは $1+X+X^2+..+X^n$ のような種類の多項式です。**

CRC データが 32 ビット未満であれば、その値は CRC_DR レジスタの最下位ビットから読み出すことができます。

信頼できる CRC 計算結果を得るために、CRC 計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC 計算が進行中である場合、アプリケーションは計算をリセットするか、または CRC_DR の読出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32 （Ethernet）多項式、0x4C11DB7.

16.4 CRC レジスタ

CRC_DR レジスタには、ワード単位、右詰めのハーフワード単位、右詰めのバイト単位でアクセスできます。他のレジスタに対しては、32 ビット単位のアクセスのみが可能です。

16.4.1 CRC データ･レジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

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

ビット 31:0 **DR[31:0]**：データレジスタビット

このレジスタを使用して、CRC 計算機に新しいデータを書き込みます。
読出し時には、前回の CRC 計算結果を保持します。
データサイズが 32 ビット未満であれば、下位ビットを使用して正しい値の書込み／読出しを行います。

16.4.2 CRC 独立型データレジスタ (CRC_IDR)

アドレスオフセット：0x04

リセット値：0x0000 0000

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

ビット 31:0 **IDR[31:0]**：汎用 32 ビットデータレジスタビット

これらのビットは、4 バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CR レジスタの RESET ビットによって生成される CRC リセットの影響を受けません。

16.4.3 CRC制御レジスタ（CRC_CR）

アドレスオフセット：0x08

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **REV_OUT**：出力データを逆にします。

このビットは、出力データビット順序の反転を制御します。

0：ビットの順序は変わりません。

1：出力フォーマットのビットが反転しています。

ビット 6:5 **REV_IN[1:0]**：入力データを逆にします。

これらのビットは、入力データビット順序の反転を制御します。

00：ビットの順序は変わりません。

01：バイト単位でビットが反転しています。

10：ハーフワード単位でビットが反転しています。

11：ワード単位でビットが反転しています。

ビット 4:3 **POLYSIZE[1:0]**：多項式のサイズ

これらのビットは、多項式のサイズを制御します。

00：32 ビットの多項式

01：16 ビットの多項式

10：8 ビットの多項式

11：7 ビットの多項式

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **RESET**：RESET ビット

このビットは、CRC 計算ユニットをリセットし、CRC_INIT レジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

16.4.4 CRC の初期値（CRC_INIT）

アドレスオフセット：0x10

リセット値：0xFFFF FFFF

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

ビット 31:0 **CRC_INIT[31:0]** *:* プログラム可能な CRC の初期値

このレジスタを使用して、CRC の初期値を書き込みます。

16.4.5 CRC 多項式（CRC_POL）

アドレスオフセット：0x14

リセット値：0x04C1 1DB7

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

ビット 31:0 **POL[31:0]** *:* プログラム可能な多項式

このレジスタを使用して、CRC 計算に使用される多項式の係数を書き込みます。

多項式のサイズが 32ビット未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

16.4.6 CRC レジスタマップ

表 101. CRC レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CRC_DR	DR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x04	CRC_IDR	IDR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	CRC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REV_OUT	REV_IN[1:0]		POLYSIZE[1:0]		Res.	Res.	RESET
	リセット値																									0	0	0	0	0			0
0x10	CRC_INIT	CRC_INIT[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	CRC_POL	POL[31:0]																															
	リセット値	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1	1	1

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

17 CORDIC コプロセッサ（CORDIC）

17.1 CORDIC の概要

CORDIC コプロセッサは、モータ制御、計量、信号処理をはじめとする多くのアプリケーションに共通して使用する数学関数（主として三角関数）の演算をハードウェアで高速化します。

ソフトウェア実装に比べてこれらの関数の演算が高速になり、より低い動作周波数を使用することや、プロセッサ·サイクルを解放して他のタスクを実行することなどができるようになります。

17.2 CORDIC の主な機能

- 24bit CORDIC ローテーション·エンジン
- 円モードと双曲線モード
- 回転モードとベクタリングモード
- 関数：sine、cosine、sinh、cosh、atan、atan2、atanh、modulus、平方根、自然対数
- プログラム可能な精度
- 低遅延 AHB スレーブインタフェース
- 結果はポーリングや割込みを行わずに準備が出来次第読出し可能
- DMA 読出し／書込みチャネル
- DMA による複数レジスタの読出し／書込み

17.3 CORDIC の機能詳細

17.3.1 概要

CORDIC は、三角関数と双曲線関数の値を求めるための低コストな逐似近似アルゴリズムです。

三角関数（円）モードでは、ある角度 θ の正弦と余弦は、回転角の累積和が入力角 θ に等しくなるまで単位ベクトル [1, 0] を回転させることで求められます。回転したベクトルのデカルト座標成分 x と y が、それぞれ θ の余弦と正弦に相当します。反対に、逆正接（y/x）に相当するベクトル [x, y] の角度は、単位ベクトル [1, 0] が得られるまで角度を連続的に減少させながら [x, y] を回転させることにより求められます。回転角の累積和から元ベクトルの角度が得られます。

CORDICアルゴリズムは、連続した円回転を双曲線に沿ったステップに置き換えることにより、双曲線関数（sinh、cosh、atanh）の計算にも使用できます。

その他の関数は、上記の基本関数から導出可能です。

17.3.2 CORDIC 関数

コプロセッサを使用する際の最初のステップは、CORDIC_CR レジスタの FUNC フィールドをしかるべくプログラムすることにより、必要な関数を選択することです。

表 102 に、CORDIC コプロセッサがサポートする関数の一覧を示します。

表 102. CORDIC 関数

機能	第 1 引数（ARG1）	第 2 引数（ARG2）	第 1 結果（RES1）	第 2 結果（RES2）
余弦	角度 θ	モジュラス m	m· cosθ	m· si nθ
正弦	角度 θ	モジュラス m	m· si nθ	m· cosθ
位相	x	y	atan2(y,x)	$\sqrt{x^2+y^2}$
モジュラス	x	y	$\sqrt{x^2+y^2}$	atan2(y,x)
逆正接	x	なし	$\tan^{-1}$ x	なし
双曲線余弦	x	なし	cosh x	sinh x
双曲線正弦	x	なし	sinh x	cosh x
双曲線逆正接	x	なし	$\tanh^{-1}$ x	なし
自然対数	x	なし	ln x	なし
平方根	x	なし	$\sqrt{x}$	なし

関数によっては、2 つの入力引数 ARG1 と ARG2 を取り、同時に 2 つの結果 RES1 と RES2 を生成するものもあります。これはアルゴリズムの副次的な効果であり、2 つの値を得るために必要な演算が 1 回ですむことを意味します。これは、たとえば極座標を直交座標に変換する場合です。sin θ によって cos θ も生成され、cos θ によって sin θ も生成されます。直交座標の極座標変換（phase(x,y)、modulus(x,y)）や、双曲線関数（cosh θ、sinh θ）も同様です。

注： **指数関数 exp x は、sinh x と cosh x の和として求められます。また、N を底とする対数 $\log_N$ x は、ln x に定数 K（ここで K = 1/ln N）を乗じることにより導出できます。**

特定の関数（atan、log、sqrt）では、スケーリングファクタ（セクション 17.3.4 : 参照）を適用することにより、q1.31 固定小数点が対応している最大値 [-1, 1] を超えて関数の範囲を拡大できます。スケーリングファクタは、その他すべての円関数では 0 に、双曲線関数では 1 にセットする必要があります。

余弦

表 103. 余弦パラメータ

パラメータ	説明	範囲
ARG1	ラジアン単位の角度 θ を π で除算した値	[-1, 1]
ARG2	ベクトルの絶対値 m	[0, 1]
RES1	m · cos θ	[-1, 1]
RES2	m · sin θ	[-1, 1]
SCALE	適用外	0

この関数は、-π から π の範囲の角度の余弦を計算します。極座標から直交座標への変換にも使用できます。

第 1 引数は、ラジアン単位の角度 θ です。ARG1 をプログラムする前に、π で除算する必要があります。

第 2 引数 m は、ベクトルの絶対値 m です。m が 1 よりも大きい場合は、ソフトウェアでスケーリングを適用して q1.31 で表現できる範囲の ARG2 とする必要があります。

第 1 結果 RES1 は、角度の余弦にベクトルの絶対値を乗算した値です。

第 2 結果 RES2 は、角度の正弦にモジュラスを掛けた値です。

正弦

表 104. 正弦パラメータ

パラメータ	説明	範囲
ARG1	ラジアン単位の角度 θ を π で除算した値	[-1, 1]
ARG2	ベクトルの絶対値 m	[0, 1]
RES1	m · sin θ	[-1, 1]
RES2	m · cos θ	[-1, 1]
SCALE	適用外	0

この関数は、-π から π の範囲の角度の正弦を計算します。極座標から直交座標への変換にも使用できます。

第 1 引数は、ラジアン単位の角度 θ です。ARG1 をプログラムする前に、π で除算する必要があります。

第 2 引数 m は、ベクトルの絶対値 m です。m が 1 よりも大きい場合は、ソフトウェアでスケーリングを適用して q1.31 で表現できる範囲の ARG2 とする必要があります。

第 1 結果 RES1 は、角度の正弦にベクトルの絶対値を乗算した値です。

第 2 結果 RES2 は、角度の余弦にベクトルの絶対値を乗算した値です。

位相

表 105. 位相パラメータ

パラメータ	説明	範囲
ARG1	x 座標	[-1, 1]
ARG2	y 座標	[-1, 1]
RES1	ラジアン単位の位相角 θ を π で除算した値	[-1, 1]
RES2	ベクトルの絶対値 m	[0, 1]
SCALE	適用外	0

この関数は、ベクトル **v** = [x y] の -π から π の範囲にある位相角を計算します（atan2(y,x) とも言います）。直交座標から極座標への変換にも使用できます。

第 1 引数は x 座標、すなわち、ベクトルの x 軸方向の大きさです。|x| > 1 の場合には、ソフトウェアでスケーリングを適用して q1.31 範囲の ARG1 とする必要があります。

第 2 引数は y 座標、すなわち、ベクトルの y 軸方向の大きさです。|y| > 1 の場合には、ソフトウェアでスケーリングを適用して q1.31 範囲の ARG2 とする必要があります。

第 1 結果 RES1 は、ベクトル **v** の位相角 θ です。ラジアン単位で角度を得るには、RES1 に π を乗算する必要があります。π に近い値は、位相角の円の性質のために -π に折り返す場合があることに注意してください。

第 2 結果 RES2 は、v=SQRT(x^2+y^2)の式によって与えられるモジュラスです。$|v| = \sqrt{x^2 + y^2}$ 。|**v**| > 1 の場合、RES2 の結果は 1 で飽和します。

ベクトルの絶対値

表 106. ベクトル絶対値のパラメータ

パラメータ	説明	範囲
ARG1	x 座標	[-1, 1]
ARG2	y 座標	[-1, 1]
RES1	ベクトルの絶対値 m	[0, 1]
RES2	位相角 θ	[-1, 1]
SCALE	適用外	0

この関数は、ベクトル **v** = [x y] の大きさ（モジュラス）を計算します。直交座標から極座標への変換にも使用できます。

第 1 引数は x 座標、すなわち、ベクトルの x 軸方向の大きさです。|x| > 1 の場合には、ソフトウェアでスケーリングを適用して q1.31 範囲の ARG1 とする必要があります。

第 2 引数は y 座標、すなわち、ベクトルの y 軸方向の大きさです。|y| > 1 の場合には、ソフトウェアでスケーリングを適用して q1.31 範囲の ARG2 とする必要があります。

第 1 結果 RES1 は、v=SQRT(x^2+y^2) の式によって与えられるモジュラスです。$|v| = \sqrt{x^2 + y^2}$ 。|**v**| > 1 の場合、RES1 の結果は 1 で飽和します。

第 2 結果 RES2 は、ベクトル **v** の位相角 θ です。ラジアン単位で角度を得るには、RES2 に π を乗算する必要があります。π に近い値は、位相角の円の性質のために -π に折り返す場合があることに注意してください。

逆正接

表 107. 逆正接パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-1, 1]
ARG2	適用外	-
RES1	ラジアン単位の $2^{-n} \cdot \tan^{-1}$ x を π で除算した値	[-1, 1]
RES2	適用外	-
SCALE	n	[0, 7]

この関数は、入力引数 x の逆正接を計算します。

第 1 引数 ARG1 は、入力値 x = tan θ です。|x| > 1 の場合には、$-1 < x \cdot 2^{-n} < 1$ となるようにソフトウェアでスケーリングファクタ 2^{-n} を適用する必要があります。スケーリングされた値 $x \cdot 2^{-n}$ は ARG1 に、スケールファクタ n を SCALE パラメータにプログラムする必要があります。

許容される最大入力値は tan(θ) = 128 であり、角度にすると θ = 89.55 度に相当します。|x| > 128 の場合には、$\tan^{-1} x$ を計算するためにソフトウェア手法を用いる必要があります。

第 2 引数 ARG2 は使用しません。

第 1 結果 RES1 は、$\theta = \tan^{-1} x$ の角度です。ラジアン単位で角度を得るために、RES1 に $2^n \cdot \pi$ を掛ける必要があります。

第 2 結果 RES2 は使用しません。

双曲線余弦

表 108. 双曲線余弦パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.559 0.559]
ARG2	適用外	-
RES1	$2^{-n} \cdot \cosh x$	[0.5 0.846]
RES2	$2^{-n} \cdot \sinh x$	[-0.683 0.683]
SCALE	n	1

この関数は、双曲角 x の双曲線余弦を計算します。指数関数 $e^x = \cosh x + \sinh x$ と $e^{-x} = \cosh x - \sinh x$ を計算するためにも使用できます。

第 1 引数は、双曲角 x です。-1.118～+1.118 の範囲の x の値のみに対応しています。cosh x の最小値は q1.31 フォーマットの範囲を超える 1 であるため、ソフトウェアでスケーリングファクタ 2^{-n} を適用する必要があります。SCALE パラメータにファクタ n = 1 をプログラムする必要があります。

第 2 引数 は使用しません。

第 1 結果 RES1 は、双曲線余弦 cosh x です。正しい結果を得るために、RES1 に 2 を掛ける必要があります。

第 2 結果 RES2 は、双曲線正弦 sinh x です。正しい結果を得るために、RES2 に 2 を掛ける必要があります。

双曲線正弦

表 109. 双曲線正弦パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.559, 0.559]
ARG2	適用外	-
RES1	$2^{-n} \cdot \sinh x$	[-0.683, 0.683]

表 109. 双曲線正弦パラメータ （続き）

パラメータ	説明	範囲
RES2	$2^{-n} \cdot \cosh x$	[0.5, 0.846]
SCALE	n	1

この関数は、双曲角 x の双曲線正弦を計算します。指数関数 $e^x = \cosh x + \sinh x$ と $e^{-x} = \cosh x - \sinh x$ を計算するためにも使用できます。

第 1 引数は、双曲角 x です。-1.118～+1.118 の範囲の x の値のみに対応しています。すべての入力値に対して、ソフトウェアでスケーリングファクタ 2^{-n}（n = 1）を適用する必要があります。スケーリングされた値 x · 0.5 を ARG1 に、ファクタ n = 1 を SCALE パラメータにプログラムする必要があります。

第 2 引数 は使用しません。

第 1 結果 RES1 は、双曲線正弦 sinh x です。正しい結果を得るために、RES1 に 2 を掛ける必要があります。

第 2 結果 RES2 は、双曲線余弦 cosh x です。正しい結果を得るために、RES2 に 2 を掛ける必要があります。

双曲線逆正接

表 110. 双曲線逆正接パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[-0.403, 0.403]
ARG2	適用外	-
RES1	$2^{-n} \cdot \text{atanh } x$	[-0.559, 0.559]
RES2	適用外	-
SCALE	n	1

この関数は、入力引数 x の双曲線逆正接を計算します。

第 1 引数 ARG1 は、入力値 x です。-0.806～+0.806 の範囲の x の値のみに対応しています。値 x は、ファクタ 2^{-n}（n = 1）でスケーリングする必要があります。
スケーリングした x 値 0.5 を ARG1 パラメータ、ファクタ n = 1 を SCALE パラメータにそれぞれプログラムする必要があります。

第 2 引数 は使用しません。

第 1 結果 RES1 は、双曲線逆正接 atanh x です。正しい値を得るために、RES1 に 2 を掛ける必要があります。

第 2 結果は使用しません。

自然対数

表 111. 自然対数パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[0.054, 0.875]
ARG2	適用外	-
RES1	$2^{-(n+1)}.\ln x$	[-0.279, 0.137]
RES2	適用外	-
SCALE	n	[1, 4]

この関数は、入力引数 x の自然対数を計算します。

第 1 引数 ARG1 は、入力値 x です。0.107～9.35 の範囲の x の値のみに対応しています。値 x は、$x \cdot 2^{-n} < 1\text{-}2^{-n}$ となるように、ファクタ 2^{-n} でスケーリングする必要があります。スケーリングした x 値 2^{-n} を ARG1 パラメータ、ファクタ n を SCALE パラメータにそれぞれプログラムする必要があります。

有効なスケーリング･ファクタ n と、それに対応する角度 x と ARG1 の範囲を表 112 に示します。

表 112. 自然対数のスケーリングファクタと対応範囲

n	x の範囲	ARG1 の範囲
1	$0.107 \le x < 1$	$0.0535 \le ARG1 < 0.5$
2	$1 \le x < 3$	$0.25 \le ARG1 < 0.75$
3	$3 \le x < 7$	$0.375 \le ARG1 < 0.875$
4	$7 \le x \le 9.35$	$0.4375 \le ARG1 < 0.584$

第 2 引数 は使用しません。

第 1 結果は自然対数 ln x です。正しい値を得るために、RES1 に $2^{(n+1)}$ を掛ける必要があります。

第 2 結果は使用しません。

平方根

表 113. 平方根パラメータ

パラメータ	説明	範囲
ARG1	$x \cdot 2^{-n}$	[0.027, 0.875]
ARG2	適用外	-
RES1	$2^{-n}\sqrt{x}$	[0.04 1]
RES2	適用外	-
SCALE	n	[0, 2]

この関数は、入力引数 x の平方根を計算します。

第 1 引数 ARG1 は、入力値 x です。0.027～2.34 の範囲の x の値のみに対応しています。値 x は、$x \cdot 2^{-n} < (1 - 2^{(-n-2)})$ となるように、ファクタ 2^{-n} でスケーリングする必要があります。

スケーリングされた値 $x \cdot 2^{-n}$ は ARG1 に、ファクタ n を SCALE パラメータにプログラムする必要があります。

有効なスケーリングファクタ n と、それに対応する角度 x と ARG1 の範囲を表 114 に示します。

表 114. 平方根のスケーリング･ファクタと対応範囲

n	x の範囲	ARG1 の範囲
0	$0.027 \leq x < 0.75$	$0.027 \leq ARG1 < 0.75$
1	$0.75 \leq x < 1.75$	$0.375 \leq ARG1 < 0.875$
2	$1.75 \leq x \leq 2.341$	$0.4375 \leq ARG1 \leq 0.585$

第 2 引数 は使用しません。

第 1 結果は x の平方根です。正しい値を得るために、RES1 に 2^n を掛ける必要があります。

第 2 結果は使用しません。

17.3.3 固定小数点表現

CORDIC は符号付き固定小数点整数フォーマットで動作します。入出力値は q1.31 と q1.15 のいずれかです。

q1.31 フォーマットでは、数字は 1bit の符合部分と 31bit の小数部分（バイナリ小数位）によって表現されます。したがって、数値範囲は -1（0x80000000）から $1 - 2^{-31}$（0x7FFFFFFF）となります。

q1.15 フォーマットでは、数値範囲は 1（0x8000）から $1 - 2^{-15}$（0x7FFF）となります。このフォーマットには、2 個の入力引数が単一の 32bit 書込みに格納可能で、2 個の結果が 1 回の 32bit 読出しでフェッチ可能であるという利点があります。

17.3.4 スケーリングファクタ

セクション 17.3.2 : に示した関数のいくつかには、スケーリングファクタ SCALE が指定されています。これによって、入力、出力、内部レジスタを飽和させることなく、CORDIC が対応している値の範囲全体を対称として含むように、関数の入力範囲を拡張可能となります。スケーリングファクタが必要な場合には、ソフトウェアで計算して、CORDIC_CSR レジスタの SCALE フィールドにプログラムする必要があります。スケーリングされた値をCORDIC_WDATA レジスタにプログラムする前に、入力引数をそれに応じてスケーリングする必要があります。また、CORDIC_RDATA レジスタから読み出された結果に対して、スケーリングを元に戻す必要があります。

注 : **スケーリングされた値の切り捨てのため、必然的にスケーリングファクタには精度の低下が伴います。**

17.3.5 精度

結果の精度は、CORDIC の繰り返し回数に依存します。このアルゴリズムは、三角関数（正弦、余弦、位相、モジュラス）の繰り返し 1 回あたり 1 バイナリデジットの一定速度で収束します（図 38 参照）。

双曲線関数（双曲線正弦、双曲線余弦、自然対数）では、CORDIC アルゴリズムの特殊性のために収束速度の一定度は劣ります（図 39 参照）。平方根関数は、双曲線関数のおよそ 2 倍の速度で収束します（図 40 参照）。

図 38. 三角関数に対する CORDIC の収束

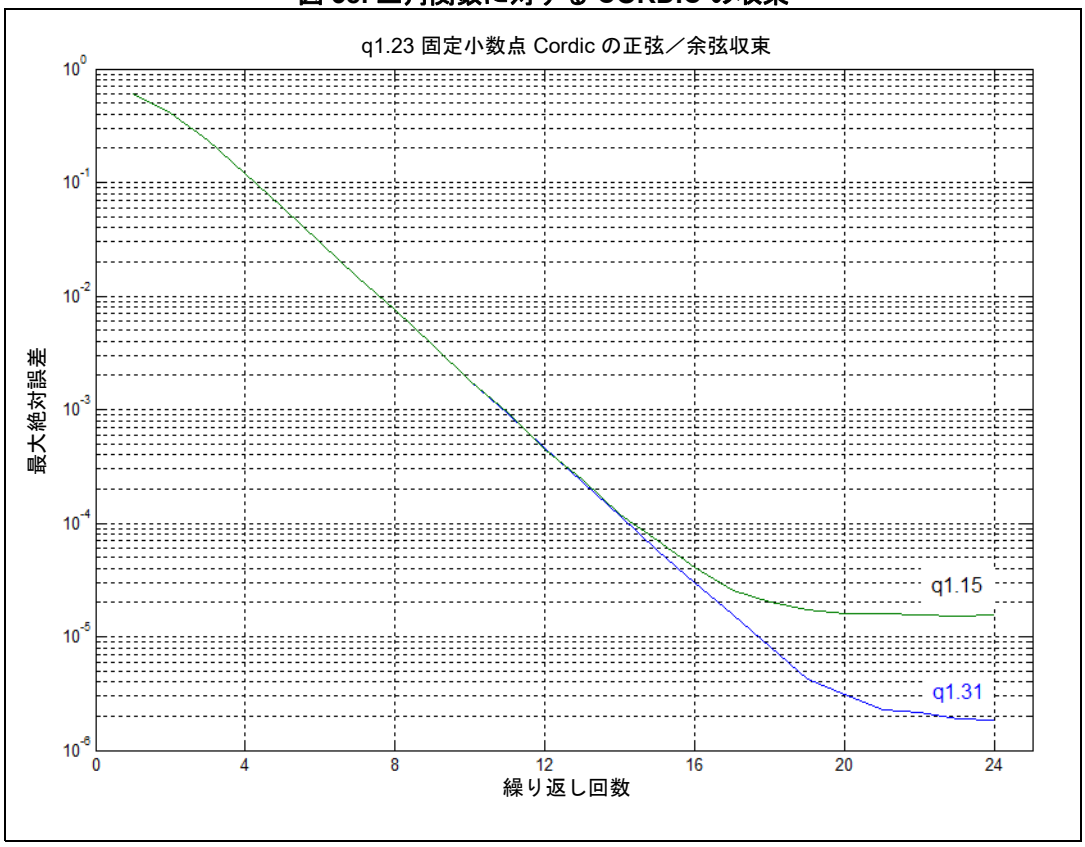


図 39. 双曲線関数に対する CORDIC の収束

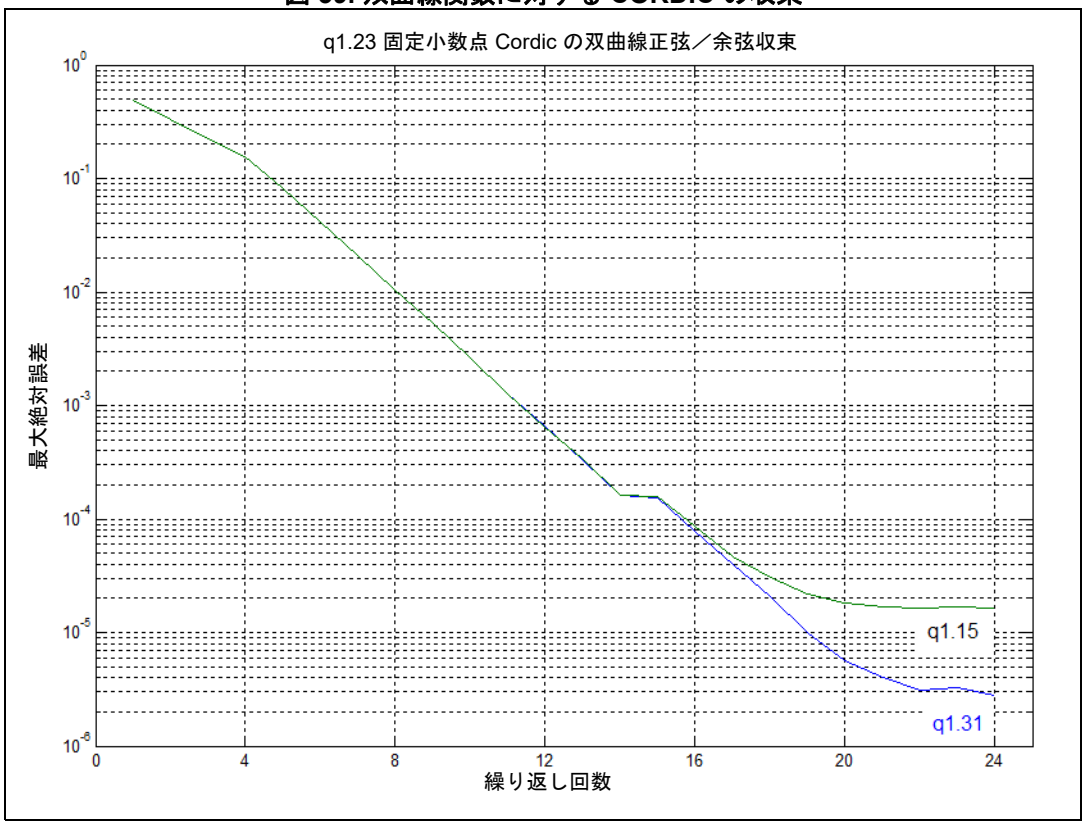
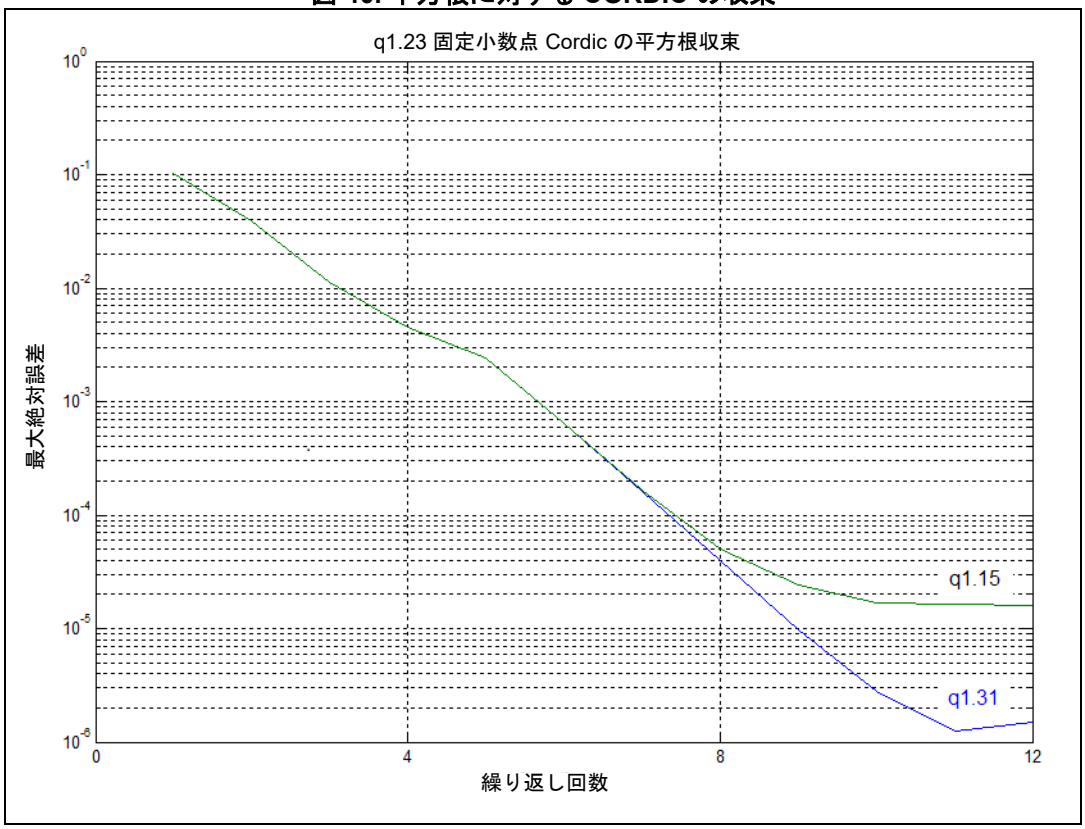


図 40. 平方根に対する CORDIC の収束



注： **収束速度は、量子化誤差が大きくなり始めるにつれて減少します。**

CORDICは、1 クロックサイクルあたり 4 回の繰り返しを実行可能です。それぞれの関数に対して、繰り返しを 4 回ずつ実行した後に残留している最大誤差を、その精度に到達するために必要なクロックサイクル数とともに表 115 に示します。この表から、必要なサイクル数を決定して、CORDIC_CR レジスタの PRECISION フィールドにプログラムすることができます。プログラムした回数の繰り返しが完了すると、コプロセッサが直ちに停止し、結果が読出し可能になります。

表 115. 精度対繰り返し回数

機能	繰り返し回数	サイクル数	最大残留誤差(1)	
			q1.31 フォーマット	q1.15 フォーマット
Sin、Cos、Phase(2)、Mod、Atan (4)	4	1	2^{-3}	2^{-3}
	8	2	2^{-7}	2^{-7}
	12	3	2^{-11}	2^{-11}
	16	4	2^{-15}	2^{-15}
	20	5	2^{-18}	2^{-16}
	24	6	2^{-19}	2^{-16}

表 115. 精度対繰り返し回数 （続き）

機能	繰り返し回数	サイクル数	最大残留誤差(1)	
			q1.31 フォーマット	q1.15 フォーマット
Sinh、Cosh、Atanh、Ln(3)	4	1	2^{-2}	2^{-2}
	8	2	2^{-6}	2^{-6}
	12	3	2^{-10}	2^{-10}
	16	4	2^{-13}	2^{-13}
	20	5	2^{-17}	2^{-15}
	24	6	2^{-18}	2^{-15}
Sqrt(4)	4	1	2^{-7}	2^{-7}
	8	2	2^{-14}	2^{-14}
	12	3	2^{-19}	2^{-15}

1. 最大残留誤差とは、倍精度浮動小数点で実行された同じ計算と比較して、所定の繰り返し回数後に残っている最大誤差のことです。この誤差のほかに、q15 フォーマットでは最大で 2^{-16}、q31 フォーマットでは最大で 2^{-20} の丸め誤差が発生することがあります。
2. モジュラスが 0.5 を上回る場合。到達可能な精度は、量子化誤差が大きくなるため、モジュラスの大きさに比例して減少します。
3. SCALE = 1。より大きなスケーリングファクタを用いると、それに比例して到達可能な精度が減少します。
4. SCALE = 0。より大きなスケーリングファクタを用いると、それに比例して到達可能な精度が減少します。

17.3.6 ゼロオーバーヘッドモード

コプロセッサを使用する最速の方法は、実行する関数（FUNC）、必要なクロックサイクル数（PRECISION）、入出力値のサイズ（ARGSIZE、RESSIZE）、入力引数の個数（NARGS）や結果の個数（NRES）、さらに、該当する場合にはスケーリングファクタ（SCALE）をCORDIC_CSR レジスタに前もってプログラムしておくことです。

その後、入力引数をCORDIC_WDATA レジスタに書き込むことにより、計算がトリガされます。正しい個数の入力引数が書き込まれると（かつ進行中の計算がすべて完了すると）、これらの入力引数ならびにその時点の CORDIC_CSR 設定値を用いて新しい計算が直ちに起動されます。変更がないのであれば、CORDIC_CSR レジスタを再プログラムする必要はありません。

32bit 入力引数が 2 個必要な場合（ARGSIZE = 0、NARGS = 1）、最初に第 1 入力引数 ARG1 を書き込み、その後に第 2 引数 ARG2 を書き込む必要があります。一連の計算において第 2 引数が不変である場合には、最初の計算が始まったところで引数の個数を 1（NARGS = 0）に再プログラムすることにより、2 番目の書込みを行わないことができます。関数が変更されない限り、第 2 引数はプログラムされた値を保持します。

注： **リセット後、ARG2 は +1（0x7FFFFFFF）にセットされます。**

16bit 引数が 2 個用いられる場合（ARGSIZE = 1）、ARG1 を最下位ハーフワード、ARG2 を最上位ハーフワードとして 32bit ワード 1 個に格納する必要があります。すると、格納された 32bit ワードは CORDIC_WDATA レジスタに書き込まれます。この場合（NARGS = 0）に必要な書込み回数は 1 回のみです。

入力引数が 1 つのみ（ARG1）の関数では、NARGS = 0 とすることをお勧めします。NARGS = 1 とした場合、計算をトリガするには CORDIC_WDATA に対して 2 回目の書込みを行う必要があります。この場合、ARG2 データは使用されません。

計算が始まってから CORDIC_RDATA レジスタの読出しを試みると、計算が完了するまでバス･ウェイトステートが挿入されてから結果が返されます。したがって、ソフトウェアが入力を書き込んだ直後に、有効かどうか確認するためのポーリングを行わずに結果を読み出すことができます。もしくは、プロセッサは、適切なクロックサイクル数だけ待ってから結果を読み出すこともできます。この時間は、次の計算のために CORDIC_CSR レジスタのプログラムを行い、必要に応じて次の入力データを準備することに使用可能です。CORDIC_CSR レジスタは計算の実行中に再プログラム可能であり、その計算結果に影響を与えることはありません。同様に、ひとたび前回の引数が反映されれば、次の引数で CORDIC_WDATA レジスタを更新できます。次の引数と設定値は、前の計算が完了するまで保留されます。

計算が完了すると、その結果は CORDIC_RDATA レジスタから読み出すことができます。32bit の結果が 2 個期待されている場合（NRES = 1、RESSIZE = 0）、最初に第 1 結果（RES1）を読み出し、その後に第 2 結果（RES2）を読み出す必要があります。32bit の結果が 1 つだけ期待されている場合（NRES = 0、RESSIZE = 0）、RES1 が 1 回目の読出しで出力されます。

16bit の結果が期待されている場合（RESSIZE = 1）、CORDIC_RDATA を 1 回読み出すと、両方の結果が 32bit ワードに詰め込まれてフェッチされます。RES1 は下位ハーフワード、RES2 は上位ハーフワードに格納されます。この場合には、NRES = 0 とプログラムすることをお勧めします。NRES = 1 とした場合、次の操作のために CORDIC を開放する目的で CORDIC_RDATA の 2 回目の読出しを行う必要があります。この 2 回目の読出しによるデータは破棄する必要があります。

次の計算は、期待される個数の引数が書込み済みであれば、期待される個数の結果が読み出された時点で始まります。このことは、常に、実行中または結果読出し待ちの計算が 1 つ、保留中の演算が 1 つ存在し得ることを意味しています。演算の保留中に CORDIC_WDATA にさらにアクセスすると、保留中の演算はキャンセルされ、データが上書きされます。

以下のシーケンスは、ゼロオーバーヘッド･モードにおける CORDIC_IP の使用方法の要約です。

1. CORDIC_CSR レジスタに適切な設定値をプログラムします。
2. CORDIC_WDATA レジスタに最初の計算の引数をプログラムします。これによって最初の計算が始まります。
3. 必要に応じて、次の計算のための設定値で CORDIC_CSR レジスタ を更新します。
4. CORDIC_WDATA レジスタに次の計算の引数をプログラムします。
5. CORDIC_RDATA レジスタから結果を読み出します。これによって次の計算がトリガされます。
6. ステップ 3 に戻ります。

17.3.7 ポーリングモード

新しい結果が CORDIC_RDATA レジスタに有効になると、CORDIC_CSR レジスタに RRDY フラグがセットされます。このフラグは、そのレジスタを読み出すことによってポーリング可能です。CORDIC_RDATA レジスタを（CORDIC_CSR レジスタの NRES フィールドに応じて 1 回か 2 回）読み出すと、このフラグはリセットされます。

有効になり次第結果が読み出されるのではないため、RRDY フラグのポーリングは、CORDIC_RDATA レジスタを直接読み出すよりも多少時間がかかります。しかしながら、CORDIC_CSR レジスタの読出しの間にプロセッサとバスインタフェースが停止しないことから、プロセッサの停止が許容されない場合（低遅延割込みが必要な場合など）に、このモードが考慮の対象となる可能性があります。

17.3.8 割込みモード

CORDIC_CSR レジスタの割込みイネーブル（IE）ビットをセットすることにより、RRDY フラグがセットされるたびに割込みが生成されます。割込みは、このフラグがリセットされたときにクリアされます。

このモードによって、割込みサービスルーチンの中で計算結果を読み出すことが可能となり、他のタスクよりも高い優先度が与えられます。ただし、割込み処理の遅延のため、結果の直接読出しやフラグのポーリングよりも時間がかかります。

17.3.9 DMA モード

CORDIC_CSR レジスタの DMA 書込みイネーブル（DMAWEN）ビットがセットされ、保留となっている演算がない場合、DMA 書込みチャネル·リクエストが生成されます。DMA コントローラは、メモリから CORDIC_WDATA レジスタへと第 1 入力引数（ARG1）を転送可能となります。レジスタに書き込むと、DMA リクエストはデアサートされます。CORDIC_CSR レジスタの NARGS = 1 である場合、2 番目の DMA 書込みチャネル·リクエストが生成され、第 2 入力引数（ARG2）が CORDIC_WDATA レジスタに転送されます。入力引数がすべて書き込まれ、（結果の読出しによって）実行中の計算がすべて完了すると、新しい計算が始まり、さらに DMA 書込みチャネル·リクエストが生成されます。

CORDIC_CSR レジスタの DMA 読出しイネーブル（DMAREN）ビットがセットされている場合、RRDY フラグがアクティブになると DMA 読出しチャネル·リクエストが生成されます。これにより、DMA コントローラは、CORDIC_RDATA レジスタからメモリへ第 1 結果（RES1）を転送可能となります。レジスタを読み出すと、DMA リクエストはデアサートされます。CORDIC_CSR レジスタの NRES = 1 である場合、2 番目の DMA リクエストが生成され、第 2 結果（RES2）が読み出されます。結果がすべて読み出されると、RRDY フラグがデアサートされます。

DMA 読出しチャネルと書込みチャネルは、別々に有効化できます。両方のチャネルが有効化された場合、CORDIC は、プロセッサの介入なしに計算をデータバッファ上で自律的に繰り返すことができます。これにより、プロセッサは別のタスクを実行できます。DMA コントローラは、書込みチャネルでは“メモリからペリフェラル”モード、読出しチャネルでは“ペリフェラルからメモリ”モードで動作しています。プロセッサが DMAWEN フラグをセットするとシーケンスが開始されることに注意してください。それ以降、CORDIC がデータを処理可能な速度で DMA 読出しリクエストと DMA 書込みリクエストが生成されます。

場合によっては、入力データはメモリに格納され、出力は一定の時間間隔で D/A コンバータなどの別のペリフェラルへと転送されることがあります。この場合、転送先のペリフェラルは、新しいデータを必要とするたびに DMA リクエストを生成します。DMA コントローラは、次のサンプルを CORDIC_RDATA レジスタから直接フェッチすることができます（この場合、DMA コントローラは、転送元がペリフェラル·レジスタであっても“メモリからペリフェラル”モードで動作します）。結果を読み出す行為によって、CORDIC は新しい計算を開始できるようになりますが、その結果として DMA 書込みチャネルリクエストが生成され、DMA コントローラが次の入力値を CORDIC_WDATA レジスタに転送します。DMA 書込みチャネルは有効化（DMAWEN = 1）されますが、読出しチャネルは有効化しないでください。

同様に、ADC など別ペリフェラルからのデータは、（“ペリフェラルからメモリ”モードで）CORDIC_WDATA レジスタに直接転送可能です。DMA 書込みチャネルは有効化しないでください。CORDIC は入力データを処理し、それが完了すると DMAREN = 1 の場合、DMA 読出しリクエストを生成します。これにより、DMA コントローラは、CORDIC_RDATA レジスタからメモリへ結果を転送します（“ペリフェラルからメモリ”モード）。

注： **CORDIC_CSR レジスタをプログラムする DMA リクエストは生成されません。したがって、DMA モードが有用なのは、同じ関数を同じ設定で繰り返し実行する場合に限られます。一連の DMA 転送の間、スケール·ファクタは変更できません。**

注： CORDIC_WDATA レジスタや CORDIC_RDATA レジスタに対して DMA が実行されるたびに、該当の DMA リクエストに対する確認応答を送信する必要があります。この確認応答の前にこれらのレジスタへの無関係なアクセスが発生すると、確認応答が本来よりも早い時点でアサートされ、DMA チャネルがブロックされることがあります。したがって、DMA 読出しチャネルが有効なときは、CPU から CORDIC_RDATA レジスタにアクセスしないようにする必要があります。同様に、DMA 書込みチャネルが有効化されたときには、プロセッサは CORDIC_WDATA レジスタへのアクセスを避ける必要があります。

17.4 CORDIC レジスタ

CORDIC レジスタには 32bit ワード･フォーマットでのみアクセスできます。

17.4.1 CORDIC 制御/ステータス･レジスタ （CORDIC_CSR）

アドレスオフセット：0x00

リセット値：0x0000 0050

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARG SIZE	RES SIZE	NARGS	NRES	DMA WEN	DMA REN	IEN
r									rw	rw	rw	rw	rw	rw	rw

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

ビット 31 **RRDY**：結果レディ･フラグ

0：出力レジスタに新しい結果がありません。

1：CORDIC_RDATA レジスタに新しいデータが入っています。

このビットは、CORDIC の演算が完了したときに、ハードウェアによってセットされます。

CORDIC_RDATA レジスタが（NRES+1）回読み込まれると、ハードウェアによってリセットされます。

このビットがセットされたときに、IEN ビットもセットされていると、CORDIC 割込みがアサートされます。DMAREN ビットがセットされていると、DMA 読出しチャネルリクエストが生成されます。このビットがセットされている間は、新しい計算は開始されません。

ビット 30:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **ARGSIZE**：入力データの幅

0：32 ビット

1：16 ビット

ARGSIZE によって入力データの表現に用いられるビット数が選択されます。

32bit データが選択されると、CORDIC_WDATA レジスタは q1.31 フォーマットの引数を期待します。

16bit データが選択されると、CORDIC_WDATA レジスタは q1.15 フォーマットの引数を期待します。

最下位ハーフワードに第 1 引数（ARG1）、最上位ハーフワードに第 2 引数（ARG2）が書き込まれます。

ビット 21 **RESSIZE**：出力データの幅

0：32 ビット

1：16 ビット

RESSIZE によって出力データの表現に用いられるビット数が選択されます。

32bit データが選択されると、CORDIC_RDATA レジスタには q1.31 フォーマットの結果が含まれます。

16bit データが選択されると、CORDIC_RDATA の最下位ハーフワードには第 1 結果（RES1）が q1.15 フォーマットで含まれ、最上位ハーフワードには第 2 結果（RES2）がこちらも q1.15 フォーマットで含まれます。

ビット 20 **NARGS**：CORDIC_WDATA レジスタが期待する引数の個数

0：次の計算には 1 回の 32bit 書込みのみ（もしくは ARGSIZE = 1 の場合に 16bit 値 2 個）が必要です。

1：次の計算のトリガには、32bit 値を 2 個 CORDIC_WDATA レジスタに書き込む必要があります。

読み出すと、ビットの現在の状態が返されます。

ビット 19 **NRES**：CORDIC_RDATA レジスタ内の結果の個数

0：次の計算が完了すると、32bit 値 1 個のみ（もしくは RESSIZE = 1 の場合に 16bit 値 2 個）が CORDIC_RDATA レジスタに転送されます。CORDIC_RDATA から 1 回読み出すと、RRDY フラグがリセットされます。

1：次の計算が完了すると、32bit 値が 2 個 CORDIC_RDATA レジスタに転送されます。RRDY フラグのリセットには、CORDIC_RDATA から 2 回読み出す必要があります。

読み出すと、ビットの現在の状態が返されます。

ビット 18 **DMAWEN**：DMA 書込みチャネルの有効化

0：無効。DMA 書込みリクエストは生成されません。

1：有効となっています。保留中の演算が存在しない場合には、DMA 書込みチャネルにリクエストが常に生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 17 **DMAREN**：DMA 読出しチャネルの有効化

0：無効。DMA 読出しリクエストは生成されません。

1：有効となっています。RRDY フラグがセットされるたびに、DMA 読出しチャネルにリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 16 **IEN**：割込みの有効化

0：無効。割込みリクエストは生成されません。

1：有効となっています。RRDY フラグがセットされるたびに、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **SCALE[2:0]**：スケーリング･ファクタ

このフィールドの値は、引数や結果に適用されたスケーリングファクタを示します。値 n は、引数に係数 2^{-n} が掛けられているか、結果に 2^{n} を掛ける必要のあることを意味します。スケーリングファクタの各関数に対する適用可能性と適用範囲については、セクション 17.3.2： を参照してください。

ビット 7:4 **PRECISION[3:0]**：必要精度（繰り返し回数）

0：予約済み

1 から 15：（繰り返し回数）／4

特定の精度に対して必要となる繰り返し回数を求めるには、表 115 を参照してください。

ほとんどの関数では、このフィールドの推奨範囲は 3 ~ 6 であることに注意してください。

ビット 3:0 **FUNC[3:0]**：関数

0：余弦

1：正弦

2：位相

3：ベクトルの絶対値

4：逆正接

5：双曲線余弦

6：双曲線正弦

7：双曲線逆正接

8：自然対数

9：平方根

17.4.2 CORDIC 引数レジスタ（CORDIC_WDATA）

アドレスオフセット：0x04

リセット値：0xXXXX XXXX

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

ビット 31:0 **ARG[31:0]**：関数の入力引数

このレジスタには、CORDIC_CSR レジスタの FUNC フィールドで選択された関数に対する入力引数をプログラムします。

32bit フォーマットが選択され（CORDIC_CSR.ARGSIZE = 0）、入力引数が 2 個必要である場合（CORDIC_CSR.NARGS = 1）、このレジスタに 2 回の連続した書込みが必要となります。1 回目には第 1 引数（ARG1）を書き込み、2 回目には第 2 引数（ARG2）を書き込みます。

32bit フォーマットが選択され、必要な入力引数は 1 個のみである場合（NARGS = 0）、このレジスタに必要な書込みは第 1 引数（ARG1）が含まれた 1 回のみとなります。

16bit フォーマットが選択された場合（CORDIC_CSR.ARGSIZE = 1）、このレジスタに対する 1 回の書込みには両方の引数が含まれます。下位ハーフの ARG[15:0] に第 1 引数（ARG1）、上位ハーフの ARG[31:16] に第 2 引数（ARG2）が含まれます。この場合、NARGS は 0 にセットする必要があります。

各関数に必要な引数とその許容範囲については、セクション 17.3.2： を参照してください。

必要個数の引数が書き込まれると、前回の計算が完了していれば、CORDIC は、与えられた入力引数を用いて CORDIC_CSR.FUNC によって指定された関数を評価します。計算が実行中である場合、その計算が完了して結果が読み出されるまで、ARG1 と ARG2 の値は保留されます。この間にレジスタへの書込みがあると、保留中の演算はキャンセルされ、引数データが上書きされます。

17.4.3 CORDIC 結果レジスタ（CORDIC_RDATA）

アドレスオフセット：0x8

リセット値：0x0000 0000

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

ビット 31:0 **RES[31:0]**：関数の結果

32bit フォーマットが選択され（CORDIC_CSR.RESSIZE = 0）、出力値が 2 個期待されている場合（CORDIC_CSR.NRES = 1）、このレジスタは RRDY がセットされたときに 2 回読み出される必要があります。1 回目の読出しによって第 1 結果（RES1）がフェッチされます。2 回目の読出しによって第 2 結果（RES2）がフェッチされ、RRDY がリセットされます。

32bit フォーマットが選択され、期待されている出力値が 1 個のみである場合（NRES = 0）、第 1 結果（RES1）のフェッチと RRDY フラグのリセットに必要となるこのレジスタの読出しは 1 回のみです。

16bit フォーマットが選択された場合（CORDIC_CSR.RESSIZE = 1）、このレジスタの下位ハーフの RES[15:0] に第 1 結果（RES1）、上位ハーフの RES[31:16] に第 2 結果（RES2）が含まれます。この場合、NRES は 0 にセットする必要があり、実行される読出しは 1 回のみとなります。

このレジスタを読み出すと、CORDIC_CSR レジスタの RRDY フラグがリセットされます。

17.4.4 CORDIC レジスタマップ

表 116. CORDIC レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	CORDIC_CSR	RRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARGSIZE	RESSIZE	NARGS	NRES	DMAWEN	DMAREN	IEN	Res.	Res.	Res.	Res.	SCALE [2:0]				PRECISION [3:0]				FUNC [3:0]			
	リセット値	0									0	0	0	0	0	0	0					0	0	0	0	0	1	0	1	0	0	0	0
0x04	CORDIC_WDATA	ARG[31:0]																															
	リセット値	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0x08	CORDIC_RDATA	RES[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2：を参照してください。

18 フィルタ演算アクセラレータ（FMAC）

18.1 FMAC の概要

フィルタ演算アクセラレータユニットは、ベクトルの算術演算を実行します。積和演算器（MAC）ユニットから構成されており、ローカルメモリに保持されているベクトル要素のインデックス付けを可能とするアドレス生成ロジックを備えています。

このユニットには、デジタルフィルタの実装を可能とする、入出力上のサーキュラバッファサポートが含まれています。有限インパルス応答フィルタと無限インパルス応答フィルタのどちらも実現可能です。

このユニットによって、高頻度であったり時間がかかったりするフィルタリング動作を CPU からオフロードして、プロセッサを他のタスクに開放することができます。多くの場合には、このような計算をソフトウェアよりも高速に実行可能ですので、速度重視のタスクが高速化されます。

18.2 FMAC の主な機能

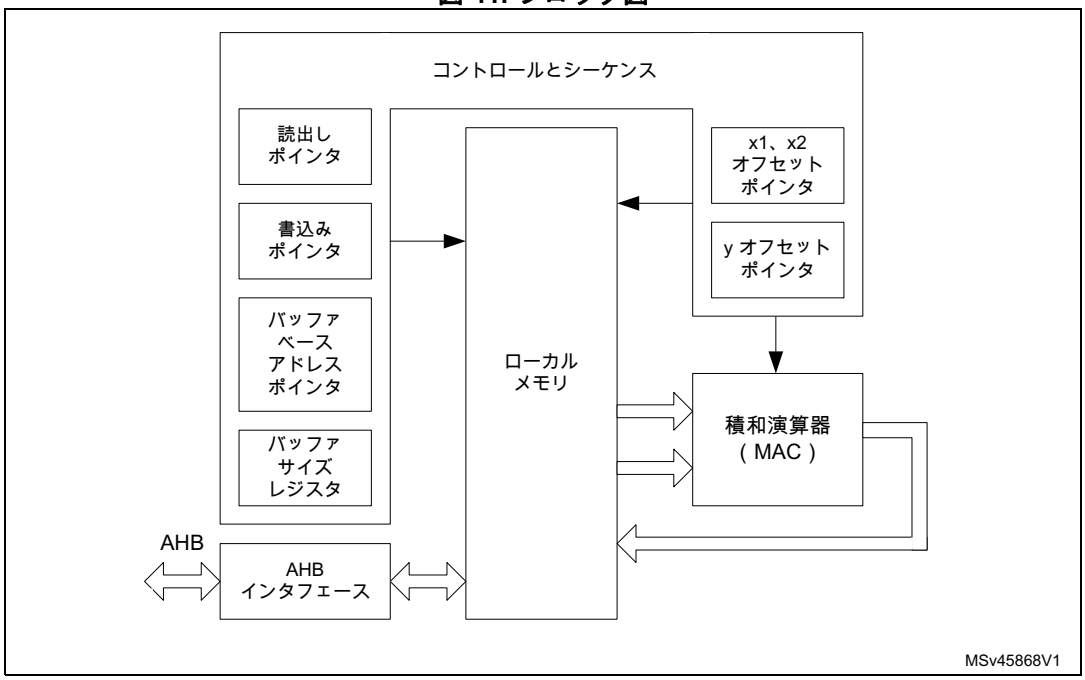
- 16 x 16bit 乗算器
- 加減算対応 24 + 2bit アキュムレータ
- 16bit 入出力データ
- 256 x 16bit ローカルメモリ
- メモリの最大 3 領域（2 領域は入力、1領域は出力）をデータバッファとして定義可能（プログラム可能なベースアドレスポインタおよび付随するサイズレジスタによって定義）
- 入出力バッファはサーキュラバッファ対応可能
- フィルタ機能：FIR、IIR（直接型 1）
- ベクトル関数：ドット積、畳み込み、相関
- AHB スレーブインタフェース
- DMA 読出し／ライトデータチャネル

18.3 FMAC の機能詳細

18.3.1 概要

FMACを 図 41 に示します。

図 41. ブロック図



このユニットは、固定小数点積和演算器（MAC）の周辺に構築されています。MAC は、メモリから16bit 符号付き入力の値を 2 個とり、互いに掛け合わせてアキュムレータの内容に加算します。メモリの入力値アドレスは、ポインタへセットすることにより決定されます。これらのポインタは、内部ハードウェアによってロード、インクリメント、デクリメント、リセットが行えます。ポインタと MAC 演算は、要求された演算を実行するために、内蔵シーケンサによって制御されます。

ドット積を計算するために、プロセッサまたは DMA によって 2 本の入力ベクトルがローカルメモリにロードされ、要求された演算を選択して開始します。入力ベクトル要素の各ペアは、メモリからフェッチされ、互いに掛け合わされてから累積されます。ベクトル要素の処理がすべて完了するとアキュムレータの内容がローカルメモリに格納され、そこからプロセッサや DMA による読出しが行えます。

有限インパルス応答（FIR）フィルタ演算（畳み込みとも呼ばれます）では、係数ベクトルと入力サンプル·ベクトルのドット積が繰り返し計算されます。繰り返しのたびに、入力サンプルを 1 サンプル分のディレイでシフトし、最も古いサンプルを破棄して新しいサンプルを追加します。

無限インパルス応答（IIR）フィルタ演算では、フィードバック係数と前回の出力サンプルとの畳み込みが FIR 畳み込みの結果に加算されます。

フィルタ演算の詳細は、セクション 18.3.6：フィルタ機能に記載されています。

18.3.2 ローカルメモリとバッファ

このユニットには、ローカル･ストレージとして用いられる 256 x 16bit リード／ライト･メモリが含まれています。

- 入力値（入力ベクトルの要素）は、バッファ X1 と X2 に格納されます。
- 出力値（演算結果）は、別のバッファ Y に格納されます。
- バッファのサイズと位置は、次のように指定されています。
  - x1_base：X1 バッファのベースアドレス
  - x2_base：X2 バッファのベースアドレス
  - y_base：Y バッファのベースアドレス
  - x1_buf_size：X1 バッファに割り当てられたデータ数（16bit ワード単位）
  - x2_buf_size：X2 バッファに割り当てられたデータ数（16bit ワード単位）
  - y_buf_size：Y バッファに割り当てられたデータ数（16bit ワード単位）

これらのパラメータは、ユニットの設定時に、対応するレジスタにプログラムされます。

CPU（または DMA コントローラ）は、初期化機能（セクション 18.3.5：初期化機能）を用いてライト･データ･レジスタに書き込むことにより、各バッファの内容を初期化できます。データは、ライト･ポインタによって示されるターゲット･バッファ内の位置に転送されます。新たな書込みが終わるたびに、ライトポインタはインクリメントされます。ライトポインタが割り当てられたバッファ空間の最後に達すると、ベースアドレスに戻って折り返します。この機能は、演算の前にベクトルの要素をロードしたり、フィルタを初期化してフィルタ係数をロードしたりするために用いられます。

バッファ設定

X1、X2 および Y バッファ設定レジスタでバッファサイズとベースアドレスオフセットを設定する必要があります。各機能に必要なバッファサイズは、セクション 18.3.6：フィルタ機能の機能詳細に規定されています。ベースアドレスは内部メモリのどこにでも選択可能ですが、すべてのバッファが内部メモリのアドレス範囲（0x00～0xFF）に収まることが条件となっていますので、ベースアドレス + バッファサイズは 256 未満である必要があります。

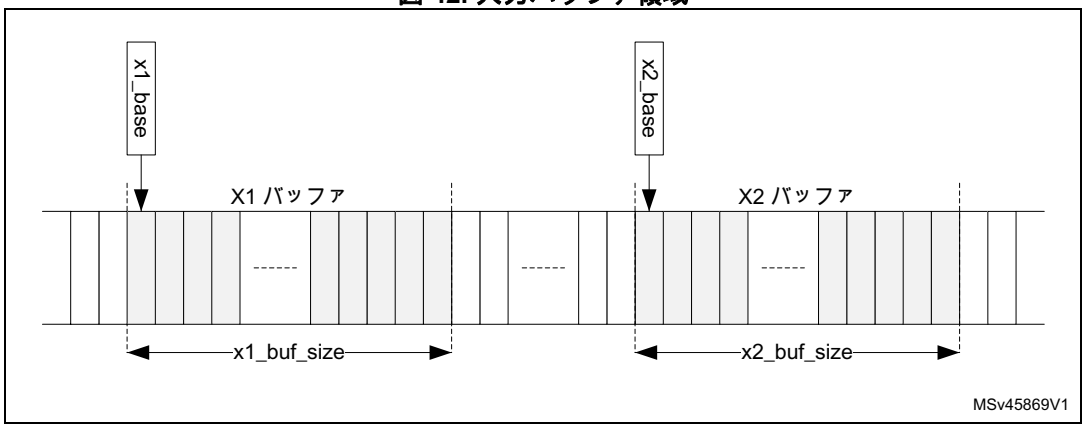
バッファのサイズと位置に関する制約はありません（オーバーラップしていても、完全に一致していても構いません）。フィルタ機能では、こうすると不正な動作を招くおそれがあるため、バッファをオーバーラップさせないことをお勧めします。

サーキュラバッファ演算が必要である場合には、オプションの「ヘッドルーム」d をバッファサイズに加算しても構いません。さらに、ウォーターマークレベルを設定して CPU や DMA の動作を調整することもできます。d の値とウォーターマークレベルは、アプリケーション性能要件に応じて選択してください。スループットを最大とするためには、入力バッファは決して空となってはなりませんので、d はウォーターマークレベルよりも多少大きな値とし、割込みや DMA の遅延を可能とするべきです。その一方で、ユニットの処理可能速度に見合った速さで入力データを供給できない場合には、次のデータが書き込まれるのを待つためにバッファを空にすることができるので、（入力にオーバーフローが発生しないことを保証するために）d はウォーターマークレベルと等しくとも構いません。

18.3.3 入力バッファ

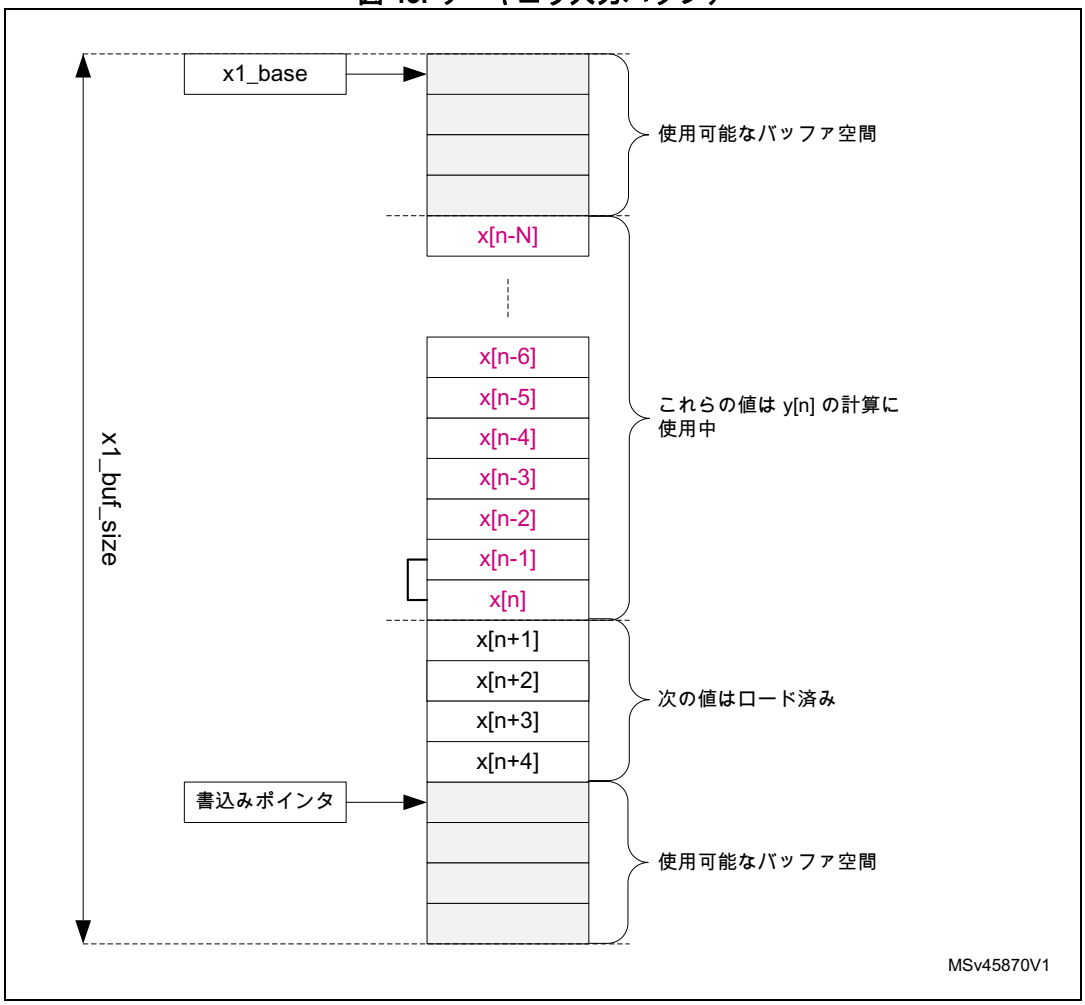
X1 バッファと X2 バッファは、MAC に入力するデータの格納に用いられます。乗算のたびに X1 バッファからの値と X2 バッファからの値が互いに掛け合わされます。制御ユニットのポインタによって、それぞれの値に対する（バッファベースアドレスを基準とした）リードアドレスオフセットが生成されます。ポインタは、現在の機能に応じてハードウェアによって管理されます。

図 42. 入力バッファ領域



X1 バッファはサーキュラバッファとして使用できますが、この場合、バッファに空きがあるときには必ず入力バッファに新しいデータが連続転送されます。デジタル･フィルタでは、このバッファのプリロードは任意となります。それは、演算が開始されたときにバッファに入力サンプルが何も書き込まれていない場合にエンプティ･フラグが立ち、このフラグによって、演算を開始するのに十分な量となるまで CPU または DMA が新しいサンプルをロードする動作がトリガされるためです。とはいえ、入力データがすでにシステム･メモリの中にあってサーキュラ演算が必要ではないベクトル演算の場合はプリロードが有用です。

図 43. サーキュラ入力バッファ



X2 バッファは（サーキュラ･モードではない）ベクトル･モードでのみ使用可能であり、バッファの内容が演算と演算の間で変化しない場合を除いて、プリロードする必要があります。フィルタ機能の場合は、X2 バッファを使用してフィルタ係数が格納されます。

サーキュラバッファとしての動作時には、バッファ内に新しい値が常に存在しているようにするために、バッファに割り当てられたサイズ（x1_buf_size）は、現在の計算に使用されている要素数よりも大きくしなければならないのが一般的です。図 43 にフィルタ動作のためのバッファレイアウトが図示されています。出力サンプル y[n] を計算している間、このユニットは、x[n-N] から x[n] まで N+1 個の入力サンプルのセットを使用します。計算が完了すると、このユニットは、x[n-N+1] から x[n+1] までの入力サンプルのセットを使用して y[n+1] の計算を開始します。最も古い入力サンプル x[n-N] が入力セットから除外され、新しいサンプル x[n+1] が追加されます。

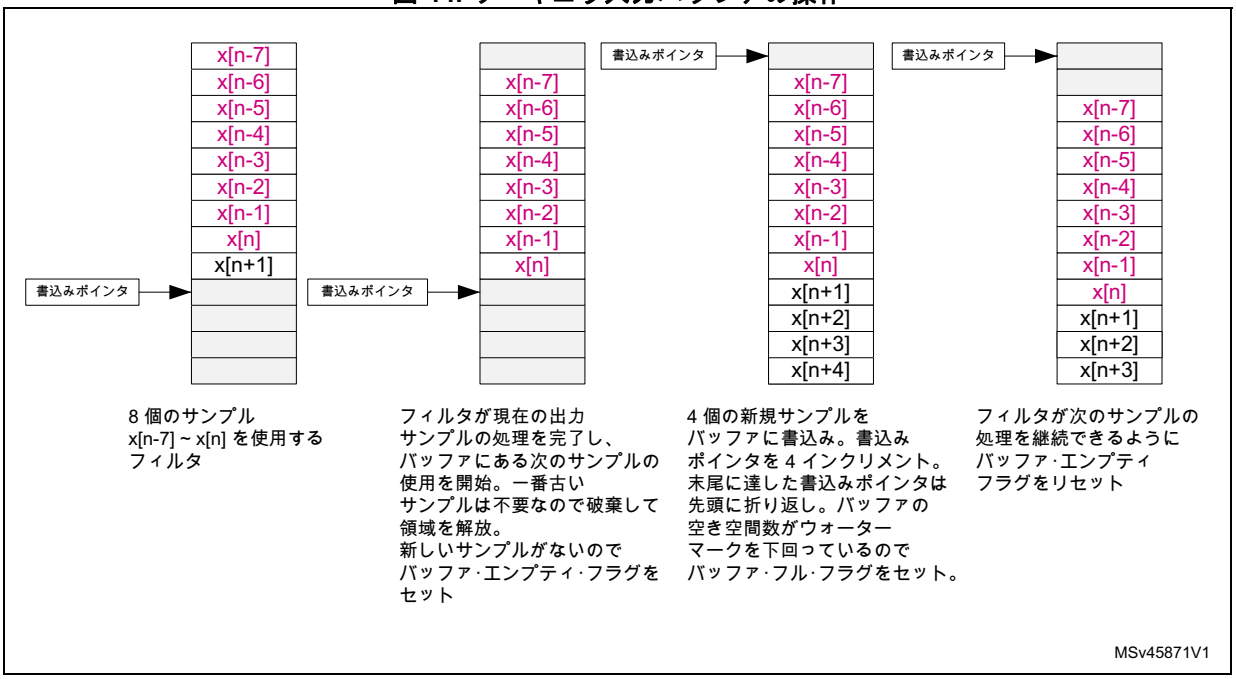
プロセッサや DMA コントローラは、必要時に新しいサンプル x[n+1] がバッファの中で利用可能であることを保証する必要があります。そうならない場合、バッファにはエンプティ･フラグが立ち、新しいサンプルが追加されるまでユニットの実行は停止します。X1 バッファについては、アンダーフロー状態が通知されません。

注： **タイマや ADC などのペリフェラルによってサンプルのフローが制御されている場合、転送元が供給可能な速度よりも、フィルタがそれぞれの新しいサンプルを処理する速度が速いため、バッファは定期的に空となります。これは、フィルタ演算の基本的な特徴です。**

FMAC_X1BUFCFG レジスタの FULL_WM ビットフィールドにプログラムされたウォーターマーク閾値よりもバッファ内の空きスペースの個数の方が少なくなると、バッファ·フルのフラグが立ちます。フル·フラグがセットされていない限り、割込みが生成されて（有効となっている場合）バッファへの追加データが要求されます。ウォーターマークによって、オーバーフローの危険を伴わずに、1回の割込みで複数のデータが転送可能となります。それにもかかわらずオーバーフローが発生した場合には、OVFL エラーフラグがセットされて、ライトデータは無視されます。オーバーフローが発生した場合には、ライト·ポインタがインクリメントされません。

フィルタ演算中の X1 バッファの操作を 図 44 に示します。この例には、ウォーターマークが 4 にセットされた 8 タップ FIR フィルタが示されています。

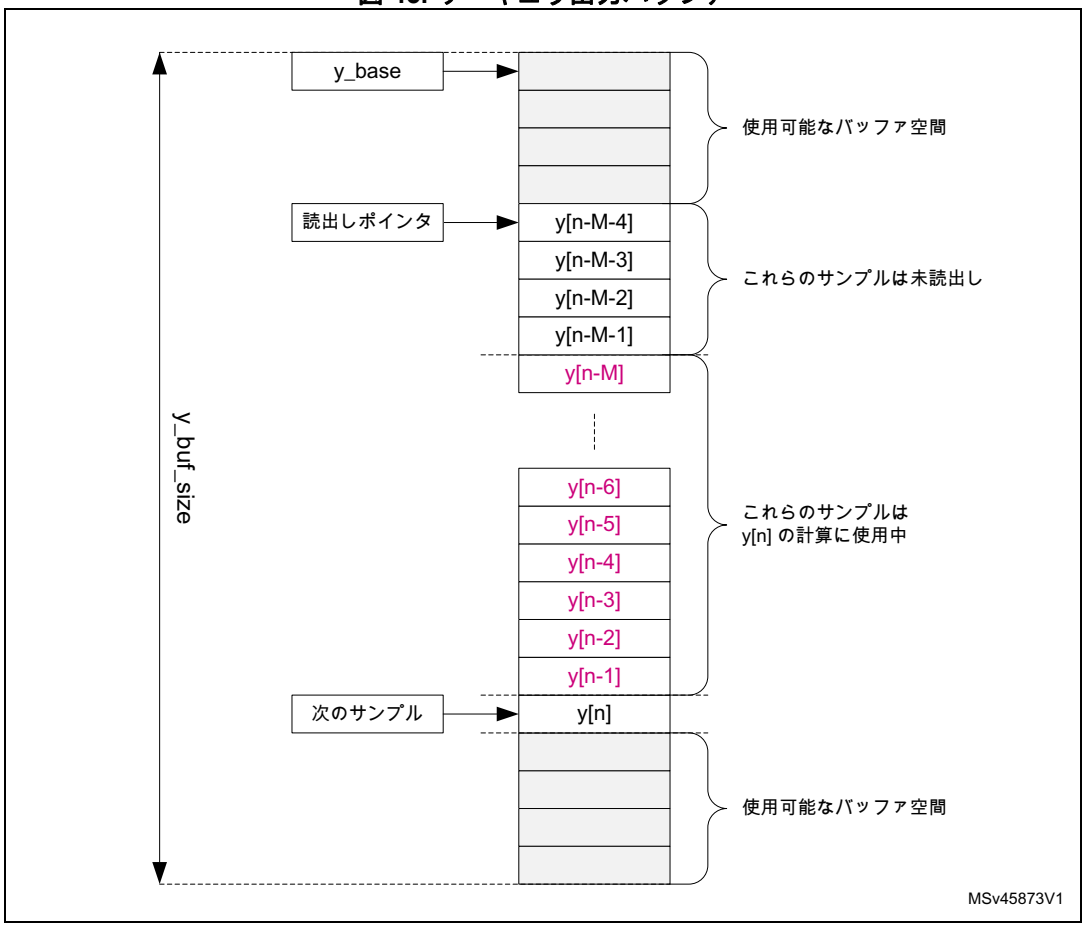
図 44. サーキュラ入力バッファの操作



18.3.4 出力バッファ

Y（出力）バッファは、演算結果の格納に用いられます。それぞれの新しい出力値は、プロセッサか DMA コントローラによって読み出されるまでバッファの中に格納されています。リードデータレジスタに対してリードアクセスが行われるたびに、リードポインタで示されているアドレスからリードデータがフェッチされます。毎回の読出しの後にこのポインタはインクリメントされ、割り当てられた Y バッファ空間の最後に到達すると、ベースアドレスに戻って折り返されます。

図 45. サーキュラ出力バッファ



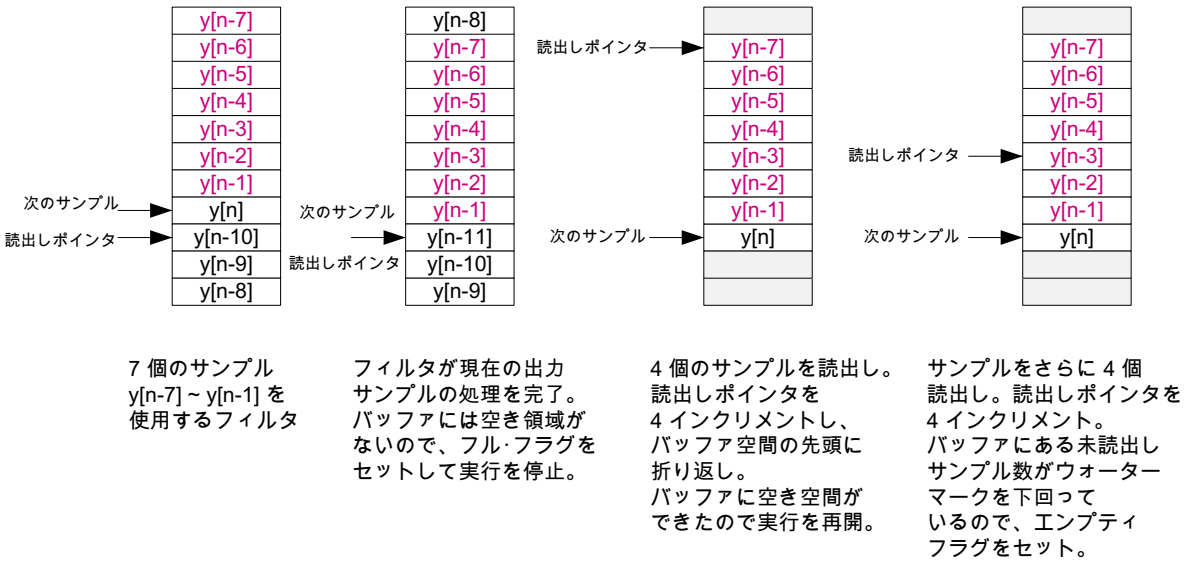
Y バッファはサーキュラバッファとしても使用可能です。次の出力値のアドレスがリード･ポインタで示されているものと同一である場合（未読出しのサンプル）には、バッファにはフル･フラグが立ち、サンプルが読み出されるまで実行は停止します。

IIR フィルタの場合、Y バッファは、次の出力サンプル y[n] の計算に使う y[n-M] から y[n-1] まで M 個の過去の出力サンプル･セットの格納に用いられます。新しいサンプルがセットに追加されるたびに、最も古いサンプルである y[n-M] が脱落します。

FMAC_YBUFCFG レジスタの EMPTY_WM ビットフィールドにプログラムされたウォーターマーク閾値よりもバッファ内の未読出しデータの個数の方が少ない場合、バッファにはエンプティ･フラグが立ちます。エンプティ･フラグがセットされていない限り、割込みまたは DMA リクエストが生成されて（有効となっている場合）バッファからの読出しが要求されます。ウォーターマークによって、アンダーフローの危険を伴わずに、1 回の割込みで複数のデータが転送可能となります。それにもかかわらずアンダーフローが発生した場合には、UNFL エラーフラグがセットされます。この場合、リード･ポインタはインクリメントされずに、リード･ポインタのアドレスにあるメモリの内容が読出し操作によって返されます。

サーキュラ･モードにおける Y バッファの操作を 図 46 に示します。この例には、ウォーターマークが 4 にセットされた 7 タップ IIR フィルタが示されています。

図 46. サーキュラ出力バッファの操作



18.3.5 初期化機能

以下の各機能によって FMAC ユニットが初期化されます。これらの機能は、START ビットをセットして、FMAC_PARAM レジスタの FUNC ビットフィールドに適切な値を書き込むことによってトリガされます。以下に示すように、P ビットフィールドと Q ビットフィールドにも各機能に適したパラメータ値が含まれている必要があります。R ビットフィールドは使用しません。この機能が完了すると、START ビットはハードウェアによって自動的にリセットされます。

初期化している間は、DMA リクエストと割込みを無効化していくことをお勧めします。フロー制御は不要であるため、FMAC メモリへのデータ転送は、ソフトウェアでもメモリ間 DMA 転送でも行うことができます。

ロード X1 バッファ

この機能を使用すると、X1_BASE にあるアドレスから始まる N 個の値が X1 バッファにプリロードされます。FMAC_WDATA レジスタへの連続書込みを行うと、X1 バッファにライト·データがロードされて、ライト·アドレスがインクリメントされます。この機能が完了すると、ライト·ポインタはアドレス X1_BASE + N を指しています。

この機能は、あるベクトルの要素をバッファにプリロードしたり、あるフィルタの入力ストレージ要素を初期化したりするのに使用できます。

パラメータ

- パラメータ P には、X1 バッファにロードされる値の個数 N が含まれています。
- パラメータ Q と パラメータ R は使用しません。

この機能は、FMAC_WDATA レジスタに N 回の書込みが行われると完了します。

ロード X2 バッファ

この機能を使用すると、X2_BASE にあるアドレスから始まる N + M 個の値が X2 バッファにプリロードされます。FMAC_WDATA レジスタへの連続書込みを行うと、X2 バッファにライト･データがロードされて、ライト･アドレスがインクリメントされます。

この機能は、ベクトルの要素やフィルタの係数をバッファにプリロードするのに使用できます。IIR の場合には、N 個のフィードフォワード係数と M 個のフィードバック係数が連結され、一緒に X2 バッファにロードされます。係数の総数は、N + M に等しくなります。FIR の場合には、フィードバック係数は存在しないので、M = 0 となります。

パラメータ

- パラメータ P には、アドレス X2_BASE から始まって X2 バッファにロードされる値の個数 N が含まれています。
- パラメータ Q には、アドレス X2_BASE + N から始まって X2 バッファにロードされる値の個数 M が含まれています。
- パラメータ R は使用しません。

この機能は、FMAC_WDATA レジスタに N + M 回の書込みが行われると完了します。

ロード Y バッファ

この機能を使用すると、Y_BASE にあるアドレスから始まる N の値が Y バッファにプリロードされます。FMAC_WDATA レジスタへの連続書込みを行うと、Y バッファにライト･データがロードされて、ライト･アドレスがインクリメントされます。この機能が完了すると、リード･ポインタはアドレス Y_BASE + N を指しています。

この機能は、IIR フィルタのフィードバック･ストレージ要素をプリロードするのに使用できます。

パラメータ

- パラメータ P には、Y バッファにロードされる値の個数が含まれています。
- パラメータ Q と パラメータ R は使用しません。

この機能は、FMAC_WDATA レジスタに N 回の書込みが行われると完了します。

18.3.6 フィルタ機能

以下のフィルタ機能が FMAC ユニットによってサポートされています。これらの機能は、START ビットをセットして、FMAC_PARAM レジスタの FUNC ビットフィールドに該当する値を書き込むことによってトリガされます。以下に示すように、P、Q、R ビットフィールドにも各機能に適したパラメータ値が含まれている必要があります。フィルタ機能の動作は、ソフトウェアによって START ビットがリセットされるまで継続します。

畳み込み（FIR フィルタ）

$\underline{\mathbf{Y}} = \underline{\mathbf{B}} * \underline{\mathbf{X}}$

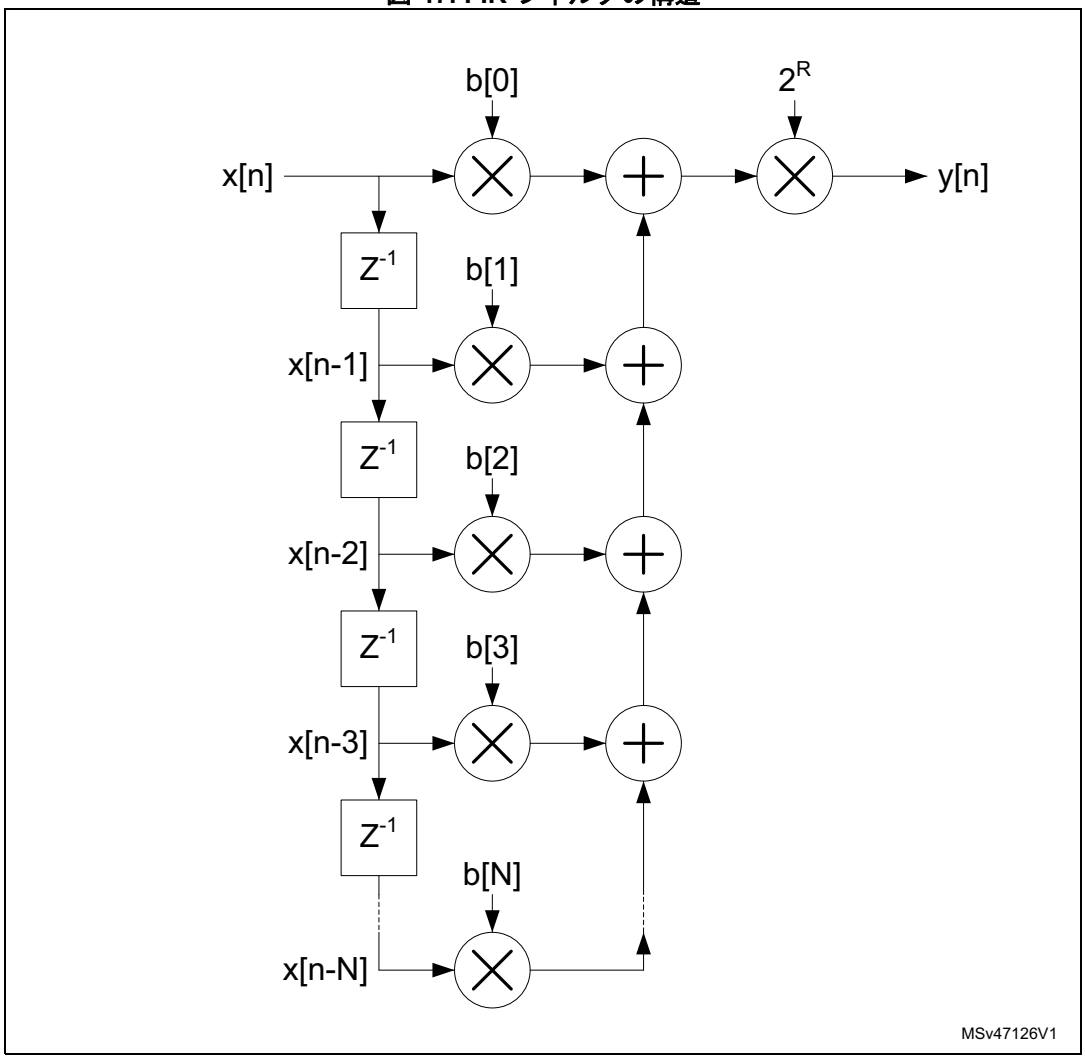
$$y_n = 2^R \cdot \sum_{k=0}^{N} b_k x_{n-k}$$

この機能は、長さが N+1 のベクトル $\underline{\mathbf{B}}$ と不定長のベクトル $\underline{\mathbf{X}}$ の畳み込みを行います。インクリメントする値 n に対する $\underline{\mathbf{Y}}$ の要素は、ドット積 $y_n = \underline{\mathbf{B}}.\underline{\mathbf{X}}_\mathbf{n}$ として計算されます。ここで $\underline{\mathbf{X}}_\mathbf{n} = [x_{n-N},...,x_n]$ は、インデックスが n - N から n までの N+1 個の X の要素から構成されます。

この機能は、有限インパルス応答（FIR）フィルタに相当するものであり、ベクトル **B** にはフィルタ係数が、ベクトル **X** にはサンプル･データが含まれています。

フィルタの構造（直接型）を 図 47 に示します。

図 47. FIR フィルタの構造



係数ベクトル **B** の次元を反対にすることにより、相互相関ベクトルの計算が可能であることに注意してください。

入力：

- X1 バッファにはベクトル **X** の要素が含まれています。長さが N + 1 + d のサーキュラ･バッファとなります。
- X2 バッファにはベクトル **B** の要素が含まれています。長さが N + 1 の固定バッファとなります。

出力：

- Y バッファには出力値 y_n が含まれています。長さが d のサーキュラバッファとなります。

パラメータ：

- パラメータ P には、[2:127] の範囲の係数ベクトル **B** の長さ N+1 が含まれています。
- パラメータ R には、アキュムレータ出力に適用されるゲインが含まれています。Y バッファに出力される値には 2^R が掛けられます。ここで、R は [0:7] の範囲です。
- パラメータ Q は使用しません。

この機能は、FMAC_PARAM レジスタの START ビットがソフトウェアによってリセットされると完了します。

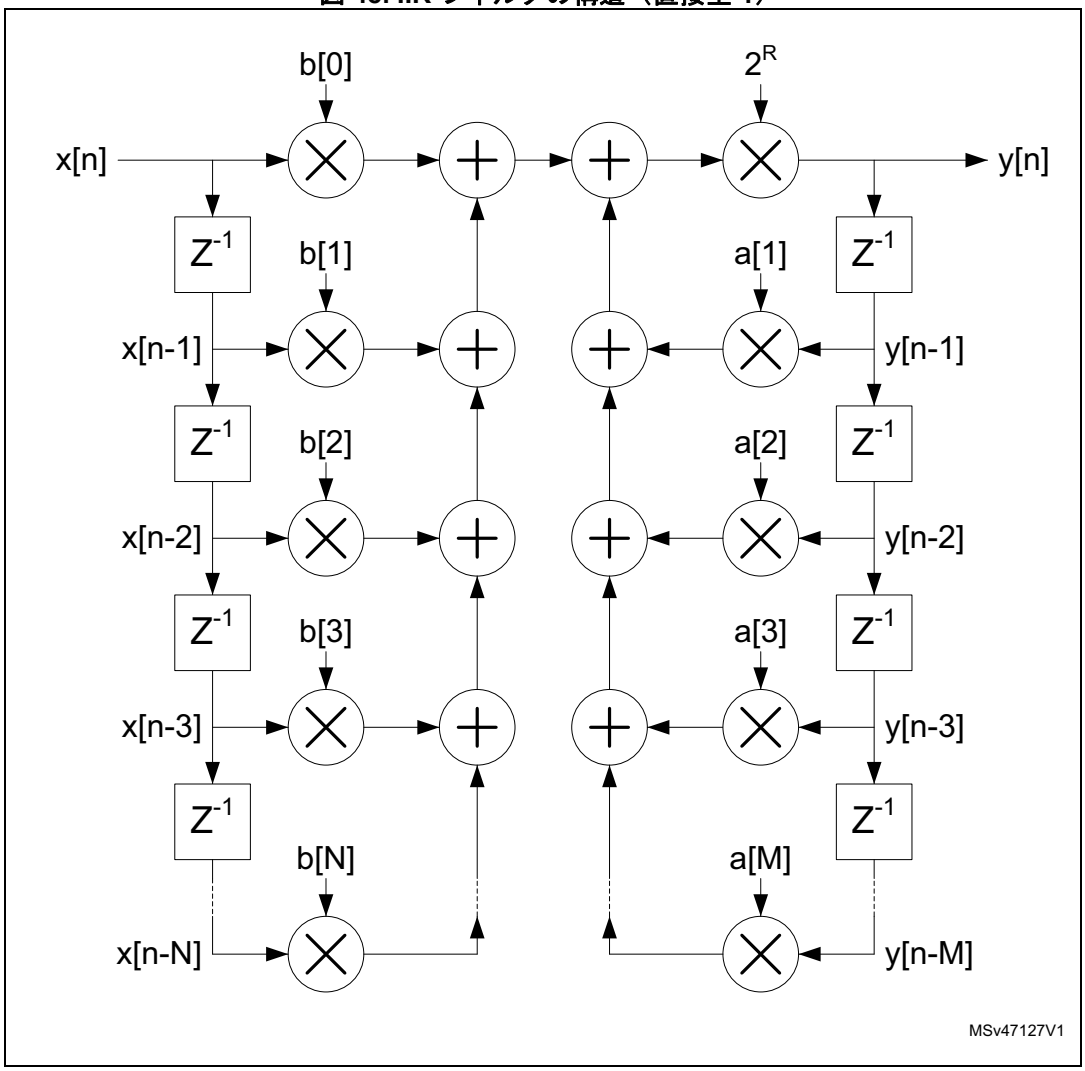
IIR フィルタ

Y = **B*****X**+**A*****Y'**

$$y_n = 2^R \cdot \left(\sum_{k=0}^{N} b_k x_{n-k} + \sum_{k=1}^{M} a_k y_{n-k} \right)$$

この機能は、無限インパルス応答（IIR）フィルタを実装します。フィルタ出力ベクトル **Y** は、長さが N+1 の係数ベクトル **B** と不定長のベクトル **X** との畳み込みと、長さが M の 2 番目の係数ベクトル **A** と遅延出力ベクトル **Y'**との畳み込みの和です。インクリメントする値 n に対する Y の要素は、y_n = **B**.$\mathbf{X_n}$ + **A**.$\mathbf{Y_{n-1}}$ として計算されます。ここで $\mathbf{X_n}$ = $[x_{n-N},...,x_n]$ は、インデックスがn - N から n までの**X**の N+1 個の要素から構成され、$\mathbf{Y_{n-1}}$ = $[y_{n-M},...,y_{n-1}]$ はインデックスが n - M から n - 1 まで M 個の **Y** の要素から構成されます。フィルタの構造（直接型 1）を 図 48 に示します。

図 48. IIR フィルタの構造（直接型 1）



入力：

- X1 バッファにはベクトル **X** の要素が含まれています。長さが N + 1 + d のサーキュラ･バッファとなります。
- X2 バッファには係数ベクトル **B** と **A** が連結された要素（b_0, b_1, b_2..., b_N, a_1, a_2, ..., a_M）が含まれています。
  長さが M + N + 1 の固定バッファとなります。

出力：

- Y バッファには出力値 y_n が含まれています。長さが M + d のサーキュラバッファとなります。

パラメータ

- パラメータ P には、[2:64] の範囲の係数ベクトル **B** の長さ N + 1 が含まれています。
- パラメータ Q には、[1:63] の範囲の係数ベクトル **A** の長さ M が含まれています。
- パラメータ R には、アキュムレータ出力に適用されるゲインが含まれています。Y バッファに出力される値には 2^R が掛けられます。ここで、R は [0:7] の範囲です。

この機能は、FMAC_PARAM レジスタの START ビットがソフトウェアによってリセットされると完了します。

18.3.7 固定小数点表現

FMAC は符号付き固定小数点整数フォーマットで動作します。入出力値は q1.15 です。

q1.15 フォーマットでは、数字は 1bit の符合部分と 15bit の小数部分（バイナリ小数位）によって表現されます。したがって、数値範囲は -1（0x8000）から$1 - 2^{-15}$（0x7FFF）となります。

アキュムレータは 26bit であり、そのうち 22bit が小数部分、4bit が整数／符合部分です（q4.22）。これによって、-8（0x2000000）から +7.99999976（0x1FFFFFF）までの範囲の部分累積和に対応可能となっています。6dB ステップで 0dB から 42dB までプログラム可能なゲインをアキュムレータの出力に適用できます。

数値範囲を超過してもアキュムレータの内容は飽和しないことに注意してください。値が +7.99999976 を上回るか -8 を下回る部分和は折り返されますが、それ以降の累積によって折り返しが取り消されるかぎり、これは無害です。ただし、FMAC_SR レジスタの SAT フラグがセットされ、FMAC_CR レジスタの SATIEN ビットがセットされている場合は、折り返しによって割込みが発生します。これはフィルタのデバッグに役立ちます。

オプションとして、FMAC_CR レジスタの CLIPEN ビットをセットすることにより、プログラム可能なゲインを適用した後に、アキュムレータによるデータ出力を飽和させることも可能です。このビットがセットされると、q1.15 の数値範囲を超えるあらゆる出力値は、符合に応じて$1 - 2^{-15}$ または -1 に設定されます。クリッピングが有効化されていない場合、ゲインを適用した後に使われていないアキュムレータのビットは単純に切り詰められます。

18.3.8 FMAC を用いた FIR フィルタの実装

FMAC は 長さが N の FIR フィルタに対応しています。ここで、N はタップまたは係数の個数です。長さが N の FIR フィルタに対するローカルメモリの最小要件は 2N + 1 です。

- 係数 N 個
- 入力サンプル N 個
- 出力サンプル 1 個

ローカルメモリのサイズは 256 ですので、N の最大値は 127 となります。

最大のスループットが必要な場合には、新しい入力サンプル待ちや出力サンプルの読出し待ちによりフィルタが停止することが絶対にないことを確実なものとするために、入力と出力のサンプルバッファそれぞれに対して、少量の追加スペース d1 と d2 を割り当てる必要があるかもしれません。この場合、ローカルメモリ要件は 2N + d1 + d2 となります。

バッファは次のように設定してください。

- X1_BUF_SIZE = N + d1
- X2_BUF_SIZE = N
- Y_BUF_SIZE = d2（追加スペースが不要な場合は 1）

バッファのベース･アドレスはどこにでも割り当て可能ですが、X2 バッファは他のものとオーバーラップさせないでください。そうしないと、係数が上書きされます。設定例を以下に示します。

- X2_BASE = 0
- X1_BASE = N
- Y_BASE = 2N + d1

ただし、メモリ空間が限られている場合には、各出力サンプルがもはや必要ではなくなった最も古い入力サンプルの場所を使用できるように、X1 バッファ領域と Y バッファ領域をオーバーラップさせることができます。

- X2_BASE = 0
- X1_BASE = N
- Y_BASE = N

この場合には Y_BUF_SIZE = X1_BUF_SIZE = N + d1 であり、バッファは同期したままとなります。

注： **X1 バッファ設定レジスタの FULL_WM ビットフィールドは、log_2(d1) 以下の値でプログラムする必要があります。そうしないと、入力サンプルが N 個書き込まれる前にバッファにフル･フラグが立ってしまい、それ以上のサンプルがリクエストされません。同様に、Y バッファ設定レジスタの EMPTY_WM ビットフィールドは、log_2(d2) 以下とする必要があります。**

フィルタ係数は、ロード X2 バッファ機能を用いて X2 バッファにプリロードする**必要があります**。オプションとして、最大 N 個までの任意の個数のサンプルを X1 バッファにプリロードできます。FIR フィルタにはフィードバック･パスが存在しないため、Y バッファのプリロードは無意味です。

バッファの設定と初期化の後、FMAC メモリにデータを読み書きする方法に従って FMAC_CR レジスタをプログラムしてください。

以下の 3 種類の方法に対応しています。

- ポーリング : DMA リクエストや割込みリクエストは生成されません。ソフトウェアは、WDATA に書き込む前に X1_FULL フラグがローであることを、RDATA から読み出す前に Y_EMPTY フラグがローであることを確認する必要があります。
- 割込み : 割込みリクエストは、書込みでは X1_FULL フラグがロー、読出しでは Y_EMPTY フラグがローである間にアサートされます。
- DMA : DMA リクエストは、X1_FULL フラグがローであるときに DMA ライト･チャネルにアサートされ、Y_EMPTY フラグがローであるときにリード･チャネルにアサートされます。

読出しと書込みには別の方法を用いることもできます。ただし、同じ操作に対して割込みと DMA リクエストの両方を使用することは推奨されません[a]。有効な組み合わせを 表 117 に示します。

表 117. 読出し手法と書込み手法の有効な組み合わせ

WIEN	RIEN	DMAWEN	DMAREN	書込み	読出し
0	0	0	0	ポーリング	ポーリング
0	1	0	0	ポーリング	割込み
1	0	0	0	割込み	ポーリング
1	1	0	0	割込み	割込み
0	0	0	1	ポーリング	DMA
0	0	1	0	DMA	ポーリング
0	0	1	1	DMA	DMA
0	1	1	0	DMA	割込み
1	0	0	1	割込み	DMA

a. 割込みと DMA リクエストの両方が有効化された場合には、DMA のみが転送を行うようにしてください。

フィルタは、FMAC_PARAM レジスタに次のビットフィールド値を書き込むことによって開始されます。

- FUNC = 8（FIR フィルタ）
- P = N（係数の個数）
- Q = 「無視」
- R = ゲイン
- START = 1

X1 バッファにプリロードされている値が N + d - 2^{FULL_WM} 個未満である場合、X1FULL フラグはローのままとなります。この状態で FMAC_CR レジスタの WIEN ビットがセットされている場合、直ちに割込みリクエストがアサートされて、追加で 2^{FULL_WM} 個のサンプルを FMAC_WDATA レジスタ経由でバッファに書き込むリクエストがプロセッサに発行されます。このリクエストは、FMAC_SR レジスタで X1FULL フラグがハイになるまでアサートされたままとなります。割込みサービス･ルーチンは、FMAC_WDATA レジスタへの 2^{FULL_WM} 回の書込みのたびに X1FULL フラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。同様に、FMAC_CR レジスタの DMAWEN ビットがセットされている場合には、X1FULL フラグがハイになるまで DMA 書込みチャネル･リクエストが生成されます。

（プリロードされたすべてのサンプルを含めて）少なくとも N 個のサンプルが X1 バッファに書き込まれると、フィルタは最初の出力サンプルの計算を行います。

2^{EMPTY_WM} 個の出力サンプルが Y バッファに書き込まれると、FMAC_SR レジスタの YEMPTY フラグがローになります。この状態で FMAC_CR レジスタの RIEN ビットがセットされている場合、割込みリクエストがアサートされて、追加で 2^{EMPTY_WM} 個のサンプルを FMAC_RDATA レジスタ経由でバッファから読み出すリクエストがプロセッサに発行されます。このリクエストは、YEMPTY フラグがハイになるまでアサートされたままとなります。割込みサービス･ルーチンは、FMAC_RDATA レジスタからの 2^{EMPTY_WM} 回の読出しのたびに YEMPTY フラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。FMAC_CR の DMAREN ビットがセットされている場合、YEMPTY フラグがハイになるまで DMA リード･チャネル･リクエストが生成されます。

ソフトウェアが START ビットをリセットして停止されるまで、フィルタはこの方法での動作を継続します。

18.3.9 FMAC を用いた IIR フィルタの実装

FMAC は 長さが N の IIR フィルタに対応しています。ここで、N はフィードフォワードタップまたは係数の個数です。フィードバック係数の個数 M は 1 から N - 1 までの任意の値とできます。直接型 1 の実装のみが実現可能であるため、それ以外の型のために設計されたフィルタは変換する必要があります。

フィードフォワード係数 N 個、フィードバック係数 M 個の IIR フィルタに対するメモリの最小要件は 2N + 2M です。

- 係数 N + M 個
- 入力サンプル N 個
- 出力サンプル M 個

M = N - 1 である場合、実装可能な最大フィルタ長は N = 64 となります。

FIR の場合、最大スループットのために、入力と出力のバッファサイズそれぞれに対して、少量の追加空間 d1 と d2 を与えるべきですので、合計のメモリ要件は 2M + 2N + d1 + d2 となります。

バッファは次のように設定してください。

- X1_BUF_SIZE = N + d1
- X2_BUF_SIZE = N + M
- Y_BUF_SIZE = M + d2

バッファのベース･アドレスはどこにでも割り当て可能ですが、オーバーラップはさせないでください。以下に設定例を示します。

- X2_BASE = 0
- X1_BASE = N + M
- Y_BASE = 2N + M + d1

注： **X1 バッファ設定レジスタの FULL_WM ビットフィールドは、log_2(d1) 以下の値でプログラムする必要があります。そうしないと、入力サンプルが N 個書き込まれる前にバッファにフル･フラグが立ってしまい、それ以上のサンプルがリクエストされません。同様に、Y バッファ設定レジスタの EMPTY_WM ビットフィールドは、log_2(d2) 以下とする必要があります。**

フィルタ係数（フィードフォワード係数 N 個の後にフィードバック係数 M 個）は、ロード X2 バッファ機能を用いて X2 バッファにプリロードする**必要があります**。オプションとして、最大 N 個までの任意の個数のサンプルを X1 バッファにプリロードできます。オプションとして、最大 M 個までの任意の個数の値を Y バッファにプリロードできます。これは、フィードバック･ディレイ･ラインを初期化する効果があります。

バッファの設定の後、FIR フィルタと同じ方法で FMAC_CR レジスタをプログラムしてください（セクション 18.3.8：FMAC を用いた FIR フィルタの実装参照）。

フィルタは、FMAC_PARAM レジスタに次のビットフィールド値を書き込むことによって開始されます。

- FUNC = 9（IIR フィルタ）
- P = N（フィードフォワード係数の個数）
- Q = M（フィードバック係数の個数）
- R = ゲイン
- START = 1

X1 バッファにプリロードされている値が N + d - 2^{FULL_WM} 個未満である場合、X1FULL フラグはローのままとなります。この状態で FMAC_CR レジスタの WIEN ビットがセットされている場合、直ちに割込みリクエストがアサートされて、追加で 2^{FULL_WM} 個のサンプルを FMAC_WDATA レジスタ経由でバッファに書き込むリクエストがプロセッサに発行されます。このリクエストは、FMAC_SR レジスタで X1FULL フラグがハイになるまでアサートされたままとなります。割込みサービス･ルーチンは、FMAC_WDATA レジスタへの 2^{FULL_WM} 回の書込みのたびに X1FULL フラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。同様に、FMAC_CR レジスタの DMAWEN ビットがセットされている場合には、X1FULL フラグがハイになるまで DMA 書込みチャネル･リクエストが生成されます。

（プリロードされたすべてのサンプルを含めて）少なくとも N 個のサンプルが X1 バッファに書き込まれると、フィルタは最初の出力サンプルの計算を行います。X1 バッファの最初の N 個のサンプルと、Y バッファの最初の M 個のサンプルを使用して（プリロードされているか否かにはかかわらず）、最初のサンプルが計算されます。最初の出力サンプルは、Y バッファの Y_BASE + M に書き込まれます。

2^{EMPTY_WM} 個の新しい出力サンプルが Y バッファに書き込まれると、FMAC_SR レジスタの YEMPTY フラグがローになります。この状態で FMAC_CR レジスタの RIEN ビットがセットされている場合、割込みリクエストがアサートされて、追加で 2^{EMPTY_WM} 個のサンプルを FMAC_RDATA レジスタ経由でバッファから読み出すリクエストがプロセッサに発行されます。このリクエストは、YEMPTY フラグがハイになるまでアサートされたままとなります。割込みサービス･ルーチンは、FMAC_RDATA レジスタからの 2^{EMPTY_WM} 回の読出しのたびに YEMPTY フラグを確認し、そのフラグがハイになるまで転送を繰り返す必要があります。FMAC_CR の DMAREN ビットがセットされている場合には、YEMPTY フラグがハイになるまで DMA リード･チャネル･リクエストが生成されます。

ソフトウェアが START ビットをリセットして停止されるまで、フィルタはこの方法での動作を継続します。

18.3.10 フィルタ初期化の例

図 49. X1 バッファの初期化

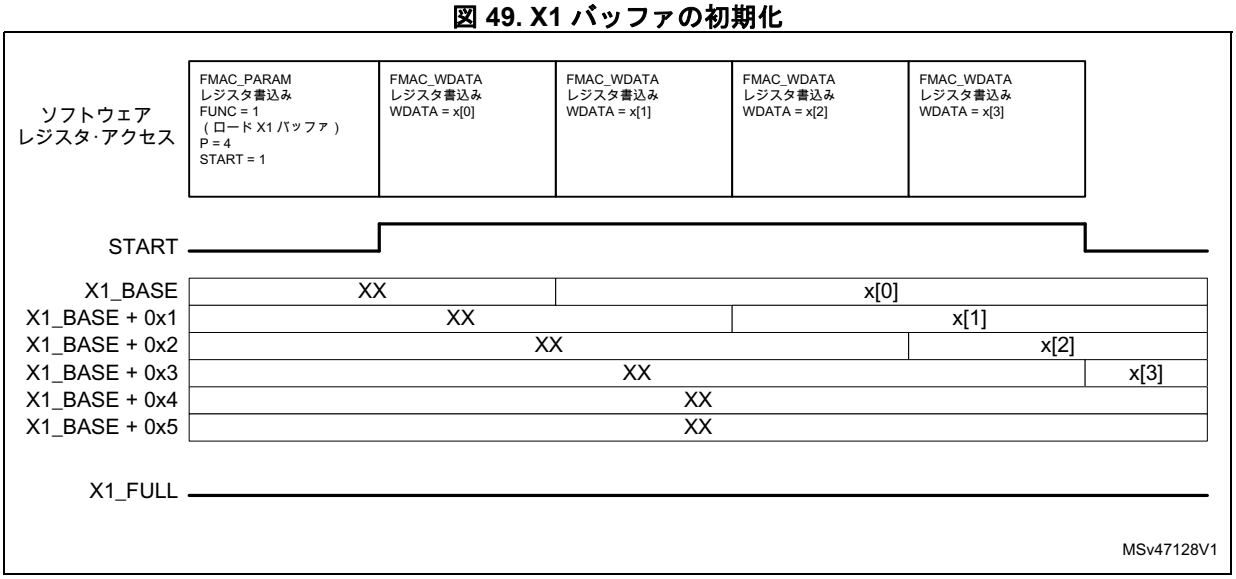


図 49 の例には、4 個のサンプル（P = 4）を用いた X1 バッファのプリロードが図示されています。バッファ·サイズは 6 です（X1_BUF_SIZE = 6）。初期化は、START ビットをセットして FMAC_PARAM レジスタをプログラムすることによって起動されます。すると、4 個のサンプルは FMAC_WDATA に書き込まれ、X1_BASE 以降のローカル·メモリに転送されます。4 番目のサンプルが書き込まれると START ビットはリセットされます。この時点で、X1 バッファには書込み順に 4 個のサンプルが格納されており、ライトポインタ（次の空き空間）は X1_BASE + 0x4 となっています。

18.3.11 フィルタ演算の例

図 50. フィルタ例 1

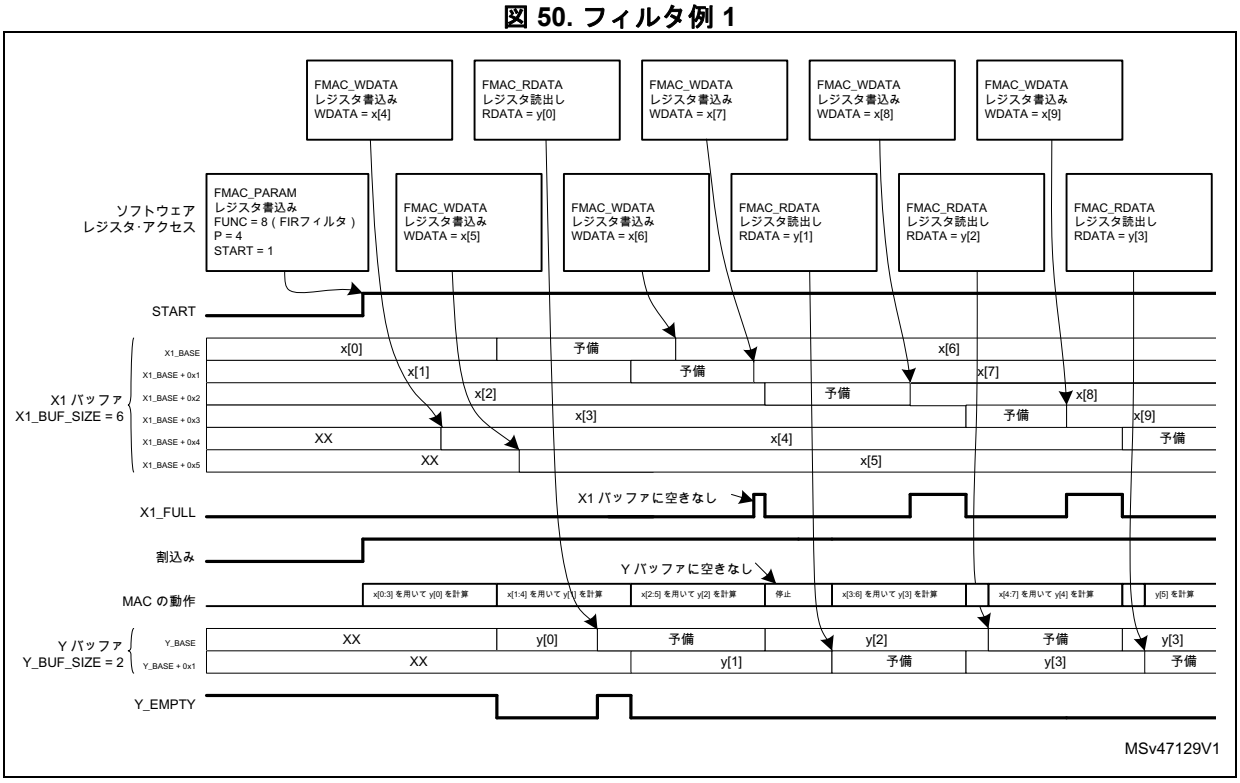


図 50 の例には、フィルタ演算の始まりが図示されています。フィルタは 4 タップです（P = 4）。X1 バッファサイズは 6、Y バッファサイズは 2 です。FULL_WM ビットフィールドと EMPTY_WM ビットフィールドはどちらも 0 に設定されています。図 49 に示すように、フィルタを開始する前に、X1 バッファには 4 個のサンプル x[0:3] がプリロードされています。したがって、START ビットがセットされると、フィルタは最初の出力サンプル y[0] の計算を直ちに開始します。X1FULL フラグはセットされていないため（X1 バッファ内の 2 つの未初期化領域のため）、直ちに割込みがアサートされ、新しいデータをリクエストします。プロセッサが 2 個の新しいサンプル x[4] および x[5] を FMAC_WDATA レジスタに書き込むと、それらのサンプルは X1 バッファの空いている位置に転送されます。

そうしている間に FMAC は最初の出力サンプル y[0] の計算を完了して Y バッファの中に書き込むことにより、Y_EMPTY フラグがローとなります。同時に、x[0] サンプルはもはや必要ないため破棄され、メモリ内のその位置（X1_BASE の位置）を開放します。必要な入力サンプル x[1:5] はすべて X1 バッファの中に存在するので、FMAC は、2番目の出力サンプル y[1] に対する処理を直ちに開始可能となります。

Y_EMPTY フラグがローであるため、プロセッサが x[5] の書込みをし終わった後も割込みはアクティブのままとなります。プロセッサが FMAC_RDATA レジスタから y[0] を読み出し、Y バッファ内のその位置を開放します。依然として y[1] の計算中であるため、出力バッファの中にサンプルがなくなり、Y_EMPTY フラグがハイになります。ただし、依然として X1 バッファの中に空き空間が存在するために割込みはアクティブのままであり、この空間には次にプロセッサが x[6] などを入れます。

注： **この例では、FMAC が処理可能な速度以上でプロセッサが入力バッファをフィルできるため、X1_full フラグが定期的にアクティブとなっています。ただし、Y バッファを十分な速度で読み出すことには苦労しており、Y バッファの空間開放待ちで定期的に FMAC が停止しています。つまり、このフィルタは、最大スループットでは動作していません。その理由は、この例では、フィルタ長が短く、プロセッサが比較的低速であるためです。したがって、Y バッファサイズを増やしても効果はありません。**

図 51. フィルタ例 2

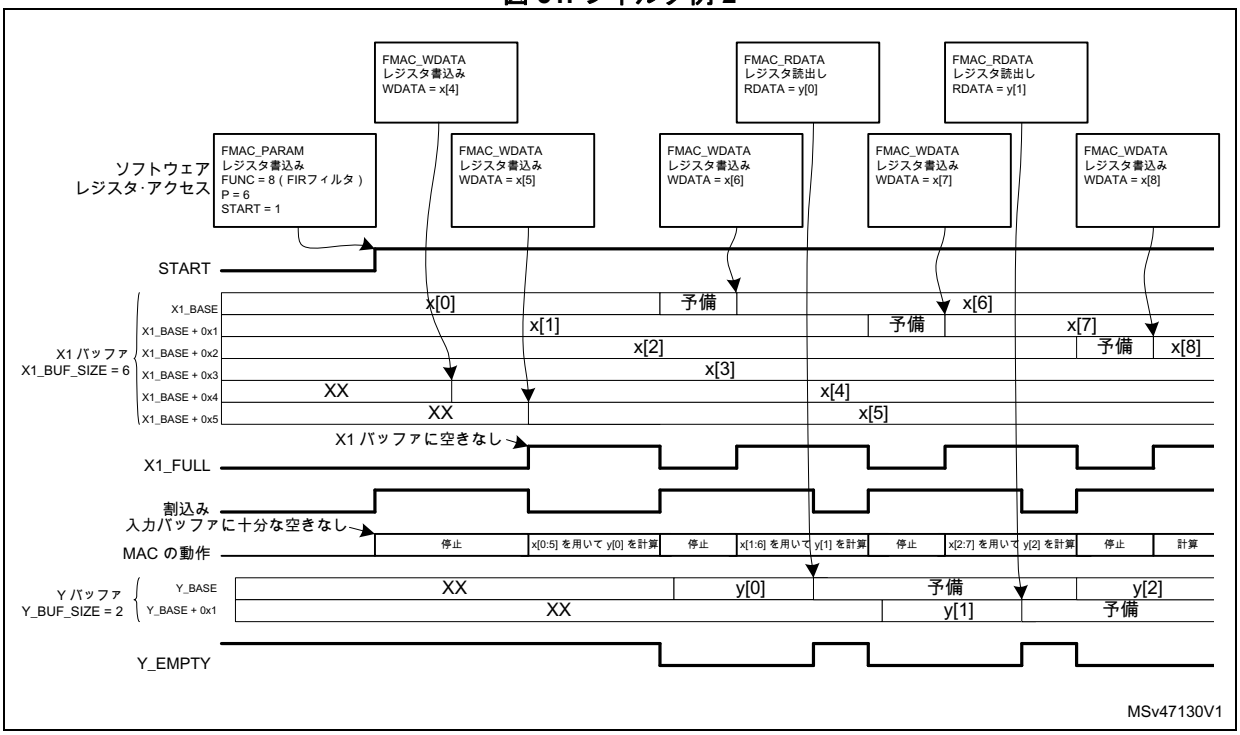


図 51 の例には、同じフィルタ演算の始まりが図示されていますが、今回のフィルタは 6 タップです（P = 6）。X1 バッファサイズは 6、Y バッファサイズは 2 です。FULL_WM ビットフィールドと EMPTY_WM ビットフィールドはどちらも 0 に設定されています。図 49 に示すように、フィルタを開始する前に、X1 バッファには 4 個のサンプル x[0:3] がプリロードされています。入力バッファに十分な数のサンプルがないことから、X1FULL フラグはセットされませんので、直ちに割込みがアサートされ、新しいデータをリクエストします。FMAC は停止します。

プロセッサが 2 個の新しいサンプル x[4] および x[5] を FMAC_WDATA レジスタに書き込むと、それらのサンプルは X1 バッファの空いている位置に転送されます。X1 バッファに未使用のサンプルが 6 個存在すると、X1_FULL フラグが直ちにアクティブとなり（バッファサイズが 6 であるため）、割り込むが無効になります。FMAC は最初の出力サンプル y[0] の計算を開始します。これには 6 個の入力サンプルがすべて必要であるため、X1 バッファには空き空間が存在せず、X1_FULL フラグはアクティブのままとなります。FMAC が y[0] の計算を完了して Y バッファに書き込んで初めて x[0] の破棄が可能となり、X1 バッファの空間が開放されて、X1_FULL がデアサートされます。同時に、Y_EMPTY フラグは無効となります。これら両方のフラグの状態によって割込みがアサートされ、最初に新しい入力サンプルの書込みを、その次に計算されたばかりの出力サンプルの読出しを、プロセッサにリクエストします。新しい入力サンプルが書き込まれるまで、FMAC は停止したままです。

この例では、プロセッサが新しい入力サンプルを書込み可能となる前に、FMAC が現在の出力サンプルの計算を完了するのを待つ必要がありますので、X1 バッファが定期的に空となり、FMAC が停止します。入力バッファに追加空間をいくらか追加することにより、これは回避可能です。

18.3.12 フィルタ設計のヒント

FMAC のアーキテクチャにより、デジタル･フィルタの設計には、以下に示すいくつかの制約条件が課されます。

1. 直接型 2 と転置型の実装は効率的ではありません。このような型のために設計されたフィルタは、直接型 1 に変換してください。
2. カスケードフィルタは、単一ステージに組み合わせるか、別のフィルタとして実装する必要があります。後者の場合には、複数セット（ステージあたり 1 セット）のフィルタ係数をメモリにプリロードしておき、X2_BASE アドレスレジスタを変更するだけで、どのセットを使用するか選択できます。複数ステージフィルタを実装する最も効率的な方法は、大きな X1 バッファに入力サンプルをプリロードしておき、最初のステージの係数を使用して IIR フィルタ機能を実行し、出力サンプルをメモリに戻して格納するというものです。次に、X2_BASE ポインタが 2 番目のステージの係数を指すように変更し、入力バッファに最初のステージの出力を再ロード（必要に応じてゲインを適用）してから IIR 機能を再実行します。すべてのステージに対して、この手順を繰り返します。最終ステージのサンプルがシステムメモリに転送されたら、入力バッファに次セットの入力サンプルをロードして、新しいラウンドの計算を開始できます。各ラウンドの間で連続性を維持するために、各ステージの N サンプル入力バッファには、最初に、前回ラウンドの N - 1 個の最後の入力をすべてプリロードしてから、新しいサンプルを 1 個加える必要があることに注意してください。同様に、同じ理由により、各ステージの出力バッファには、前回ラウンドから最後の M 個のサンプルをロードする必要があります。
3. IIR 設計に直接型 1 を用いると、たとえば入力に大きなステップが生じたり、一部のフィルタ係数の絶対値が 1 を上回ったりするような場合に、アキュムレータの中に正または負の大きな部分和が生じることがあります。アキュムレータは 26bit に制限されているため、折り返し（符合の変化）なしに処理できる最大値は、プラス側で 0x1FFFFFF、マイナス側で 0x2000000 となります。q3.23 固定小数点フォーマットでは、これらの値はそれぞれ 3.99999988 と -4 に対応します。累積の最後まで折り返しが取り消されない限り、折り返しが問題となることはありません。ただし、フィルタが開始されて予期しない結果に至る可能性がある場合には、このことは必ずしも当てはまりません。これを避けるためには、出力バッファに適切な値をプリロードすることを検討してください。
4. IIR フィルタには、フィードフォワード（分子）係数 $[b_0, b_1, ..., b_{N-1}]$ とフィードバック（分母）係数 $[1, a_1, ..., a_M]$ があります。多くの IIR フィルタでは、周波数応答において急峻なロールオフを得るために、一部の分母係数が 1 を上回る絶対値であることが要求されます。固定小数点の q1.15 フォーマットで係数がコード化されることから考えれば、これは不可能です。それにもかかわらず、$2^{-R}.[1, a_1, ..., a_M]$ がすべて 1 未満となるようにファクタ 2^{-R} によって分母係数をスケーリングすることにより、このようなフィルタについても実装可能となります。ただし、スケーリングを打ち消すために、アキュムレータの出力に対して 2^R の逆ゲインを適用する必要があります。こうすることにより、信号対雑音比（SNR）に悪影響が生じます。

18.4 FMAC レジスタ

18.4.1 FMAC X1 バッファ設定レジスタ（FMAC_X1BUFCFG）

アドレス･オフセット：0x00

リセット値：0x0000 0000

アクセス：ワード･アクセス

このレジスタは、FMAC_PARAM レジスタの START = 0 の場合にのみ変更できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FULL_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X1_BUF_SIZE [7:0]								X1_BASE [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **FULL_WM[1:0]**：バッファ･フル･フラグに対するウォーターマーク

サーキュラ･モードでの動作時に X1 バッファ･フル･フラグをセットする閾値を定義します。バッファ内の空き空間の個数が 2^{FULL_WM} 未満になると、このフラグがセットされます。

0：閾値 = 1

1：閾値 = 2

2：閾値 = 4

3：閾値 = 8

1 よりも大きい閾値を設定すると、1 回の割込みで複数のデータをバッファに転送可能となります。

DMA 書込みリクエストが有効である場合（FMAC_CR レジスタの DMAWEN = 1）、閾値は 1 にセットしてください。

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **X1_BUF_SIZE[7:0]**：X1 バッファに割り当てられたサイズ（16bit ワード単位）

最小バッファサイズは、フィルタのフィードフォワードタップ数（+ ウォーターマーク閾値 - 1）です。

ビット 7:0 **X1_BASE[7:0]**：X1 バッファのベース･アドレス

18.4.2 FMAC X2 バッファ設定レジスタ（FMAC_X2BUFCFG）

アドレス･オフセット：0x04

リセット値：0x0000 0000

アクセス：ワード･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
X2_BUF_SIZE [7:0]								X2_BASE [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **X2_BUF_SIZE[7:0]**：X2 バッファのサイズ（16bit ワード単位）
機能の動作中（START = 1）は、このビットフィールドは変更できません。

ビット 7:0 **X2_BASE[7:0]**：X2 バッファのベース･アドレス
係数値の変更などのために、START = 1 の間にも X2 バッファのベース･アドレスは変更可能です。計算中に係数を変更すると結果に影響が生じるため、これを行っている間はフィルタを停止する必要があります。

18.4.3 FMAC Y バッファ設定レジスタ（FMAC_YBUFCFG）

アドレス･オフセット：0x08

リセット値：0x0000 0000

アクセス：ワード･アクセス

このレジスタは、FMAC_PARAM レジスタの START = 0 の場合にのみ変更できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	EMPTY_WM[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw	rw								

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

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **EMPTY_WM[1:0]**：バッファ･エンプティ･フラグに対するウォーターマーク
サーキュラ･モードでの動作時に Y バッファ･エンプティ･フラグをセットする閾値を定義します。バッファ内の未読値の個数が 2^{EMPTY_WM} 未満になると、このフラグがセットされます。
0：閾値 = 1
1：閾値 = 2
2：閾値 = 4
3：閾値 = 8
1 よりも大きい閾値を設定すると、1 回の割込みで複数のデータをバッファから転送可能となります。
DMA 読出しリクエストが有効である場合（FMAC_CR レジスタの DMAREN = 1）、閾値は 1 にセットしてください。

ビット 23:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **Y_BUF_SIZE[7:0]**：Y バッファのサイズ（16bit ワード単位）
FIR フィルタでは、最小バッファ･サイズは 1（+ ウォーターマーク閾値）です。IIR フィルタでは、最小バッファサイズはフィードバックタップ数（+ ウォーターマーク閾値）です。

ビット 7:0 **Y_BASE[7:0]**：Y バッファのベース･アドレス

18.4.4 FMAC パラメータ･レジスタ（FMAC_PARAM）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

アクセス：ワード･アクセス

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

ビット 31 **START**：実行の有効化

0：実行を停止します。

1：実行を開始します。

このビットをセットすると、FUNC ビットフィールドで選択された機能の実行がトリガされます。ソフトウェアでリセットすると、実行中のあらゆる機能が停止します。初期化機能では、このビットはハードウェアによってリセットされます。

ビット 30:24 **FUNC[6:0]**：機能

0：予約済みです。

1：X1 バッファのロード

2：X2 バッファのロード

3：Y バッファのロード

4 から 7：予約済みです。

8：畳み込み（FIR フィルタ）

9：IIR フィルタ（直接型 1）

10 から 127：予約済みです。

機能の動作中（START = 1）は、このビットフィールドは変更できません。

ビット 23:16 **R[7:0]**：入力パラメータ R

このパラメータの値は機能に依存します。

機能の動作中（START = 1）は、このビットフィールドは変更できません。

ビット 15:8 **Q[7:0]**：入力パラメータ Q

このパラメータの値は機能に依存します。

機能の動作中（START = 1）は、このビットフィールドは変更できません。

ビット 7:0 **P[7:0]**：入力パラメータ P

このパラメータの値は機能に依存します。

機能の動作中（START = 1）は、このビットフィールドは変更できません。

18.4.5 FMAC 制御レジスタ（FMAC_CR）

アドレス･オフセット：0x10

リセット値：0x0000 0000

アクセス：ワード･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CLIP EN	Res.	Res.	Res.	Res.	Res.	DMA WEN	DMA REN	Res.	Res.	Res.	SAT IEN	UNFL IEN	OVFL IEN	WIEN	RIEN
rw						rw	rw				rw	rw	rw	rw	rw

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RESET**：FMAC ユニットのリセット。

ライト･ポインタとリード･ポインタ、内部制御ロジック、FMAC_SR レジスタ、FMAC_PARAM レジスタ（アクティブである場合には START ビットを含む）がリセットされます。それ以外のレジスタ設定には影響しません。このビットは、ハードウェアによってリセットされます。

0：リセットが非アクティブです。

1：リセットがアクティブです。

ビット 15 **CLIPEN**：クリッピングの有効化

0：クリッピングが無効です。q1.15 の範囲を超過したアキュムレータの出力の値は折り返されます。

1：クリッピングが有効です。q1.15 の範囲を超過したアキュムレータの出力の値は、符合に応じて正または負の最大値（+1 または -1）に飽和します。

ビット 14:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **DMAWEN**：DMA ライト･チャネル･リクエストの有効化

0：無効にします。DMA リクエストは生成されません。

1：有効にします。X1 バッファがフルでない間、DMA リクエストが生成されます。

このビットは、FMAC_PARAM レジスタの START = 0 の場合にのみ変更できます。読み出すと、ビットの現在の状態が返されます。

ビット 8 **DMAREN**：DMA リード･チャネル･リクエストの有効化

0：無効にします。DMA リクエストは生成されません。

1：有効にします。Y バッファが空ではない間、DMA リクエストが生成されます。

このビットは、FMAC_PARAM レジスタの START = 0 の場合にのみ変更できます。読み出すと、ビットの現在の状態が返されます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SATIEN**：飽和エラー割込みの有効化

0：無効。飽和が検出されても割込みは生成されません。

1：有効となっています。SAT フラグがセットされると割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 3 **UNFLIEN**：アンダーフロー･エラー割込みの有効化

0：無効。アンダーフローが検出されても割込みは生成されません。

1：有効となっています。UNFL フラグがセットされると割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 2 **OVFLIEN**：オーバーフロー･エラー割込みの有効化

0：無効。オーバーフローが検出されても割込みは生成されません。

1：有効となっています。OVFL フラグがセットされると割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 1 **WIEN**：書込み割込みの有効化

0：無効。書込み割込みリクエストは生成されません。

1：有効となっています。X1 バッファの FULL フラグがセットされていない間は、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

ビット 0 **RIEN**：読出し割込みの有効化

0：無効。読出し割込みリクエストは生成されません。

1：有効となっています。Y バッファの EMPTY フラグがセットされていない間は、割込みリクエストが生成されます。

このビットは、ソフトウェアによってセット／クリアされます。読み出すと、ビットの現在の状態が返されます。

18.4.6 FMAC ステータス･レジスタ（FMAC_SR）

アドレス･オフセット：0x14

リセット値：0x0000 0001

アクセス：ワード･アクセス

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SAT	UNFL	OVFL	Res.	Res.	Res.	Res.	Res.	Res.	X1 FULL	Y EMPTY
					r	r	r							r	r

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **SAT**：飽和エラー･フラグ。

飽和は、演算の結果がアキュムレータの数値範囲を超えたときに発生します。

0：飽和は検出されていません。

1：飽和が検出されています。SATIEN ビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット 9 **UNFL**：アンダーフロー･エラー･フラグ。

Y バッファに有効なデータが存在しないときに FMAC_RDATA から読出しを行うとアンダーフローが発生します。

0：アンダーフローは検出されていません。

1：アンダーフローが検出されています。UNFLIEN ビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット 8 **OVFL**：オーバーフロー･エラー･フラグ。

X1 バッファに空き空間が存在しないときに FMAC_WDATA への書込みを行うとオーバーフローが発生します。

0：オーバーフローは検出されていません。

1：オーバーフローが検出されています。OVFLIEN ビットがセットされている場合は、割込みが生成されます。

このフラグは、ユニットのリセットによってクリアされます。

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **X1FULL**：X1 バッファ･フル･フラグ。

利用可能な空間の数が FULL_WM 閾値未満になるとバッファのフル･フラグが立ちます。利用可能な空間の数は、ライトポインタと現在使用中のサンプルの中で最も古いものとの差となります。

0：X1 バッファはフルではありません。WIEN ビットがセットされている場合、このフラグがセットされるまで割込みリクエストがアサートされます。DMAWEN がセットされている場合、このフラグがセットされるまで DMA 書込みチャネル･リクエストが生成されます。

1：X1 バッファはフルです。

このフラグはハードウェアまたはリセットによってクリアされます。

注： X1 バッファの中で利用可能な最後の空間がフィルされた後、X1FULL フラグがハイになるまでに、3 クロック･サイクルの遅延があります。オーバーフローのリスクを回避するために、X1 バッファに書き込んでから FMAC_SR を読み出すまでの間にソフトウェアディレイを挿入することをお勧めします。その代わりに、FULL_WM 閾値を 2 とすることもできます。

ビット 0 **YEMPTY**：Y バッファ･エンプティ･フラグ。

未読データの数が EMPTY_WM 閾値未満になると、バッファのエンプティ･フラグが立ちます。未読データの数は、リードポインタと現在の出力転送先アドレスとの差となります。

0：Y バッファは空ではありません。RIEN ビットがセットされている場合、このフラグがセットされるまで割込みリクエストがアサートされます。DMAREN がセットされている場合、このフラグがセットされるまで DMA 読出しチャネル･リクエストが生成されます。

1：Y バッファは空です。

このフラグはハードウェアまたはリセットによってクリアされます。

注： Y バッファから最後のサンプルが読み出された後、YEMPTY フラグがハイになるまでに、3 クロック･サイクルの遅延があります。アンダーフローのリスクを回避するために、Y バッファから読み出してから FMAC_SR を読み出すまでの間にソフトウェアディレイを挿入することをお勧めします。その代わりに、EMPTY_WM 閾値を 2 とすることもできます。

18.4.7 FMAC ライト･データ･レジスタ（FMAC_WDATA）

アドレス･オフセット：0x18

リセット値：0x0000 0000

アクセス：ワード･アクセス

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WDATA[15:0]**：書込みデータ

このレジスタへのライト･アクセスが発生すると、ライト ポインタで示されているアドレス･オフセットにライト･データが転送されます。ポインタ･アドレスは、毎回のライト･アクセスの後に自動的にインクリメントされます。

18.4.8 FMAC 読出しデータ･レジスタ（FMAC_RDATA）

アドレス･オフセット：0x1C

リセット値：0x0000 0000

アクセス：ワード･アクセス

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RDATA[15:0]**：リード･データ

このレジスタへのリード･アクセスが発生すると、リード･データはリード ポインタで示されているアドレス･オフセットの Y 出力バッファの内容となります。ポインタアドレスは、毎回のリードアクセスの後に自動的にインクリメントされます。

18.4.9 FMAC レジスタマップ

表 118. FMAC レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMAC_X1BUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	FULL_WM [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	X1_BUF_SIZE [7:0]								X1_BASE [7:0]							
	リセット値							0	0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	FMAC_X2BUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	X2_BUF_SIZE [7:0]								X2_BASE [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	FMAC_YBUFCFG	Res.	Res.	Res.	Res.	Res.	Res.	EMPTY_WM [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Y_BUF_SIZE[7:0]								Y_BASE[7:0]							
	リセット値							0	0									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	FMAC_PARAM	START	FUNC[6:0]							R[7:0]								Q[7:0]								P[7:0]							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	FMAC_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RESET	CLIPEN	Res.	Res.	Res.	Res.	Res.	DMAWEN	DMAREN	Res.	Res.	Res.	SATIEN	UNFLIEN	OVFLIEN	WIEN	RIEN
	リセット値																0	0						0	0				0	0	0	0	0
0x14	FMAC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SAT	UNFL	OVFL	Res.	Res.	Res.	Res.	Res.	Res.	X1FULL	YEMPTY
	リセット値																						0	0	0							0	1
0x18	FMAC_WDATA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	FMAC_RDATA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2 : を参照してください。

19 フレキシブル･スタティック･メモリ･コントローラ（FSMC）

19.1 概要

フレキシブル･スタティック･メモリ･コントローラ（FSMC）には、次の 2 つのメモリ･コントローラが含まれています。

- NOR/PSRAM メモリ･コントローラ
- NAND メモリコントローラ

このメモリ･コントローラはフレキシブル･メモリ･コントローラ（FMC）とも呼ばれています。

19.2 FMC の主な特徴

FMC の機能ブロックは、同期および非同期のスタティック･メモリ、NAND Flash メモリとのインタフェースを構成します。主要目的は、

- AHB トランザクションを該当する外部デバイスプロトコルに変換すること
- 外部メモリデバイスのアクセス時間条件を満たすようにすること

すべての外部メモリが、アドレス信号、データ信号、制御信号をコントローラと共用します。各外部デバイスは、一意なチップセレクトによってアクセスされます。FMCは外部デバイスへ一度に 1 回だけアクセスします。

FMCコントローラの主要機能は次の通りです。

- 次のスタティックメモリにマップされたデバイスとのインタフェース
  - スタティックランダムアクセスメモリ（SRAM）
  - NOR 型Flashメモリ/OneNAND Flashメモリ
  - PSRAM （4 メモリバンク）
  - 強誘電性 RAM（FRAM）
  - 最大 8 KB のデータをチェックできる ECC ハードウェア付き NAND Flashメモリ
- Intel 8080 およびモトローラ 6800 モードをサポートしている並列 LCD モジュール用インタフェース
- NOR Flashメモリ、PSRAMなどの同期デバイスに対するアクセスを高速化するバーストモードのサポート
- 非同期アクセスと同期アクセス用のプログラマブルな連続クロック出力
- 8 ビット、16 ビット幅のデータバス
- メモリバンクごとに独立したチップセレクト制御
- メモリバンクごとに独立した設定
- PSRAM、SRAM デバイスで使用する書込みイネーブルとバイトレーン選択出力
- 外部非同期ウェイト制御
- 16x32 ビットの深さの書込み FIFO

書込み FIFO はすべてのメモリコントローラに共通であり、以下の構成となっています。

- ライトデータ FIFO は、メモリへ書き込む AHB データ（最大 32 ビット）と、AHB 転送（バーストモードまたは非シーケンシャルモード）用の 1 ビットを格納
- 書込みアドレス FIFO は、AHB アドレス（最大 28 ビット）と AHB データ･サイズ（最大 2 ビット）を格納バーストモードで動作する場合、ページ境界（PSRAM）を跨ぐとき以外は、開始アドレスのみを格納この場合、AHBバーストは 2 つの FIFO エントリに分割されます。

開始時に、ユーザアプリケーションから FMC ピンを設定する必要があります。アプリケーションで使用しない FMC I/O ピンは、他の用途に使用することができます。

外部デバイスタイプおよびその特性を定義する FMC レジスタは、通常、起動時に設定され、次のリセットまたは電源投入まで変更されません。ただし、設定値は何時でも変更することができます。

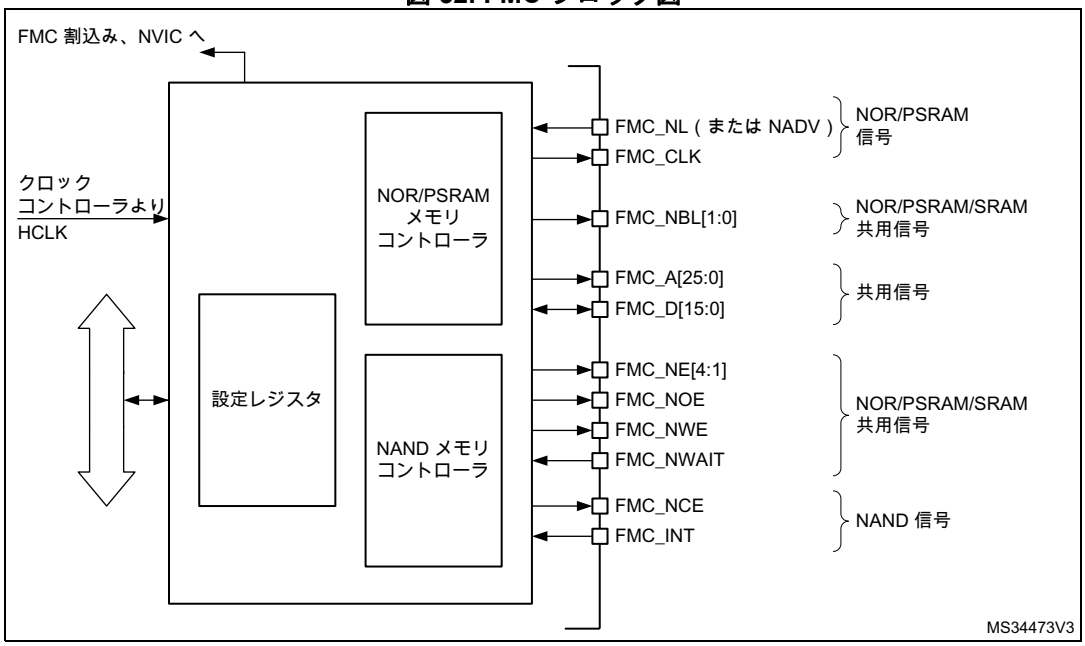
19.3 FMC ブロック図

FMC は 次のメインブロックで構成されます。

- AHB インタフェース（FMC 設定レジスタを含む）
- NOR Flash/PSRAM/SRAM コントローラ

以下にブロック図を示します。

図 52. FMC ブロック図



19.4 AHB インタフェース

AHB スレーブインタフェースを使用することで、内部 CPU やその他のバスマスタペリフェラルから外部メモリへアクセスすることができます。

AHB トランザクションは、外部デバイス･プロトコルに変換されます。特に、選択された外部メモリが16 または 8 ビット幅の場合、AHB 上の 32 ビット幅のトランザクションは、16 または 8 ビットの連続した複数のアクセスに分割されます。FMCチップセレクト（FMC_NEx）は、拡張モードが有効になっている際にアクセスモード D の場合を除いて、複数の連続アクセス間にトグルしません。

FMC は、以下の条件下で AHB エラーを生成します。

- 有効化してない FMC バンク（バンク 1 ~ 4）の読出しまたは書込みを行ったとき。
- FMC_BCRx レジスタの FACCEN ビットがリセットされている状態で NOR Flash バンクの読出しまたは書込みを行ったとき。

この AHB エラーの影響は、読出し／書込みアクセスを試みた AHB マスタに依存します。

- アクセスが FPU 搭載 Cortex®-M4 CPU から行われた場合は､ハードフォールト割込みが発生します。
- アクセスが DMA コントローラから行われた場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効化されます。

AHB クロック（HCLK）は、FMC のリファレンスクロックです。

19.4.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

リクエストされる AHB トランザクションのデータサイズは、アクセスされる外部デバイスのデータ幅が固定であっても､8､16､または 32 ビット幅です｡このため､転送に矛盾が生じることがあります。

したがって、次のようないくつかの単純なトランザクション規則を守る必要があります。

- AHB トランザクションサイズとメモリデータサイズが等しい場合：

  この場合は問題ありません。

- AHB トランザクションサイズがメモリサイズより大きい場合：

  この場合、FMC は、AHB トランザクションを外部データ幅に合わせて、連続した小さなメモリアクセスに分割します。FMC チップセレクト（FMC_NEx）は、複数の連続アクセス間にトグルしません｡バスターンアラウンドタイミングが 0 以外の値に設定されている場合は､FMC チップセレクト（FMC_NEx）は複数の連続アクセス間にトグルします。この機能は、FRAM メモリとインタフェースするときに必要です。

- AHB トランザクションサイズがメモリサイズより小さい場合：

  外部デバイスのタイプによっては、転送に一貫性がなくなる場合があります：

  – バイト選択機能を持つデバイス（SRAM、ROM、PSRAM）に対するアクセス

    この場合、FMC は、読出し／書込みトランザクションを許可して、バイトレーン NBL[1:0] を通じて正しいデータにアクセスします。

    書込み対象バイトは NBL[1:0] によりアドレス指定されます。

    すべてのメモリバイトが読み出され（読出しトランザクション中 NBL[1:0] はローに駆動されます）、不要なバイトは無視されます。

  – バイト選択機能を持たないデバイス（NOR および NAND Flashメモリ）に対するアクセス

    この状況は、16 ビット幅のFlashメモリへのバイトアクセスが要求されたときに発生します。バイトモードでデバイスにアクセスできないため（Flashメモリに対しては 16 ビットワードの読出し／書込みのみが可能）、書込みトランザクションと読出しトランザクションが可能です（コントローラは 16 ビットメモリワード全体を読み出して、必要なバイトのみを使用）。

NOR 型Flash/PSRAM のラップサポート

同期メモリに対するラップバーストモードはサポートされていません。メモリは、不定長のリニアバーストモードに設定する必要があります。

設定レジスタ

FMC は、レジスタのセットとして設定することができます。NOR Flash／PSRAM コントローラレジスタの詳細については、セクション 19.6.6 : を参照してください。NAND Flash レジスタの詳細については、セクション 19.7.7 : を参照してください。

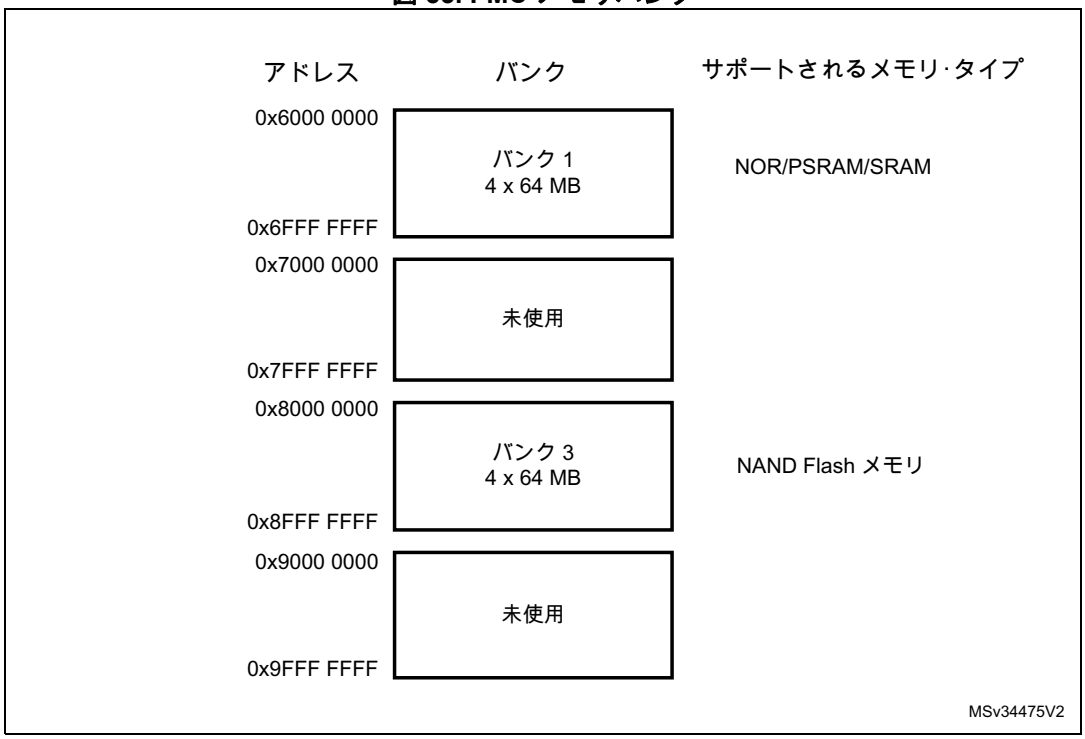
19.5 外部デバイスアドレスマッピング

FMC から見ると、外部メモリは、それぞれ 256 MB の固定サイズのバンクに分けられています（図 53 を参照）。

- バンク 1 は、4 つまでの NOR Flashメモリまたは PSRAM メモリデバイスのアドレス指定に使用されます。このバンクは、次のように 4 つの専用チップセレクトを持つ 4 つの NOR/PSRAM サブバンクに分割されます。
  - バンク 1 - NOR/PSRAM 1
  - バンク 1 - NOR/PSRAM 2
  - バンク 1 - NOR/PSRAM 3
  - バンク 1 - NOR/PSRAM 4
- バンク 3 は、NAND Flashメモリデバイスのアドレス指定に使用されます。この空間の MPU メモリ属性をソフトウェアによってデバイスに再設定する必要があります。

各バンクで使用されるメモリのタイプは、ユーザアプリケーションから設定レジスタを使って設定することができます。

図 53. FMC メモリバンク



19.5.1 NOR/PSRAM アドレスマッピング

HADDR[27:26] ビットは、表 119 に示される 4 つのメモリバンクの 1 つを選択するために使用されます。

表 119. NOR/PSRAM バンク選択

HADDR[27:26](1)	選択されるバンク
00	バンク 1 - NOR/PSRAM 1
01	バンク 1 - NOR/PSRAM 2
10	バンク 1 - NOR/PSRAM 3
11	バンク 1 - NOR/PSRAM 4

1. HADDR は、外部メモリに変換される内部 AHB アドレス･ラインです。

HADDR[25:0] ビットは、外部メモリアドレスは相当します。HADDR はバイト･アドレスですが、メモリはワード･レベルでアドレス指定されるので、メモリに対して実際に発行されるアドレスは、次の表に示されるように、メモリのデータ幅に応じて変わります。

表 120. NOR/PSRAM 外部メモリアドレス

メモリ幅(1)	メモリに発行されるデータアドレス	最大メモリ容量（ビット）
8 ビット	HADDR[25:0]	64 MB * 8 = 512 Mbit
16 ビット	HADDR[25:1] >> 1	64 MB/2 * 16 = 512 Mbit

1. 外部メモリが 16 ビット幅の場合、FMC は、内部で HADDR[25:1] を使用して、外部メモリ FMC_A[24:0] に対するアドレスを生成します。
外部メモリの幅にかかわらず、FMC_A[0] は外部メモリ･アドレス A[0] に接続されます。

19.5.2 NAND Flashメモリ･アドレス･マッピング

NAND バンクは、表 121 に示されるメモリ領域に分割されます。

表 121. NAND メモリマッピングおよびタイミングレジスタ

開始アドレス	終了アドレス	FMC バンク	メモリ空間	タイミングレジスタ
0x8800 0000	0x8BFF FFFF	バンク 3 - NAND Flash	属性	FMC_PATT（0x8C）
0x8000 0000	0x83FF FFFF		共通	FMC_PMEM（0x88）

NAND 型Flashメモリでは、共通および属性メモリ空間は、下位 256 KB に位置する 3 つのセクションに分割されます（表 122 を参照）。

- データセクション（共通/属性メモリ空間の最初の 64 KB）
- コマンドセクション（共通/属性メモリ空間の 2 番目の 64 KB）
- アドレスセクション（共通/属性メモリ空間の次の 128 KB）

表 122. NAND バンク選択

セクション名	HADDR[17:16]	アドレス範囲
アドレスセクション	1X	0x020000-0x03FFFF
コマンドセクション	01	0x010000-0x01FFFF
データセクション	00	0x000000-0x0FFFF

アプリケーションソフトウェアは、3 つのセクションを使用して、NAND 型Flashメモリにアクセスします。

- **NAND Flashメモリにコマンドを送信するときは、**ソフトウェアからコマンドセクションの任意のメモリ位置にコマンドの値を書き込む必要があります。
- **読出しまたは書込みを行う必要がある NAND Flashのアドレスを指定するには、**ソフトウェアは、アドレスセクションの任意のメモリ位置にアドレス値を書き込む必要があります。アドレスは 4 または 5 バイト長なので（実際のメモリサイズに依存）、アドレス全体を指定するには、アドレスセクションへのいくつかの連続した書込みが必要です。
- **データの読出しまたは書込みを行うときは、**ソフトウェアは、データセクションの任意のメモリ位置からデータを読み出すか、書き込む必要があります。

NAND 型Flashメモリはアドレスを自動的にインクリメントするので、連続したメモリ位置にアクセスするには、データセクションのアドレスをインクリメントする必要はありません。

19.6 NOR 型Flash／PSRAM コントローラ

FMC は、以下のタイプのメモリを駆動するのに適した信号タイミングを発生します。

- 非同期 SRAM、FRAM、および ROM
  - 8 ビット
  - 16 ビット
- PSRAM（CellularRAM™）
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス
- NOR Flashメモリ
  - 非同期モード
  - 同期アクセスに対するバーストモード
  - マルチプレクスまたは非マルチプレクス

FMC は、バンクごとに 1 つのチップセレクト信号 NE[4:1] を出力します。その他の信号（アドレス、データ、および制御）はすべて、共有されます。

FMC は、次のようなプログラム可能なタイミングにより、広範囲なデバイスをサポートしています。

- プログラム可能なウェイトステート（15 まで）
- プログラム可能なバスターンアラウンドサイクル（15 まで）
- プログラム可能な出力イネーブルおよび書込みイネーブル遅延（15 まで）
- 独立した読出しおよび書込みタイミングとプロトコルにより､広範囲なメモリおよびタイミングをサポート
- プログラム可能な連続クロック（FMC_CLK）出力

FMC クロック（FMC_CLK）は HCLK クロックの約数です。このクロックは、同期アクセス中にのみ、または FMC_BCR1 レジスタの CCKEN ビット設定に応じて非同期アクセスおよび同期アクセス中に、選択した外部デバイスへ供給することができます。

- CCLKEN ビットがリセットされている場合、FMC は同期アクセス（読出し／書込みトランザクション）中にのみクロック（CLK）を生成します。
- CCLKEN ビットがセットされている場合､FMC は非同期アクセス中および同期アクセス中に連続クロックを生成します。FMC_CLK 連続クロックを生成するためには、バンク 1 を同期モードに設定する必要があります(セクション 19.6.6:NOR/PSRAM コントローラレジスタを参照)。すべての同期メモリに対して同じクロックが使用されるため､連続出力クロックを生成して同期アクセスを実行する場合、AHB データ･サイズをメモリ･データ幅（MWID）と同じ値にする必要があります。そうしないと、AHB データ･トランザクションに応じて FMC_CLK 周波数が変化します(FMC_CLK 分周比の式については､セクション 19.6.5:同期トランザクションを参照)。

各バンクのサイズは 64 MB に固定されます。各バンクは、専用レジスタによって設定されます（セクション 19.6.6 : NOR/PSRAM コントローラレジスタを参照)。

プログラム可能なメモリパラメータは、アクセスタイミング（表 123 を参照）を含み、ウェイト管理をサポートします（バーストモードでの PSRAM および NOR Flashへのアクセス)。

表 123. プログラム可能な NOR/PSRAM のアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	AHB クロックサイクル（HCLK）	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクス I/O	AHB クロックサイクル（HCLK）	1	15
NBL セットアップ	バイトレーンセットアップフェーズ時間	非同期	AHB クロックサイクル（HCLK）	0	3
データセットアップ	データセットアップフェーズ時間	非同期	AHB クロックサイクル（HCLK）	1	256
データホールド	データホールドフェーズ時間	非同期	AHB クロックサイクル（HCLK）	0	3
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読出し／書込み	AHB クロックサイクル（HCLK）	0	15
クロック分周比	1 つのメモリクロックサイクル（CLK）に必要な AHB クロックサイクル（HCLK）数	同期	AHB クロックサイクル（HCLK）	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル（CLK）	2	17

19.6.1 外部メモリインタフェース信号

表 124、表 125 および表 126 に、NOR Flashメモリ、SRAM、および PSRAM へのインタフェースで一般に使用される信号を示します。

注： **接頭辞「N」は、アクティブローの信号を表します。**

NOR Flashメモリ、非マルチプレクス I/O

表 124. 非マルチプレクス I/O NOR Flashメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型 Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR 型Flashウェイト入力信号

最大容量は、512 M ビット（26 アドレスライン）です。

NOR Flashメモリ、16 ビットマルチプレクス I/O

表 125. 16 ビットマルチプレクス I/O NOR Flashメモリ

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス（16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	ラッチイネーブル（この信号は、一部の NOR 型Flashデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FMC への NOR 型Flashウェイト入力信号

最大容量は 512 M ビットです。

PSRAM/FRAM/SRAM、非マルチプレクス I/O

表 126. 非マルチプレクス I/O PSRAM/SRAM

FMC 信号名	I/O	機能
CLK	O	クロック （PSRAM 同期アクセス専用）
A[25:0]	O	アドレスバス
D[15:0]	I/O	双方向データバス
NE[x]	O	チップセレクト、x = 1..4 （PSRAM（CellularRAM™、すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	PSRAM 入力の場合のみアドレス有効（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は 512 M ビットです。

PSRAM、16 ビットマルチプレクス I/O

表 127. 16 ビットマルチプレクス I/O PSRAM

FMC 信号名	I/O	機能
CLK	O	クロック （同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス／データバス（16 ビットアドレス A[15:0] とデータ D[15:0] がデータバスでマルチプレクスされています）
NE[x]	O	チップセレクト、x = 1..4 （PSRAM（CellularRAM™、すなわち CRAM）では NCE と呼ばれます。）
NOE	O	出力イネーブル
NWE	O	書込みイネーブル
NL(= NADV)	O	アドレス有効 PSRAM 入力（メモリ信号名：NADV）
NWAIT	I	FMC への PSRAM ウェイト入力信号
NBL[1:0]	O	バイトレーン出力。バイト 0 およびバイト 1 制御（上位および下位バイトイネーブル）

最大容量は、512 M ビット（26 アドレスライン）です。

19.6.2 サポートされるメモリおよびトランザクション

以下の表 128 に、NOR Flashメモリ、PSRAM、SRAM に対してメモリデータバスが 16 ビット幅の場合にサポートされているデバイス、アクセスモード、トランザクションの例を示します。FMC では許容されない（すなわち非サポート）トランザクションは、この例で灰色表示しています。

表 128. NOR Flash/PSRAM：サポートされているメモリおよびトランザクションの例

デバイス	モード	読出し／書込み	AHB データサイズ	メモリデータサイズ	許可/禁止	コメント
NOR 型Flash（マルチプレクス I/O と非マルチプレクス I/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-

表 128. NOR Flash/PSRAM：サポートされているメモリおよびトランザクションの例 （続き）

デバイス	モード	読出し／書込み	AHB データサイズ	メモリデータサイズ	許可/禁止	コメント
PSRAM（マルチプレクス I/O および非マルチプレクス I/O）	非同期	R	8	16	Y	-
	非同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	-
	同期	R	16	16	Y	-
	同期	R	32	16	Y	-
	同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	同期	W	16/32	16	Y	-
SRAM と ROM	非同期	R	8/16	16	Y	-
	非同期	W	8/16	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割、バイトレーン NBL[1:0] の使用

19.6.3 一般的なタイミング規則

信号の同期

- すべてのコントローラ出力信号は、内部クロック（HCLK）の立ち上がりエッジで変化します。
- 同期モード（読出しまたは書込み）では、すべての出力信号が HCLK の立ち上がりエッジで変化します。CLKDIV の値によらず、すべての信号は次のように変化します。
  - NOEL/NWEL/ NEL/NADVL/ NADVH /NBLL/ アドレスの有効出力は､FMC_CLK クロックの立ち下がりエッジで変化します。
  - NOEH/ NWEH / NEH/ NOEH/NBLH/ アドレスの無効出力は、FMC_CLK クロックの立ち上がりエッジで変化します。

19.6.4 NOR Flash/PSRAM コントローラ非同期トランザクション

非同期スタティックメモリ（NOR Flash、PSRAM、SRAM、FRAM）

- 信号は、内部クロック HCLK によって同期されます。このクロックはメモリには発行されません。
- FMC は、NOE 信号をネゲートする前に、常にデータをサンプリングします。これにより、メモリのデータホールドタイミングに関する制約が満たされます（チップイネーブルハイからデータ遷移までの最小時間は通常 0 ns です）。
- 拡張モードを有効にした場合（FMC_BCRx レジスタの EXTMOD ビットをセット）、最大 4 つの拡張モード（A、B、C、D）が使用できます。書込み動作および読出し動作で、A、B、C、D モードをミックスすることができます。例えば、読出し動作をモード A で実行し、書込み動作をモード B で実行することができます。
- 拡張モードを無効にした場合（FMC_BCRx レジスタの EXTMOD ビットをリセット）、FMC は次のようにモード 1 またはモード 2 で動作できます。
  - SRAM/PSRAM メモリ･タイプを選択した場合（FMC_BCRx レジスタの MTYP = 0x0 または 0x01）、モード 1 がデフォルト･モードです。
  - NOR メモリタイプを選択した場合（FMC_BCRx レジスタで MTYP = 0x10）、モード 2 がデフォルトモードです。

モード 1 - SRAM/FRAM/PSRAM（CRAM）

次の図に、サポートされているモードに対する読出しトランザクションと書込みトランザクションを示します。それに続いて、FMC_BCRx レジスタ、および FMC_BTRx/FMC_BWTRx レジスタの必要とされる設定を示します。

図 54. モード 1読出しアクセス波形

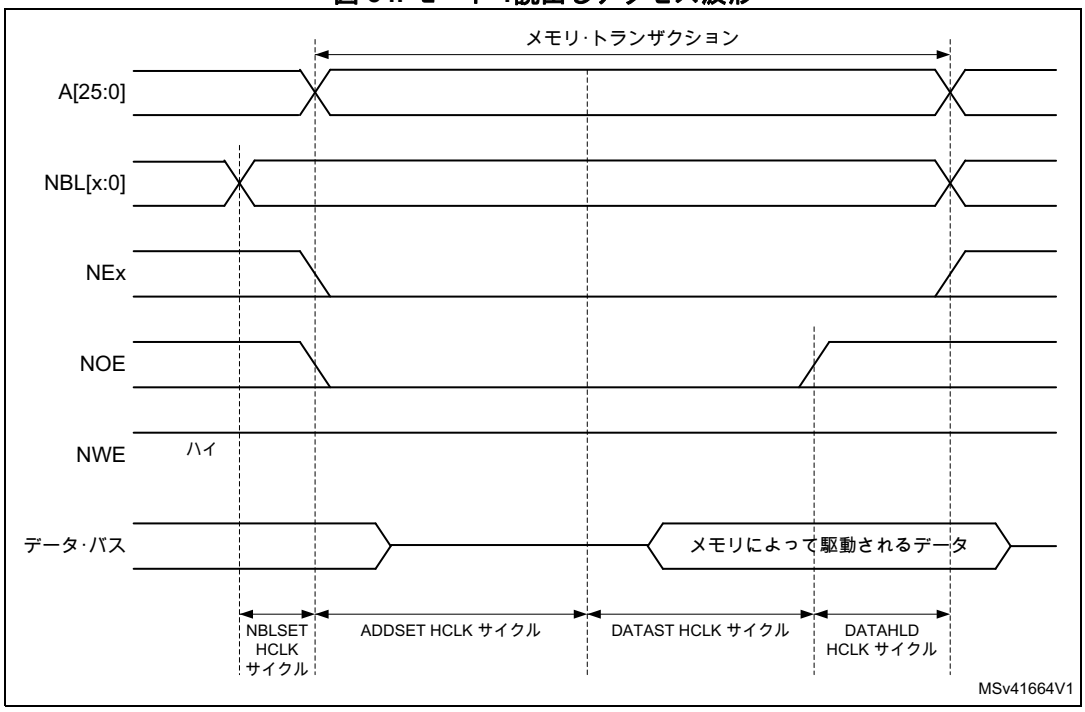
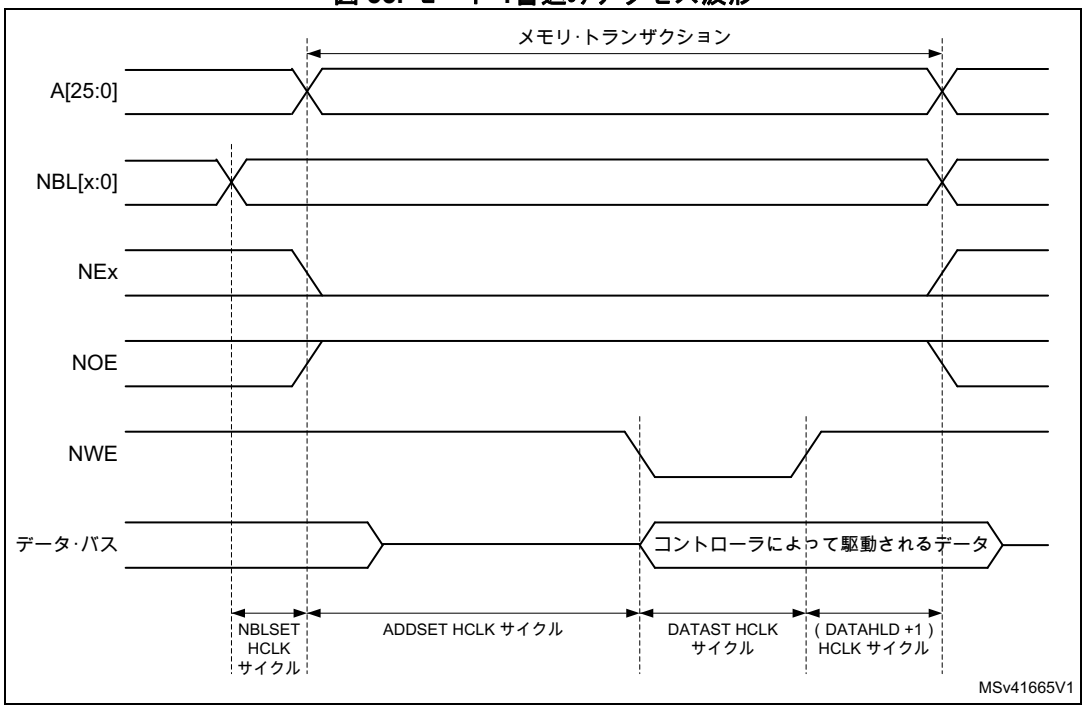


図 55. モード 1書込みアクセス波形



読出しおよび書込みトランザクションの最後の DATAHLD 時間は、NOE/NWE の立ち上がりエッジ後のアドレスおよびデータ･ホールド時間を保証します。DATAST の値はゼロより大きくなければなりません（DATAST ＞ 0）。

表 129. FMC_BCRx ビットフィールド（モード 1）

ビット番号	ビット名	設定値
31:24	予約済み	0x000
23:22	NBLSET[1:0]	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。0x2（NOR Flashメモリ）を除きます。

表 129. FMC_BCRx ビットフィールド（モード 1）（続き）

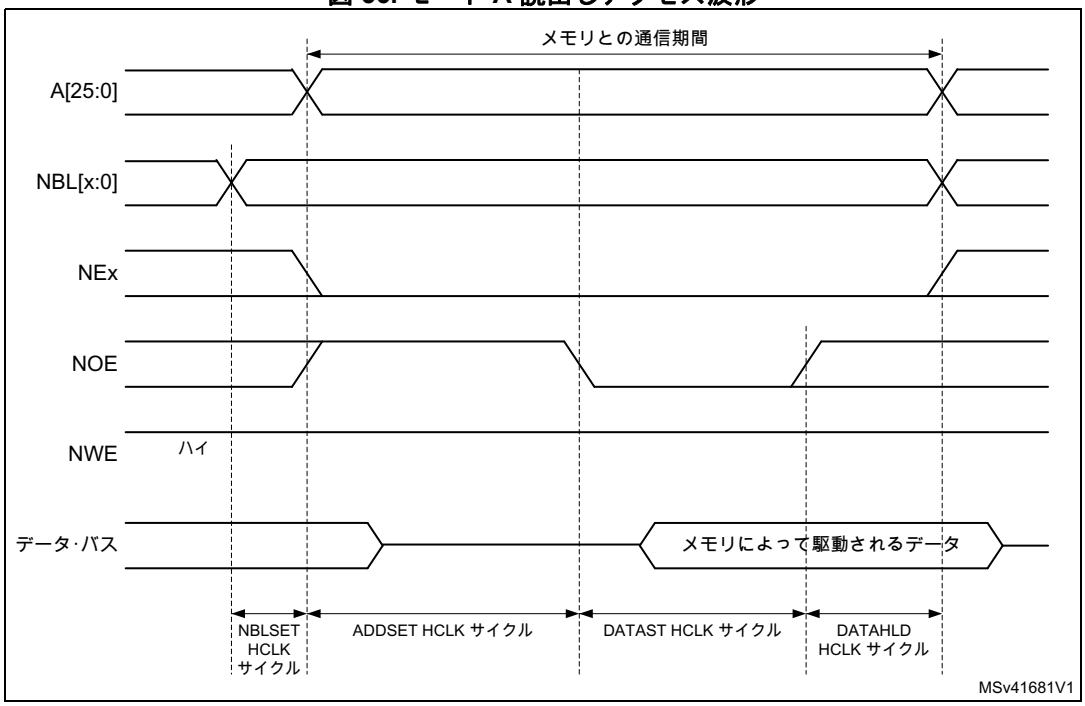
ビット番号	ビット名	設定値
1	MUXE	0x0
0	MBKEN	0x1

表 130. FMC_BTRx ビットフィールド（モード 1）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合は DATAHLD HCLK サイクル、書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	無視
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0。

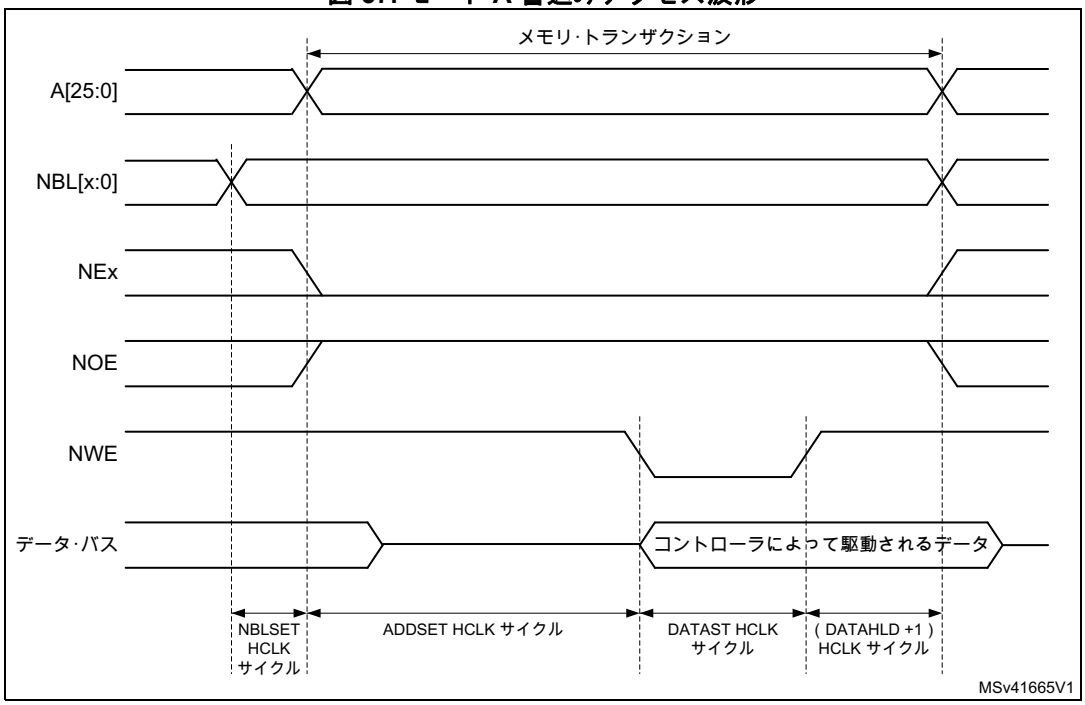
モード A - SRAM/FRAM/PSRAM（CRAM）OE トグル

図 56. モード A 読出しアクセス波形



1. NBL[1:0] は読出しアクセス中ロー･レベルに駆動されます。

図 57. モード A 書込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグルと、独立した読出しおよび書込みのタイミングです。

表 131. FMC_BCRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31:24	予約済み	0x000
23:22	NBLSET[1:0]	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	無視
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。0x2（NOR Flashメモリ）を除きます。

表 131. FMC_BCRx ビットフィールド（モード A）（続き）

ビット番号	ビット名	設定値
1	MUXEN	0x0
0	MBKEN	0x1

表 132. FMC_BTRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合は DATAHLD HCLK サイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0。

表 133. FMC_BWTRx ビットフィールド（モード A）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0。

モード 2/B - NOR 型Flash

図 58. モード 2 およびモード B 読出しアクセス波形

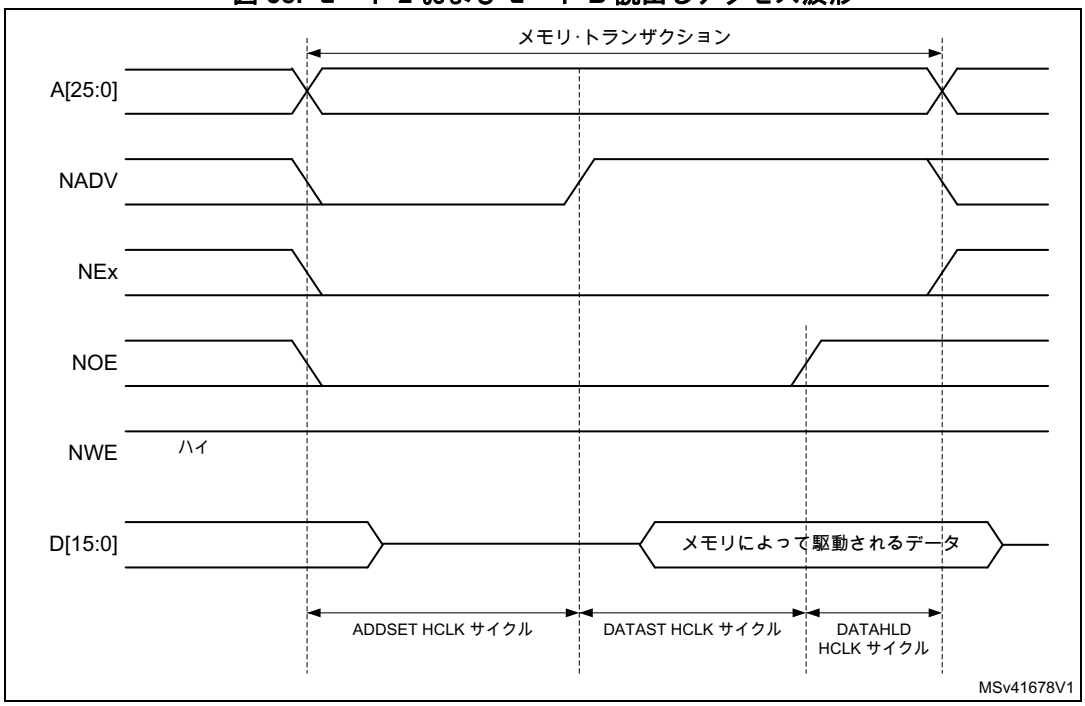


図 59. モード 2 書込みアクセス波形

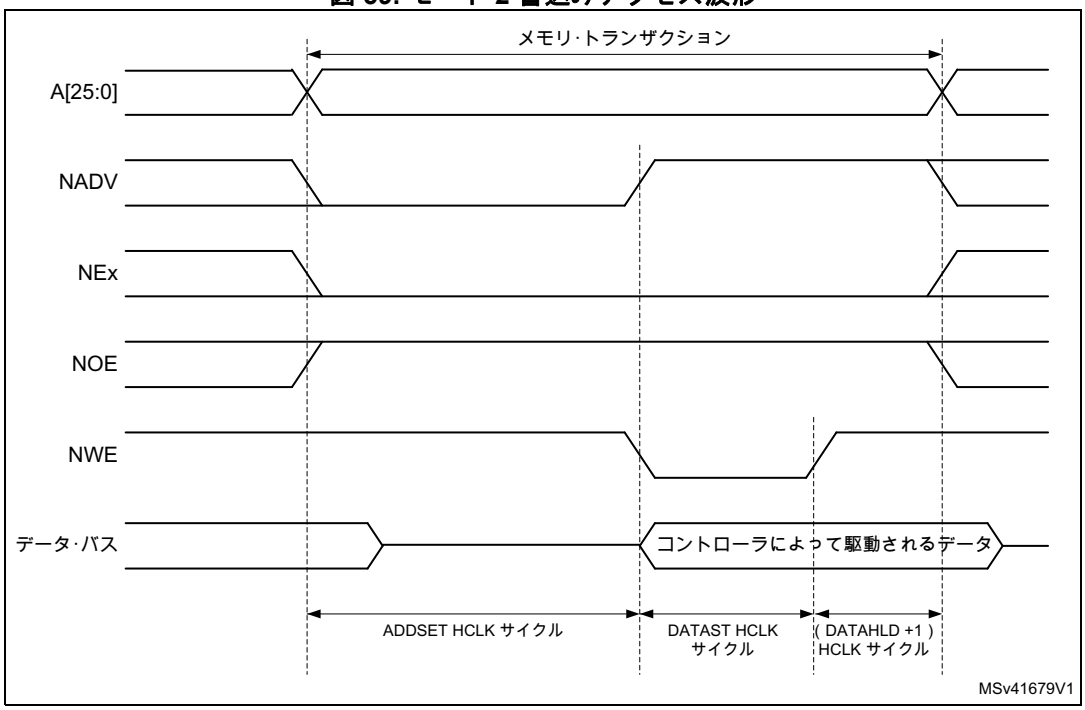
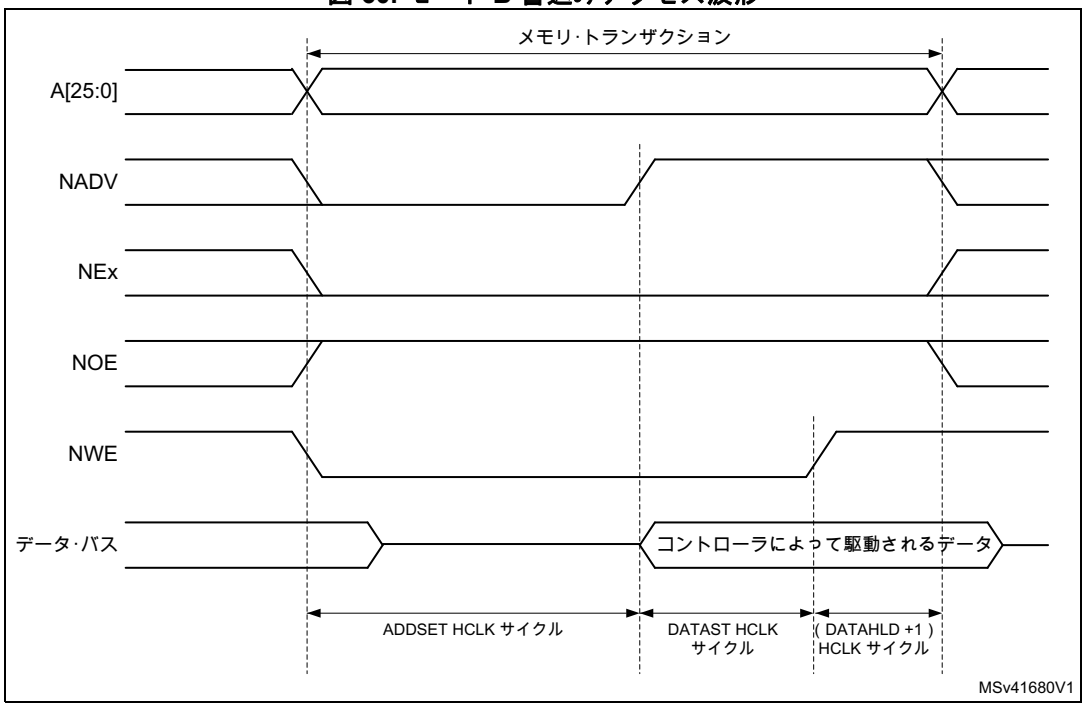


図 60. モード B 書込みアクセス波形



モード 1 と比較した場合の相違点は、NWE のトグルと拡張モードがセットされたとき（モード B）の独立した読出しおよび書込みのタイミングです。

表 134. FMC_BCRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31:24	予約済み	0x000
23:22	NBLSET[1:0]	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2 （NOR 型 Flash メモリ）

表 134. FMC_BCRx ビットフィールド（モード 2/B）（続き）

ビット番号	ビット名	設定値
1	MUXEN	0x0
0	MBKEN	0x1

表 135. FMC_BTRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（拡張モードが無効のとき、読出しアクセスの場合は DATAHLD HCLK サイクル、書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

表 136. FMC_BWTRx ビットフィールド（モード 2/B）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	拡張モードがセットされている場合は 0x1
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

注： **FMC_BWTRx レジスタは、拡張モードがセットされている場合（モード B）のみ有効です。それ以外の場合には、その内容は意味を持ちません。**

モード C - NOR Flash - OE トグリング

図 61. モード C 読出しアクセス波形

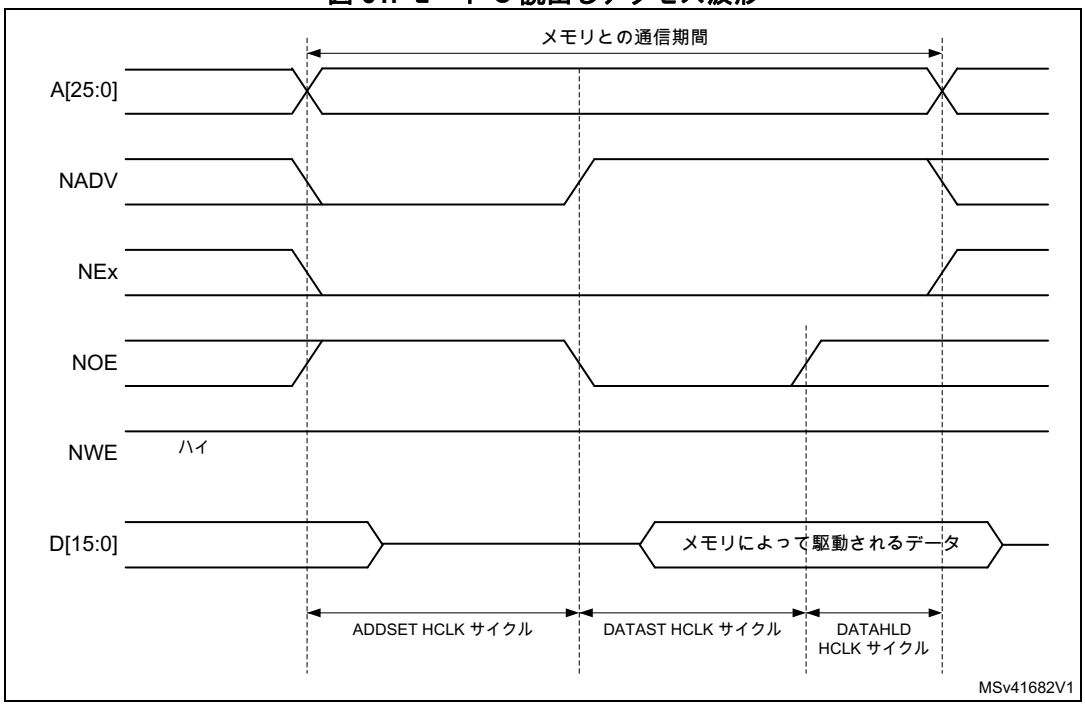
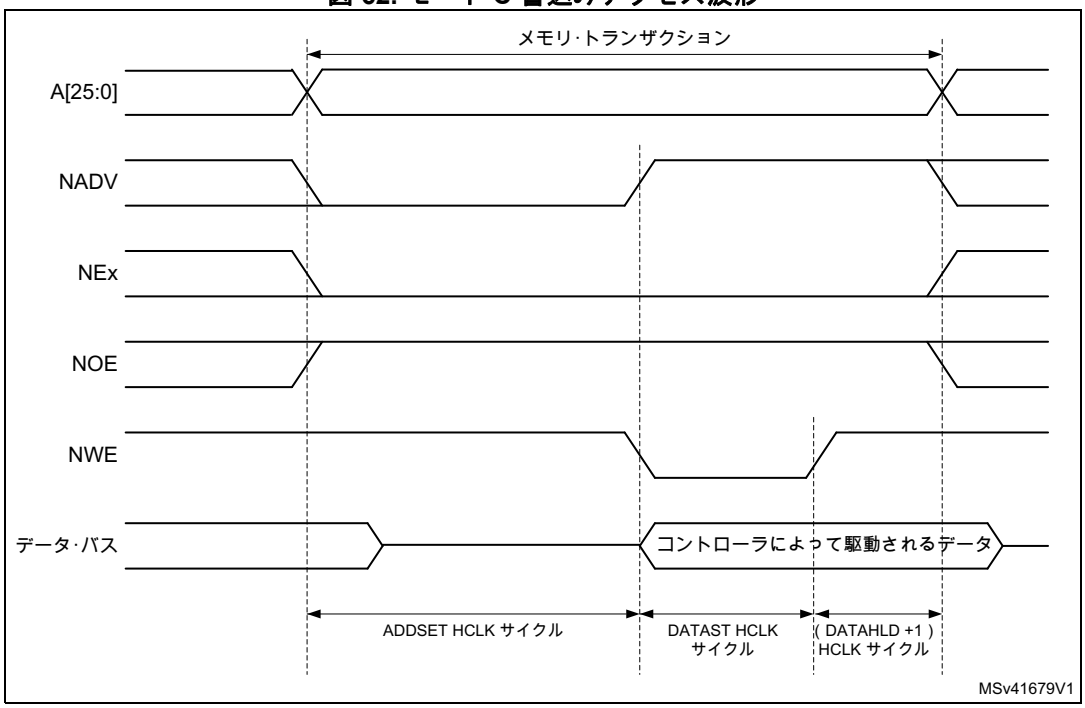


図 62. モード C 書込みアクセス波形



モード 1 と比較した場合の相違点は、NOE のトグルと、独立した読出しおよび書込みのタイミングです。

表 137. FMC_BCRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31:24	予約済み	0x000
23:22	NBLSET[1:0]	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x02 （NOR 型 Flash メモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 138. FMC_BTRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合は DATAHLD HCLK サイクル）。
29:28	ACCMOD	0x2
27:24	DATLAT	0x0
23:20	CLKDIV	0x0
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	読出しアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

表 139. FMC_BWTRx ビットフィールド（モード C）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	0x2
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	書込みアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	無視
3:0	ADDSET	書込みアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 0。

モード D - 拡張アドレスによる非同期アクセス

図 63. モード D 読出しアクセス波形

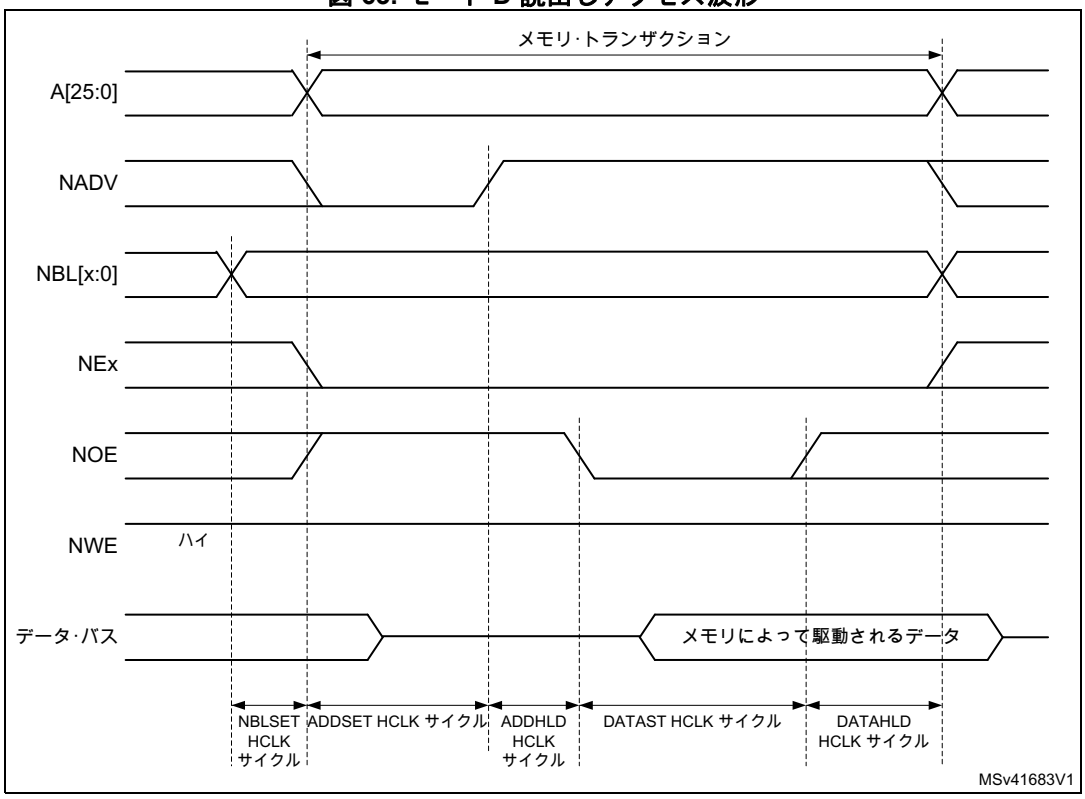
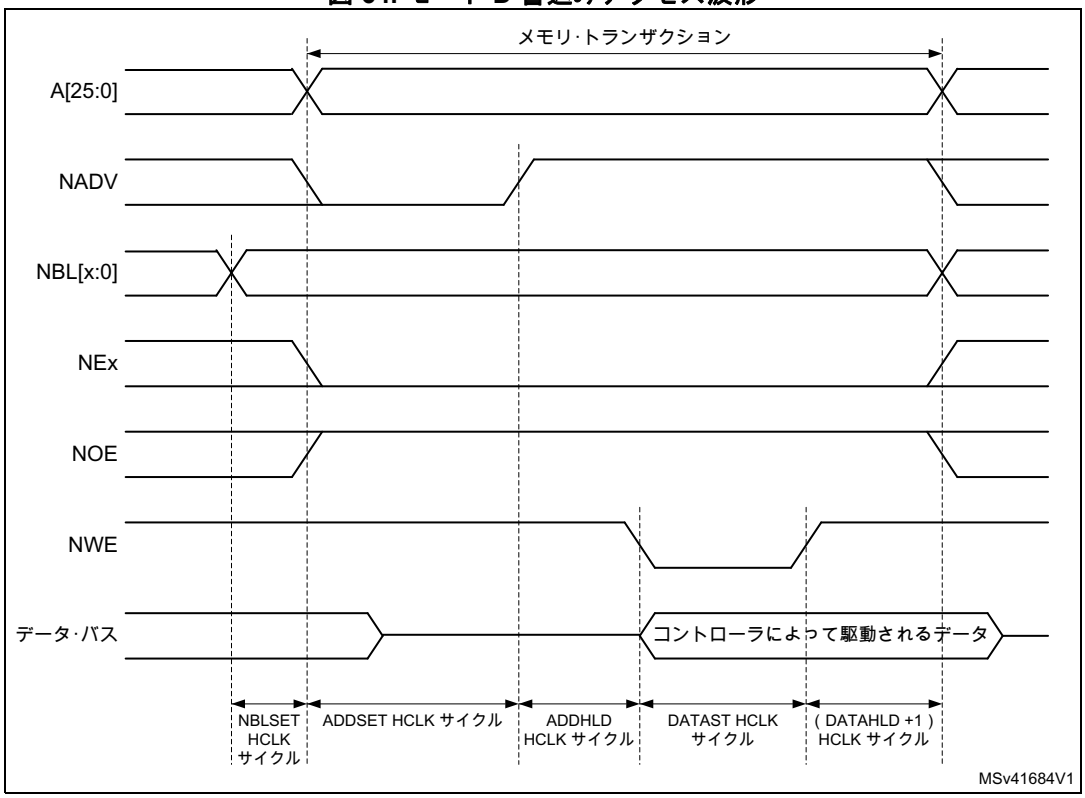


図 64. モード D 書込みアクセス波形



モード 1 と比較した場合の相違点は、NADV が変化した後で NOE のトグルが継続することと独立した読出しと書込みのタイミングです。

表 140. FMC_BCRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31:24	予約済みです。	0x000
23:22	NBLSET[1:0]	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。

表 140. FMC_BCRx ビットフィールド（モード D）（続き）

ビット番号	ビット名	設定値
5:4	MWID	必要に応じて設定します。
3:2	MTYP	必要に応じて設定します。
1	MUXEN	0x0
0	MBKEN	0x1

表 141. FMC_BTRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合は DATAHLD HCLK サイクル）。
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	読出しアクセスに対する 2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	読出しアクセスにおける中間アクセス･フェーズの時間（ADDHLD HCLK サイクル）
3:0	ADDSET	読出しアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

表 142. FMC_BWTRx ビットフィールド（モード D）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	0x3
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	書込みアクセスにおける中間アクセス･フェーズの時間（ADDHLD HCLK サイクル）
3:0	ADDSET	書込みアクセスに対する最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

マルチプレクスモード - NOR 型Flashメモリに対するマルチプレクス非同期アクセス

図 65. マルチプレクス読出しアクセス波形

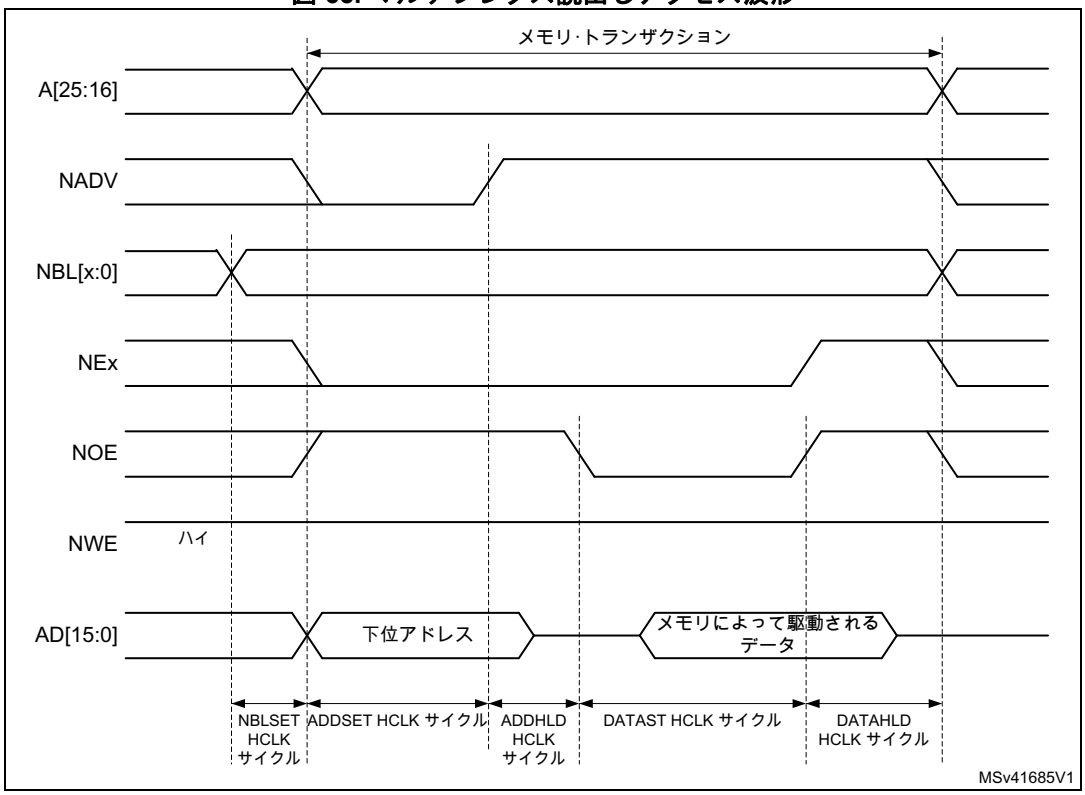
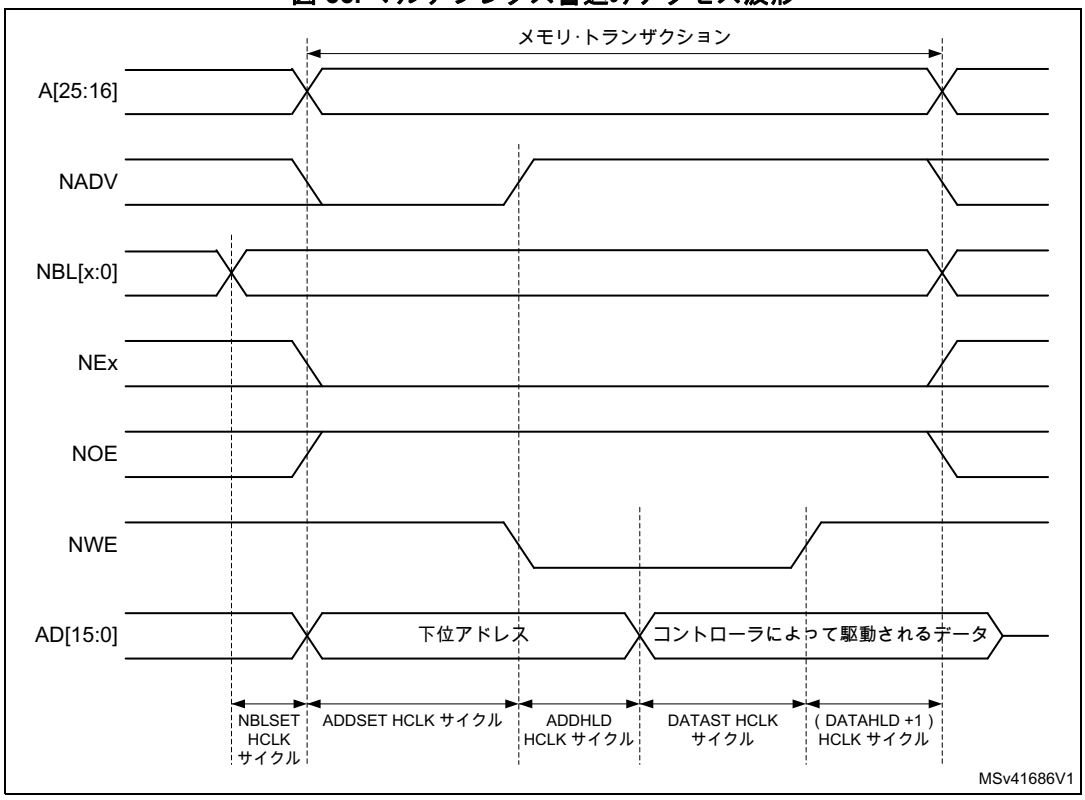


図 66. マルチプレクス書込みアクセス波形



モード D との相違点は、データ･バス上の下位アドレス･バイトの駆動方法です。

表 143. FMC_BCRx ビットフィールド（マルチプレクス･モード）

ビット番号	ビット名	設定値
31:24	予約済みです。	0x000
23:22	NBLSET[1:0]	必要に応じて設定します。
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x0（非同期モードで無効）
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードで無効）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	予約済みです。	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済みです。	0x1
6	FACCEN	0x1

表 143. FMC_BCRx ビットフィールド（マルチプレクス･モード）（続き）

ビット番号	ビット名	設定値
5:4	MWID	必要に応じて設定します。
3:2	MTYP	0x2（NOR 型 Flash メモリ）または 0x1（PSRAM）
1	MUXEN	0x1
0	MBKEN	0x1

表 144. FMC_BTRx ビットフィールド（マルチプレクス･モード）

ビット番号	ビット名	設定値
31:30	DATAHLD	データホールドフェーズの時間（読出しアクセスの場合は DATAHLD HCLK サイクル、書込みアクセスの場合は DATAHLD+1 HCLK サイクル）。
29:28	ACCMOD	0x0
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15:8	DATAST	2 番目のアクセス･フェーズの時間（DATAST HCLK サイクル）。
7:4	ADDHLD	アクセスの中間フェーズの時間（ADDHLD HCLK サイクル）。
3:0	ADDSET	最初のアクセス･フェーズの時間（ADDSET HCLK サイクル）。ADDSET の最小値は 1。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを知らせるために、WAIT 信号をアサートした場合、FMC_BCRx レジスタの ASYNCWAIT ビットをセットする必要があります。

WAIT 信号がアクティブな場合（WAITPOL ビットに依存して、ハイまたはロー）、DATAST ビットによってプログラムされる 2 番目のアクセスフェーズ（データセットアップフェーズ）は、WAIT ビットが非アクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSET およびADDHLD ビットによってプログラムされる最初のアクセスフェーズ（アドレスセットアップおよびアドレスホールドフェーズ）は、WAIT に対応しないので、延長されません。

データセットアップフェーズは、メモリトランザクションが終了する 4 HCLK サイクル前に WAIT が検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルする NOE/NWE に合わせた WAIT 信号をアサートする場合。

$$DATAST \geq (4 \times HCLK) + max_wait_assertion_time$$

2. メモリが、NEx（またはトグルしない NOE/NWE）に合わせた WAIT 信号をアサートする場合。
次の条件が成立するならば、

$$max_wait_assertion_time > address_phase + hold_phase$$

以下の通りです。

$$DATAST \geq (4 \times HCLK) + (max_wait_assertion_time - address_phase - hold_phase)$$

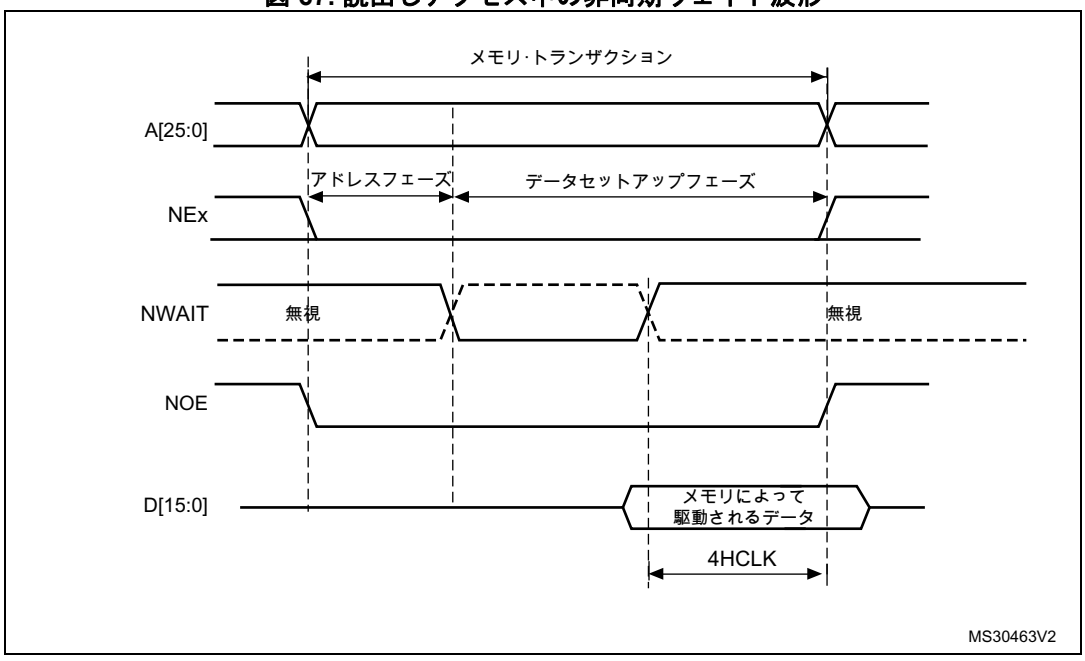
それ以外の場合は、

$$DATAST \geq 4 \times HCLK$$

ここで、max_wait_assertion_time は、NEx/NOE/NWE がローになったときにメモリが WAIT 信号をアサートするために必要な最大時間です。

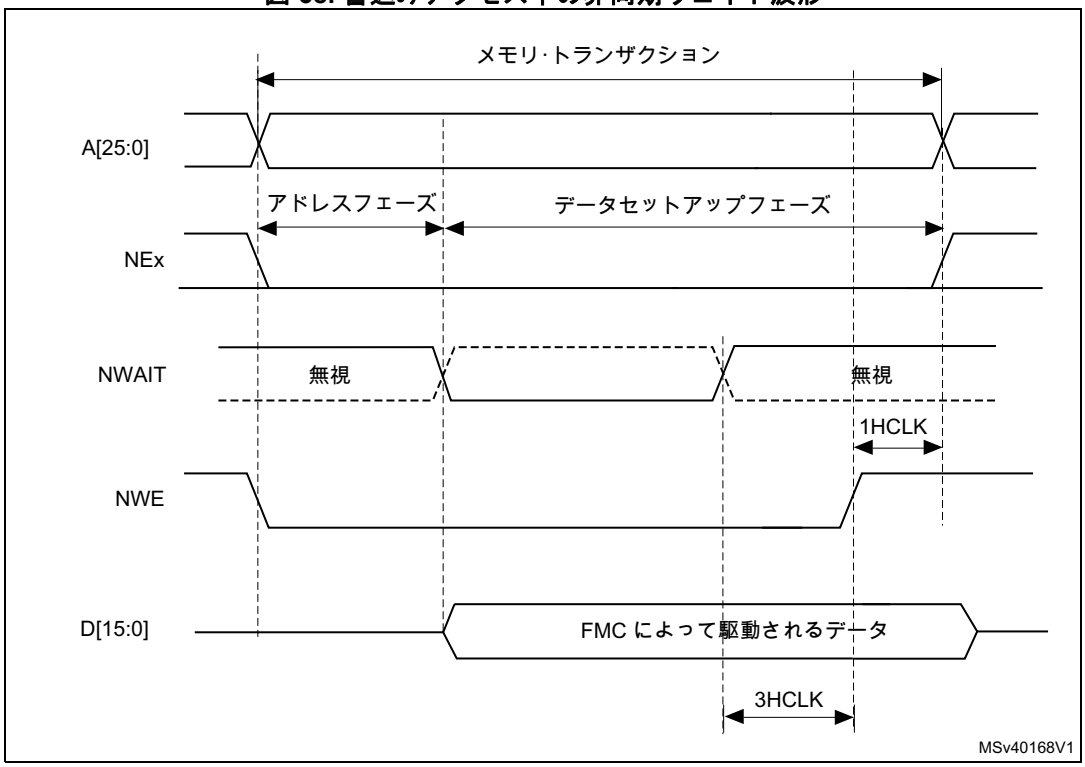
図 67 と図 68 に、非同期メモリによって WAIT が解除された後、メモリ･アクセス･フェーズに追加される HCLK クロックのサイクル数を示します（上記のケースに無関係）。

図 67. 読出しアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

図 68. 書込みアクセス中の非同期ウェイト波形



1. NWAIT 極性は、FMC_BCRx レジスタの WAITPOL ビット設定値に依存します。

CellularRAM™（PSRAM）のリフレッシュ管理

CellularRAM™ では、チップセレクト信号（NE）をメモリデバイスに指定されている t_{CEM} タイミングよりも長くローに維持することはできません。このタイミングは FMC_PCSCNTR レジスタでプログラムすることができます。そこには、非同期アクセスの場合は HCLK サイクルで、同期アクセスの場合は FMC_CLK サイクルで、NE のローパルスの最大期間を定義します。

19.6.5 同期トランザクション

メモリクロック FMC_CLK は、HCLK の約数です。この値は CLKDIV 値と MWID/AHB データ･サイズに依存し、次式で与えられます。

MWID サイズにかかわりなく（16 ビット と 8 ビットのいずれであっても）、FMC_CLK の分周比は必ず CLKDIV のプログラム値で決まります。

例：

- CLKDIV = 1、MWID = 16ビット、AHB データサイズ = 8 ビットの場合、FMC_CLK = HCLK / 2

NOR 型Flashメモリは、NADV のアサーションから CLK がハイになるまでの最小時間を指定します。この制約を満たすために、FMC は、同期アクセスの最初の内部クロックサイクルでは（NADV アサーションの前）、クロックをメモリに供給しません。これにより、メモリクロックの立ち上がりエッジは、NADV ローパルスの中間で発生します。

データ遅延と NOR メモリ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLAT の値は、NOR 型Flash設定レジスタで指定された遅延の値と一致しなければなりません。FMC は、NADV がローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： **一部の NOR Flashメモリは、データ遅延カウントに NADV ローサイクルを含めるので、NOR Flash の遅延と FMC DATLAT パラメータの厳密な関係は、次のいずれかです。**

- NOR 型Flash遅延 = (DATLAT + 2) CLK クロックサイクル、または
- NOR 型Flash遅延 = (DATLAT + 3) CLK クロックサイクル

最近のメモリの中には、遅延フェーズにおいて NWAIT をアサートするものがあります。そのような場合には、DATLAT を最小値に設定することができます。結果として、FMC はデータをサンプリングして、データが有効であるかどうか評価するのに十分な時間、ウェイトします。このように、FMC は、メモリが遅延から抜けたことを検出し、実際のデータが処理されます。

その他のメモリは、遅延中には NWAIT をアサートしません。この場合、FMC とメモリの両方について、遅延が正しく設定されなければなりません。そうしないと、正しいデータの代わりに無効なデータが取られたり、有効なデータがメモリサイクルの初期フェーズで失われることになります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードに設定されている場合、例えば、16 ビットメモリで AHB シングルバーストトランザクションがリクエストされると、FMC は長さ 1（AHB 転送が 16 ビットの場合）または 2（AHB 転送が 32 ビットの場合）のバーストトランザクションを実行して、最後のデータがストローブされると、チップセレクト信号をデアサートします。

このような転送はサイクル数の点で、非同期読出し動作に比較して最も効率的ではありません。しかし、時間がかかるメモリアクセスモードの再プログラムには、まず、ランダム非同期アクセスが必要です。

CellularRAM™ 1.5 のページ境界跨ぎ

CellularRAM™ 1.5 では、ページ境界を跨ぐバーストアクセスは許可されていません。メモリページサイズに従って FMC_BCR1 レジスタの CPSIZE ビットを設定することで、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割できます。

ウェイト管理

同期 NOR Flashメモリでは、NWAIT がプログラムされた遅延期間（(DATLAT+2) CLK サイクルに対応）後に評価されます。

NWAIT がアクティブの場合（WAITPOL = 0 ではローレベル、WAITPOL = 1 ではハイレベル）、ウェイトステートは NWAIT が非アクティブ（WAITPOL = 0 ではハイレベル、WAITPOL = 1 ではローレベル）になるまで挿入されます。

NWAIT が非アクティブのときには、データはただちに（ビット WAITCFG=1）または次のクロックエッジで（ビット WAITCFG=0）、有効とみなされます。

NWAIT 信号によるウェイトステートの挿入時には、コントローラはメモリへのクロックパルスの送信を続け、チップセレクトと出力イネーブル信号を有効に維持します。データは有効と見なしません。

バーストモードでの NOR Flashの NWAIT 信号には、以下の 2 つのタイミング設定があります。

- Flashメモリは、ウェイトステートの 1 データサイクル前に NWAIT 信号をアサートします（リセット後のデフォルト）。
- Flashメモリは、ウェイトステートの間に NWAIT 信号をアサートします。

FMC は FMC_BCRx レジスタ（x = 0..3）の WAITCFG ビットを使って、各チップセレクトに対する両 NOR Flashウェイトステート設定をサポートします。

図 69. 待ち設定波形

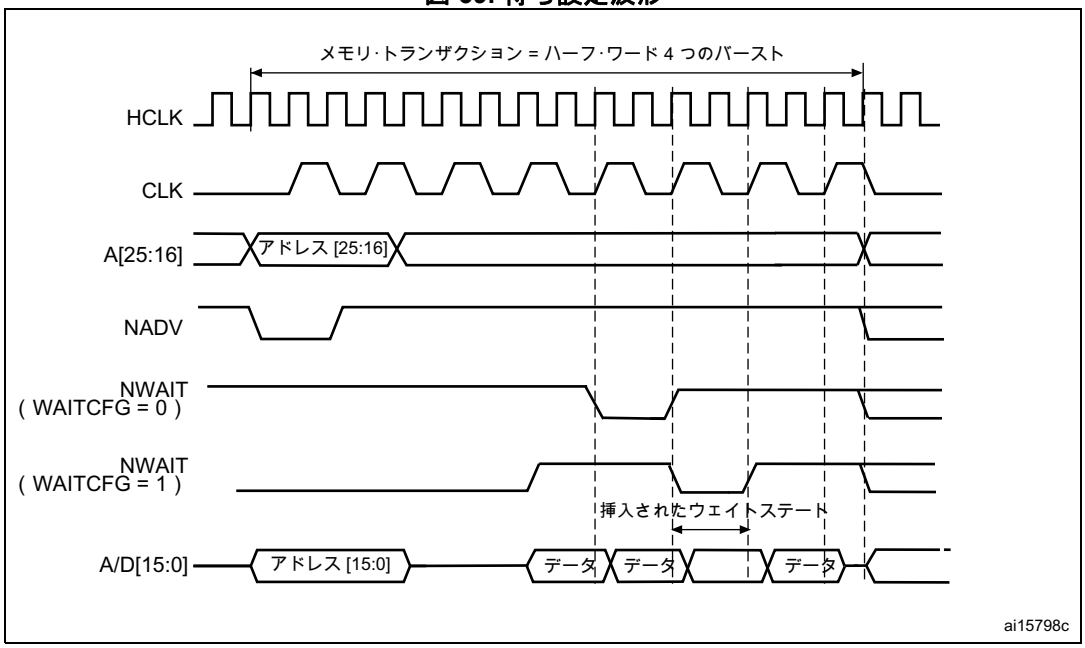
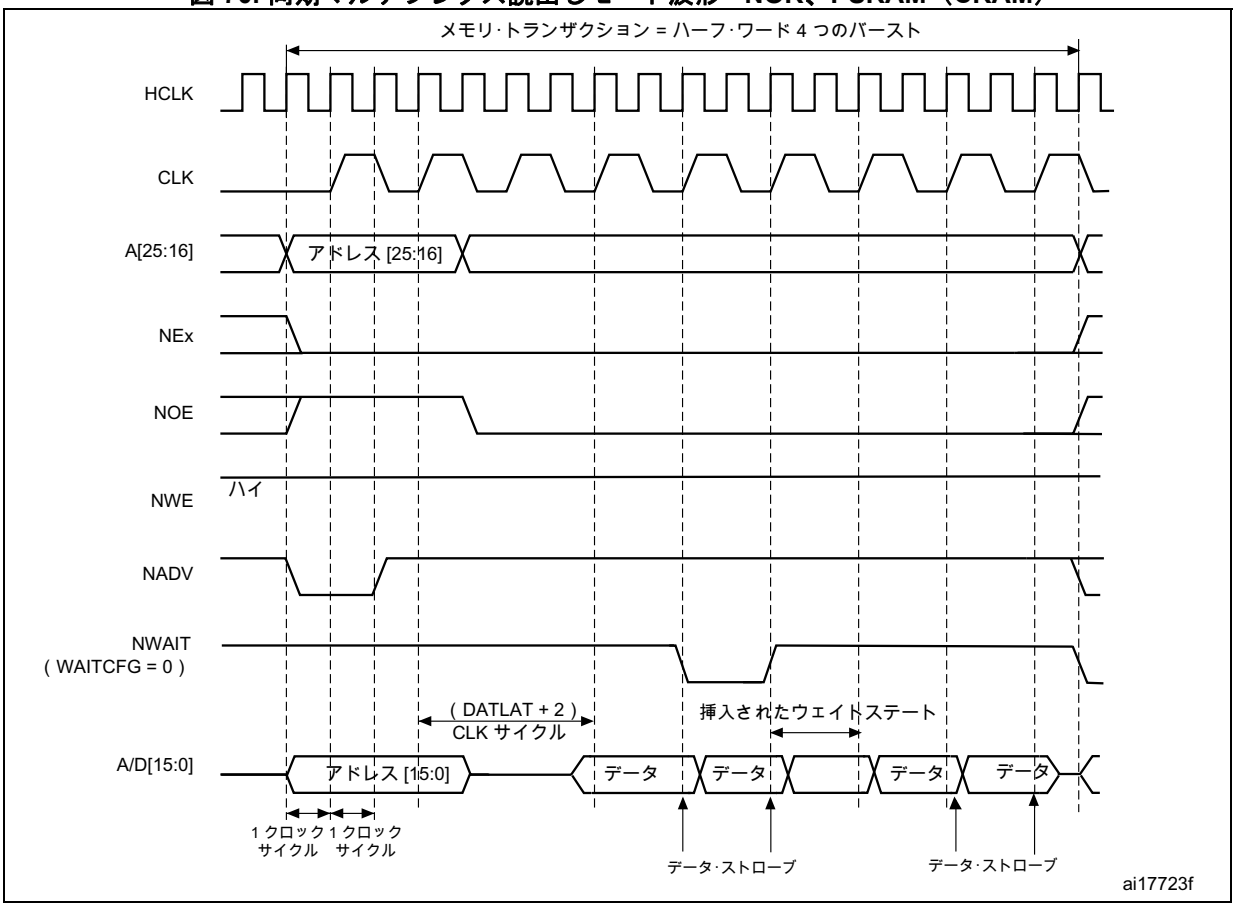


図 70. 同期マルチプレクス読出しモード波形 - NOR、PSRAM（CRAM）



1. バイトレーン出力（NBL は示されていません。NOR アクセス時にはハイに保たれ、PSRAM（CRAM）アクセス時にはローに保たれます。）

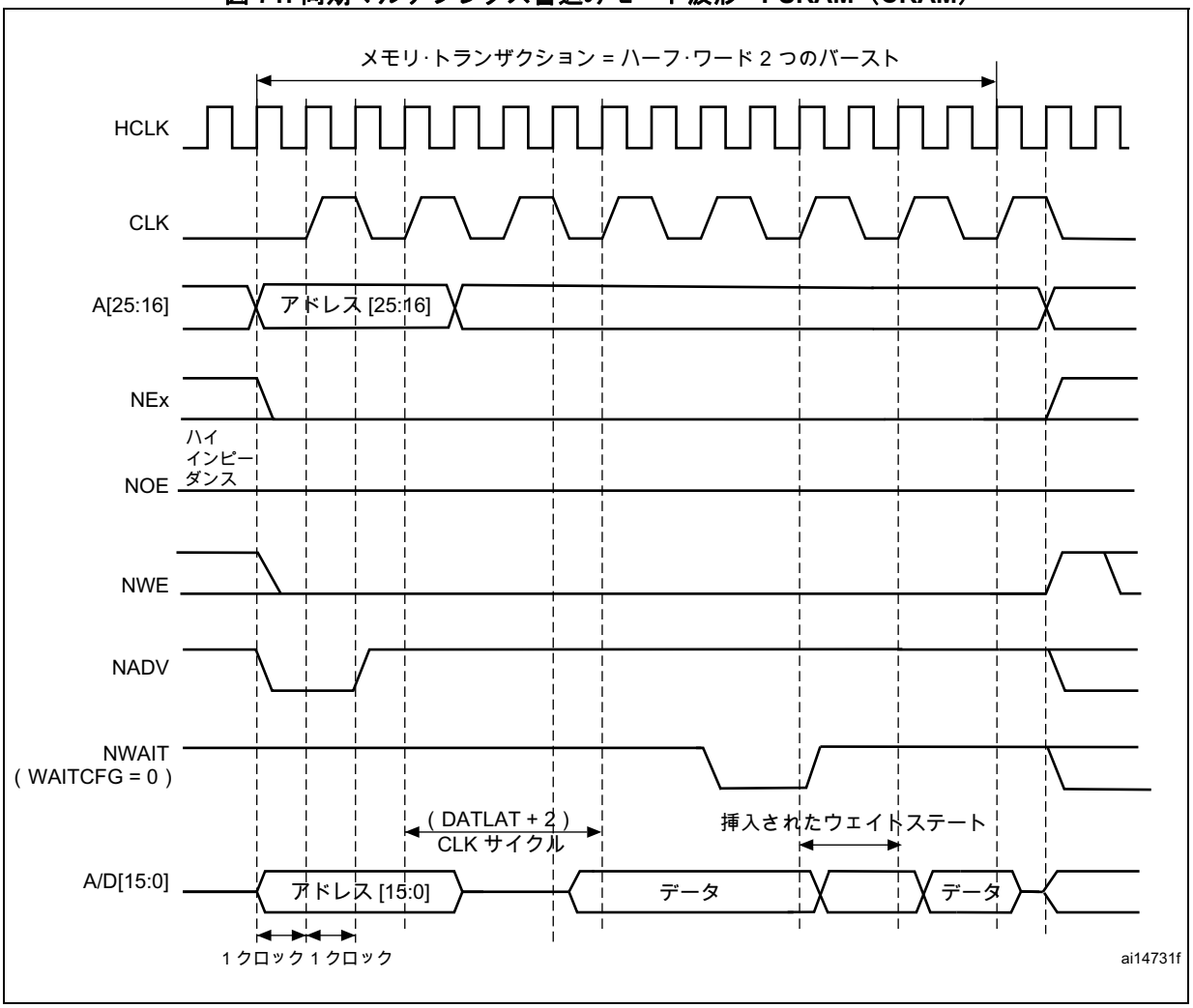
表 145. FMC_BCRx ビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31:24	予約済みです。	0x000
23:22	NBLSET[1:0]	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	同期読出し時は影響しません。
18:16	CPSIZE	0x0（非同期モードで無効）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	同期読出し時は影響しません。
11	WAITCFG	メモリに応じて設定します。
10	予約済みです。	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済みです。	0x1
6	FACCEN	メモリ･サポートに従って設定します（NOR 型 Flash メモリ）
5～4	MWID	必要に応じて設定します。
3～2	MTYP	0x1 または 0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 146. FMC_BTRx ビットフィールド（同期マルチプレクス読出しモード）

ビット番号	ビット名	設定値
31:30	DATAHLD	無視
29:28	ACCMOD	0x0
27 ~ 24	DATLAT	データ遅延
27～24	DATLAT	データ遅延
23～20	CLKDIV	0x0 の場合、CLK＝HCLK となります。 0x1 の場合、CLK＝2×HCLK となります。 ..
19～16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15～8	DATAST	無視
7～4	ADDHLD	無視
3～0	ADDSET	無視

図 71. 同期マルチプレクス書込みモード波形 - PSRAM（CRAM）



1. メモリは、NWAIT 信号を 1 サイクル前に発行しなければならないので、WAITCFG を 0 にプログラムする必要があります。
2. バイトレーン（NBL）出力は、図に示されておらず、NEx がアクティブな間はローに保たれます。

表 147. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31:24	予約済みです。	0x000
23:22	NBLSET[1:0]	無視
20	CCLKEN	必要に応じて設定します。
19	CBURSTRW	0x1
18:16	CPSIZE	必要に応じて設定します（CRAM 1.5 では 0x1）
15	ASYNCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	0x1

表 147. FMC_BCRx ビットフィールド（同期マルチプレクス書込みモード）（続き）

ビット番号	ビット名	設定値
11	WAITCFG	0x0
10	予約済みです。	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期書込み時は影響しません。
7	予約済みです。	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5〜4	MWID	必要に応じて設定します。
3〜2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 148. FMC_BTRx ビットフィールド（同期マルチプレクス書込みモード）

ビット番号	ビット名	設定値
31 ~ 30	DATAHLD	無視
29:28	ACCMOD	0x0
27 ~ 24	DATLAT	データ遅延
23〜20	CLKDIV	0x0 の場合、CLK＝HCLK となります。 0x1 の場合、CLK＝2×HCLK となります。
19〜16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）。
15〜8	DATAST	無視
7〜4	ADDHLD	無視
3〜0	ADDSET	無視

19.6.6 NOR/PSRAM コントローラレジスタ

SRAM/NOR 型 Flash のバンク x のチップセレクト制御レジスタ（FMC_BCRx）（x = 1 ~ 4）

アドレスオフセット：8 * (x – 1)、x = 1〜4

リセット値：バンク 1：0x0000:30DB

リセット値：バンク 2：0x0000:30D2

リセット値：バンク 3：0x0000:30D2

リセット値：バンク 4：0x0000:30D2

このレジスタは、SRAM、PSRAM、FRAM、および NOR 型 Flash メモリで使用される各メモリ･バンクの制御情報を含みます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBLSET[1:0]		WFDIS	CCLK EN	CBURST RW	CPSIZE[2:0]		
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ASYNCW AIT	EXT MOD	WAIT EN	WREN	WAIT CFG	Res.	WAIT POL	BURSTE N	Res.	FACC EN	MWID[1:0]		MTYP[1:0]		MUX EN	MBK EN
rw	rw	rw	rw	rw		rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:22 **NBLSET[1:0]**：バイト･レーン（NBL）セットアップ

これらのビットは、NBLx ローからチップセレクト NEx ローまでの NBL セットアップ時間を設定します。

00：NBL セットアップ時間は、0 AHB クロック･サイクルです。

01：NBL セットアップ時間は、1 AHB クロック･サイクルです。

10：NBL セットアップ時間は、2 AHB クロック･サイクルです。

11：NBL セットアップ時間は、3 AHB クロック･サイクルです。

ビット 21 **WFDIS**：書込み FIFO 無効化

このビットは、FMC コントローラで使用される書込み FIFO を無効にします。

0：書込み FIFO 有効（リセット後のデフォルト）

1：書込み FIFO 無効

注： FMC_BCR2..4 レジスタの WFDIS ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。

ビット 20 **CCLKEN**：連続クロック有効化

このビットは、外部メモリ･デバイスに対する FMC_CLK クロック出力を有効化します。

0：FMC_CLK は同期メモリ･アクセス（読出し／書込みトランザクション）時にのみ生成されます。FMC_CLK クロック比は、FMC_BCRx レジスタに設定した CLKDIV 値により指定されます（リセット後のデフォルト）。

1：FMC_CLK は、非同期アクセスと同期アクセス時に連続的に生成されます。CCLKEN がセットされると、FMC_CLK クロックがアクティブになります。

注： FMC_BCR2..4 レジスタの CCLKEN ビットは無視されます。このビットは FMC_BCR1 レジスタからのみ有効化できます。FMC_CLK 連続クロックを発生するときは、バンク 1 を同期モードに設定する必要があります。

注： CCLKEN ビットがセットされている場合、FMC_CLK クロック比は FMC_BTR1 レジスタの CLKDIV 値で指定されます。FMC_BWTR1 の CLKDIV は、無視されます。

注： 同期モードを使用し、かつ CCLKEN ビットがセットされる場合、バンク 1 以外のバンクに接続される同期メモリはすべて、同じクロックで駆動されます（他のバンクに対する FMC_BTR2..4 レジスタと FMC_BWTR2..4 レジスタの CLKDIV 値は無効です）。

ビット 19 **CBURSTRW**：書き込みバースト有効化

バーストモードで動作する PSRAM（CRAM）の場合、このビットは書込み動作時に同期アクセスを有効にします。同期読出しアクセスのイネーブルビットは、FMC_BCRx レジスタの BURSTEN ビットです。

0：書込み動作は、常に非同期モードで実行されます。

1：書込み動作は、同期モードで実行されます。

ビット 18:16 **CPSIZE[2:0]**：CRAM ページ･サイズ

これらのビットは、ページ間でアドレス境界を跨ぐバースト･アクセスができない CellularRAM™ 1.5 に使用されます。これらのビットを設定すると、FMC コントローラは、メモリページサイズに達した時点で自動的にバーストアクセスを分割します（ページサイズについてはメモリのデータシートを参照）。

000：ページ境界を跨ぐ場合、バーストは分割されません（リセット後のデフォルト）

001：128 バイト

010：256 バイト

011：512 バイト

100：1024 バイト

その他：予約済み

ビット 15 **ASYNCWAIT**：非同期転送中のウエイト信号

このビットは、非同期プロトコル時でも、FMC が ウェイト信号を有効化／無効化できるようにします。

0：非同期プロトコル動作中は、NWAIT 信号は考慮されません（リセット後のデフォルト）。

1：非同期プロトコル動作中に NWAIT 信号が考慮されます。

ビット 14 **EXTMOD**：拡張モード有効化

このビットを使うと、FMC が FMC_BWTR レジスタに非マルチプレクス非同期アクセスの書込みタイミングをプログラムできるため、読出しと書込みで異なるタイミングを設定できます。

0：FMC_BWTR レジスタ内部の値は考慮されません（リセット後のデフォルト）。

1：FMC_BWTR レジスタ内部の値が考慮されます。

注： 拡張モードを無効化した場合、FMC は次のようにモード 1 またはモード 2 で動作できます。

- **モード 1 は、SRAM/PSRAM メモリ･タイプを選択した場合 （MTYP = 0x0 または 0x01）のデフォルト･モードです。**
- **モード 2 は、NOR メモリ･タイプを選択した場合 （MTYP = 0x10）のデフォルト･モードです。**

ビット 13 **WAITEN**：ウェイト有効ビット

このビットは、同期モードでFlashメモリをアクセスするとき NWAIT 信号を使用したウェイトステートの挿入を有効化／無効化します。

0：NWAIT 信号は無効です（レベルは考慮されず、プログラムされた Flash 遅延時間後にウェイトステートは挿入されません）。

1：NWAIT 信号は有効です（レベルが考慮され、アサートされた場合、プログラムされた Flash 遅延時間後にウェイトステートが挿入されます）（リセット後のデフォルト）。

ビット 12 **WREN**：書き込み有効ビット

このビットは、FMC によるバンクへの書込み動作の有効または無効を示します。

0：FMC によるバンクへの書込み動作は無効にされています。AHB エラーが報告されます。

1：FMC によるバンクへの書込み動作は有効にされています（リセット後のデフォルト）。

ビット 11 **WAITCFG**：ウェイト･タイミング設定

NWAIT 信号は、メモリからのデータが有効か否か、または同期モードでメモリをアクセスするときウェイトステートを挿入すべきかを表示します。この設定ビットは、ウェイトステートの 1 クロック前、またはウェイトステート中に、メモリによって NWAIT がアサートされるかどうかを定義します。

0：NWAIT 信号は、ウェイトステートの 1 データ･サイクル前にアクティブになります（リセット後のデフォルト）。

1：NWAIT 信号は、ウェイトステートの間アクティブです（PSRAM には不使用）。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **WAITPOL**：ウェイト信号極性ビット

同期モードまたは非同期モードで使用される、メモリからのウェイト信号の極性を定義します。
0：NWAIT はアクティブ･ローです（リセット後のデフォルト）。
1：NWAIT はアクティブ･ハイです。

ビット 8 **BURSTEN**：バースト有効ビット

このビットは、読出し動作での同期アクセスを有効化/無効化します。これは、バースト･モードで動作している同期メモリに対してのみ有効です。
0：バーストアクセスモードは無効です（リセット後のデフォルト）。読出しアクセスは非同期モードで実行されます。
1：バーストモードを有効化。読出しアクセスは同期モードで実行されます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FACCEN**：Flash アクセス有効化

NOR 型Flashメモリアクセス動作を有効にします。
0：対応する NOR 型 Flash メモリ･アクセスは無効です。
1：対応する NOR 型 Flash メモリ･アクセスは有効です（リセット後のデフォルト）。

ビット 5:4 **MWID[1:0]**：メモリのデータ･バス幅

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。
00：8 ビット
01：16 ビット（リセット後のデフォルト）
10：予約済み
11：予約済み

ビット 3:2 **MTYP[1:0]**：メモリタイプ

対応するメモリ･バンクに接続される外部メモリのタイプを定義します。
00：SRAM/FRAM（バンク 2...4 のリセット後のデフォルト）
01：PSRAM（CRAM）/ FRAM
10：NOR 型 Flash/OneNAND Flash（バンク 1 のリセット後のデフォルト）
11：予約済み

ビット 1 **MUXEN**：アドレス / データ･マルチプレクス有効ビット

このビットがセットされているときには、データバス上でアドレスとデータがマルチプレクスされます。NOR および PSRAM メモリでのみ有効です。
0：アドレス/データはマルチプレクスされません。
1：アドレス/データはデータ･バス上でマルチプレクスされます（リセット後のデフォルト）。

ビット 0 **MBKEN**：メモリ･バンク有効ビット

メモリ･バンクを有効にします。リセット後は、バンク 1 は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AHB バスでエラーが発生します。
0：対応するメモリ･バンクは無効です。
1：対応するメモリ･バンクは有効です。

SRAM/NOR 型 Flash のバンク x のチップセレクト･タイミング･レジスタ（FMC_BTRx）

アドレスオフセット：0x04 + 8 * (x – 1)、x = 1～4

リセット値：0x0FFF FFFF

このレジスタは、SRAM、PSRAM、および NOR 型 Flash メモリに使用する各メモリ･バンクの制御情報を収めています。FMC_BCRx レジスタの EXTMOD ビットがセットされている場合、このレジスタは、書込みアクセスと読出しアクセスに分割されます。すなわち、読出しアクセスを設定するためのレジスタ（このレジスタ）と、書込みアクセスを設定するためのレジスタ（FMC_BWTRx レジスタ）の 2 つのレジスタがあります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 **DATAHLD[1:0]**：データホールドフェーズ時間

これらのビットは、データホールドフェーズの時間を HCLK サイクル数で定義するためにソフトウェアによって書き込まれ（図 54 ～ 図 66 を参照）、非同期アクセスで使用されます。

読出しアクセスの場合

00：DATAHLD フェーズ時間 = 0 * HCLK クロック･サイクル（デフォルト）

01：DATAHLD フェーズ時間 = 1 * HCLK クロック･サイクル

10：DATAHLD フェーズ時間 = 2 * HCLK クロック･サイクル

11：DATAHLD フェーズ時間 = 3 * HCLK クロック･サイクル

書込みアクセスの場合

00：DATAHLD フェーズ時間 = 1 * HCLK クロック･サイクル（デフォルト）

01：DATAHLD フェーズ時間 = 2 * HCLK クロック･サイクル

10：DATAHLD フェーズ時間 = 3 * HCLK クロック･サイクル

11：DATAHLD フェーズ時間 = 4 * HCLK クロック･サイクル

ビット 29:28 **ACCMOD[1:0]**：アクセスモード

タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 のときのみ考慮されます。

00：アクセス･モード A

01：アクセス･モード B

10：アクセス･モード C

11：アクセス･モード D

ビット 27:24 **DATLAT[3:0]**：（下のビット説明参照）同期メモリのデータ遅延

読出し／書込みバーストモードを有効にした同期アクセスの場合（BURSTEN / CBURSTRW ビットをセット）、最初のデータの読み書きの前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、HCLK 周期ではなく、FMC_CLK 周期で表されます。

非同期アクセスでは、この値は無視されます。

0000：最初のバースト･アクセスでは、2 CLK クロック･サイクルのデータ遅延

1111：最初のバースト･アクセスでは、17 CLK クロック･サイクルのデータ遅延（リセット後のデフォルト）

ビット 23:20 **CLKDIV[3:0]**：クロック分周比（FMC_CLK 信号）

FMC_CLK クロック出力信号の周期を定義します。HCLK サイクル数で表されます。

0000：FMC_CLK 周期 = 1 × HCLK 周期

0001：FMC_CLK 周期 = 2 × HCLK 周期

0010：FMC_CLK 周期 = 3 × HCLK 周期

1111：FMC_CLK 周期 = 16 × HCLK 周期（リセット後のデフォルト値）

非同期 NOR 型Flash、SRAM、または PSRAM アクセスでは、この値は無視されます。

注： FMC_CLK 分周比の式については、セクション 19.6.5：同期トランザクションを参照してください。

ビット 19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

これらのビットはソフトウェアから書き込まれて、同じバンクの次のトランザクションまでの現在の読出しまたは書込みトランザクションの終わりに遅延を追加します。

この遅延を使うと、連続トランザクション間の最小時間（NEx ハイから NEx ローまでの t_{EHEL}）と、読出しアクセス後にメモリがデータ･バスを開放するまでに要する最大時間（チップ･イネーブル･ハイから出力ハイ･インピーダンスまでの t_{EHQZ}）を次のように一致させることができます。モード D およびマルチプレクスモードでは、この遅延が推奨されます。非マルチプレクスメモリでは、バスターンアラウンド遅延を最小値にセットできます。

(BUSTURN + 1) HCLK 周期 ≥ max(t_{EHEL} min, t_{EHQZ} max)

FRAM メモリでは、バス･ターンアラウンド遅延を最小 tPC（プリチャージ時間）タイミングに一致するように設定する必要があります。バスターンアラウンド遅延は、tPC メモリタイミングと合わせて、同じバンクでの連続トランザクション間（読出し／読出し、書込み／書込み、読出し／書込み、および書込み／読出し）に挿入されます。チップセレクトは、連続アクセス間でトグルします。

(BUSTURN + 1) HCLK 周期 ≥ t_{PC} min

0000：BUSTURN フェーズ時間 = 1 HCLK クロック･サイクル追加

.....

1111：BUSTURN フェーズ時間 = 16 × HCLK クロック･サイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST[7:0]**：データフェーズ時間

これらのビットは、非同期アクセスで使用されるデータフェーズの時間を定義するためにソフトウェアにより書き込まれます（図 54 ～ 図 66 を参照）。

0000 0000：予約済みです。

0000 0001：DATAST フェーズ時間 = 1 * HCLK クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 * HCLK クロックサイクル

.....

1111 1111：DATAST フェーズ時間 = 255 * HCLK クロックサイクル（リセット後のデフォルト値）

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図（図 54 ～ 図 66）を参照してください。

例：モード 1 書込みアクセス、DATAST = 1：データフェーズ時間 = DATAST + 1 = 2 HCLK クロック･サイクル

注： 同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD[3:0]**：アドレスホールド･フェーズ時間

これらのビットは、アドレスホールドフェーズの時間を定義するためにソフトウェアから書き込まれ（図 54 ～ 図 66 を参照）、モード D またはマルチプレクスアクセスで使用されます。

0000：予約済みです。

0001：ADDHLD フェーズ時間 = 1 * HCLK クロック･サイクル

0010：ADDHLD フェーズ時間 = 2 * HCLK クロック･サイクル

.....

1111：ADDHLD フェーズ時間 = 15 * HCLK クロック･サイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図（図 54 ～ 図 66）を参照してください。

注： 同期アクセスでは、この値は使用されず、アドレスホールドフェーズは、常に 1 メモリクロック周期の長さです。

ビット 3:0 **ADDSET[3:0]**：アドレス･セットアップ･フェーズ時間

これらのビットは、アドレスセットアップフェーズの時間を定義するために、ソフトウェアによって書き込まれ（図 54 ～ 図 66 を参照）、SRAM、ROM、非同期 NOR Flash、および PSRAM で使用されます。

0000：ADDSET フェーズ時間 = 0 * HCLK クロック･サイクル

.....

1111：ADDSET フェーズ時間 = 15 * HCLK クロック･サイクル（リセット後のデフォルト値）

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図（図 54 ～ 図 66）を参照してください。

注： 同期アクセスでは、この値は無視されます。

マルチプレクスモードとモード D では、ADDSET の最小値は 1です。

モード 1 と PSRAM メモリでは、ADDSET の最小値は 1 です。

注： PSRAM（CRAM）は内部リフレッシュがあるため遅延（レイテンシ）が可変になっています。したがって、これらのメモリは必要に応じてレイテンシ期間を延長するために、レイテンシフェーズ全体に NWAIT 信号を発行します。
PSRAM（CRAM）では、フィールドDATLATを0に設定しなければなりません。こうすると、FSMCは、すぐにレイテンシフェーズを終了して、メモリからのNWAIT信号のサンプリングを開始し、メモリがレディ状態になると、読出しまたは書込みを開始します。
この方法は、古い世代のFlashメモリと異なり、NWAIT 信号を発行する最新世代の同期Flashメモリでも使用できます（使用中の特定のFlashメモリのデータシートを参照してください）。

SRAM/NOR 型 Flash のバンク x の書込みタイミング･レジスタ（FMC_BWTRx）

アドレスオフセット：0x104 + 8 * (x – 1)、x = 1～4

リセット値：0x0FFF FFFF

このレジスタには各メモリバンクの制御情報が格納されます。SRAM、PSRAM、NOR Flashメモリに使用されます。FMC_BCRx レジスタの EXTMOD ビットがセットされているときには、このレジスタは書込みアクセスについてアクティブになります。

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

ビット 31:30 **DATAHLD[1:0]**：データホールドフェーズ時間

これらのビットは、データホールドフェーズの時間を HCLK サイクル数で定義するためにソフトウェアによって書き込まれ（図 54 ～ 図 66 を参照）、非同期書込みアクセスで使用されます。

00：DATAHLD フェーズ時間 = 1 * HCLK クロック･サイクル（デフォルト）

01：DATAHLD フェーズ時間 = 2 * HCLK クロック･サイクル

10：DATAHLD フェーズ時間 = 3 * HCLK クロック･サイクル

11：DATAHLD フェーズ時間 = 4 * HCLK クロック･サイクル

ビット 29:28 **ACCMOD[1:0]**：アクセス･モード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FMC_BCRx レジスタの EXTMOD ビットが 1 にセットされている場合のみ考慮されます。

00：アクセス･モード A

01：アクセス･モード B

10：アクセス･モード C

11：アクセス･モード D

ビット 27:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BUSTURN[3:0]**：バスターンアラウンドフェーズ時間

これらのビットはソフトウェアから書き込まれて、同じバンクの次のトランザクションまでの現在の書込みトランザクションの終わりに遅延を追加します。

FRAM メモリでは、バス･ターンアラウンド遅延を最小 t_{PC}（プリチャージ時間）タイミングに一致するように設定する必要があります。バス･ターンアラウンド遅延は、同じバンクでの連続トランザクション間（読出しと読出し、書込みと書込み、読出しと書込み、書込みと読出し）に挿入されます。チップセレクトは、連続アクセス間でトグルします。

(BUSTURN + 1) HCLK 周期 ≥ tPC min

0000：BUSTURN フェーズ時間 = 1 HCLK クロック･サイクル追加

.....

1111：BUSTURN フェーズ時間 = 16 × HCLK クロック･サイクル追加 （リセット後のデフォルト値）

ビット 15:8 **DATAST[7:0]：**データフェーズ時間

これらのビットは、非同期 SRAM、PSRAM、NOR Flashメモリアクセスで使用されるデータフェーズの時間を定義するために、ソフトウェアによって書き込まれます（図 54 ～ 図 66 を参照）。

0000 0000：予約済みです。

0000 0001：DATAST フェーズ時間 = 1 * HCLK クロックサイクル

0000 0010：DATAST フェーズ時間 = 2 * HCLK クロックサイクル

.....

1111 1111：DATAST フェーズ時間 = 255 * HCLK クロックサイクル（リセット後のデフォルト値）

ビット 7:4 **ADDHLD[3:0]：**アドレスホールド･フェーズ時間

これらのビットは、アドレスホールドフェーズの時間を定義するためにソフトウェアによって書き込まれ（図 63 ～ 図 66 を参照）、非同期マルチプレクスアクセスで使用されます。

0000：予約済みです。

0001：ADDHLD フェーズ時間 = 1 * HCLK クロック･サイクル

0010：ADDHLD フェーズ時間 = 2 * HCLK クロック･サイクル

.....

1111：ADDHLD フェーズ時間 = 15 * HCLK クロック･サイクル（リセット後のデフォルト値）

注： 同期 NOR 型Flashアクセスでは、この値は使用されず、アドレスホールドフェーズは常に 1 Flashクロック周期の長さです。

ビット 3:0 **ADDSET[3:0]：**アドレス･セットアップ･フェーズ時間。

これらのビットは、アドレスセットアップフェーズの時間を HCLK サイクル数で定義するためにソフトウェアによって書き込まれ（図 54 ～ 図 66 を参照）、非同期アクセスで使用されます。

0000：ADDSET フェーズ時間 = 0 * HCLK クロック･サイクル

.....

1111：ADDSET フェーズ時間 = 15 * HCLK クロック･サイクル（リセット後のデフォルト値）

注： 同期アクセスでは、この値は使用されず、アドレスセットアップフェーズは常に 1 Flashクロック周期の長さです。マルチプレクスモードでは、ADDSET の最小値は 1 です。

PSRAM チップセレクトカウンタレジスタ（FMC_PCSCNTR）

アドレスオフセット：0x20

リセット値：0x0000 0000

このレジスタは、同期および非同期モードの PSRAM チップ･セレクト･カウンタの値を格納します。チップセレクトカウンタはすべてのバンクに共通で、バンクごとに個別に有効にできます。PSRAM の読出しまたは書込みアクセス時に、この値はタイマにロードされ、タイマは NE 信号がローに保持されている間、デクリメントされます。タイマが 0 に達すると、PSRAM コントローラは現在のアクセスを分割し、NE をトグルして PSRAM デバイスのリフレッシュを可能にし、そして新しいアクセスを再開します。プログラムされたカウンタ値によって、PSRAM デバイスに指定された最大 NE パルス幅（t_{CEM}）が保証されます。カウンタは、NE のハイからローへの遷移によって新しいアクセスが開始されるたびに、リロードされてデクリメントが開始されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTB4EN	CNTB3EN	CNTB2EN	CNTB1EN
												rw	rw	rw	rw

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **CNTB4EN**：カウンタバンク 4 有効化

このビットは、PSRAM/NOR バンク 4 に対するチップセレクトカウンタを有効にします。

0：バンク 4 に対するカウンタは無効です。

1：バンク 4 に対するカウンタは有効です。

ビット 18 **CNTB3EN**：カウンタバンク 3 有効化

このビットは、PSRAM/NOR バンク 3 に対するチップセレクトカウンタを有効にします。

0：バンク 3 に対するカウンタは無効です。

1：バンク 3 に対するカウンタは有効です。

ビット 17 **CNTB2EN**：カウンタバンク 2 有効化

このビットは、PSRAM/NOR バンク 2 に対するチップセレクトカウンタを有効にします。

0：バンク 2 に対するカウンタは無効です。

1：バンク 2 に対するカウンタは有効です。

ビット 16 **CNTB1EN**：カウンタバンク 1 有効化

このビットは、PSRAM/NOR バンク 1 に対するチップセレクトカウンタを有効にします。

0：バンク 1 に対するカウンタは無効です。

1：バンク 1 に対するカウンタは有効です。

ビット 15:0 **CSCOUNT[15:0]**：チップセレクトカウンタ

これらのビットは、チップセレクトローパルス時間の最大値を定義するために、ソフトウェアによって書き込まれます。それは、同期アクセスの場合 FMC_CLK サイクルで、非同期アクセスの場合 HCLK サイクルで表されます。

プログラムされた値が 0 の場合、カウンタは無効になります。

19.7 NAND 型Flashコントローラ

FMC は、以下のタイプのデバイスを駆動するのに適した信号タイミングを発生します。

- 8 ビットと 16 ビットの NAND Flashメモリ

NAND バンクは、専用レジスタによって設定されます（セクション 19.7.7：を参照）。プログラム可能なメモリパラメータは、アクセスタイミング（表 149 を参照）と ECC 構成を含みます。

表 149. プログラム可能な NAND Flashのアクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
メモリセットアップ時間	コマンドアサーション前にアドレスをセットアップするために必要なクロックサイクル（HCLK）数	読出し／書込み	AHB クロックサイクル（HCLK）	1	255
メモリウェイト	コマンドアサーションの最小時間（HCLK クロックサイクル数）	読出し／書込み	AHB クロックサイクル（HCLK）	2	255
メモリホールド	コマンドアサーション解除後に、アドレスを（書込みアクセス時はデータも）保持する必要があるクロックサイクル数（HCLK）	読出し／書込み	AHB クロックサイクル（HCLK）	1	254
メモリデータバスハイインピーダンス	書込みアクセス開始後にデータバスがハイインピーダンス状態に保たれるクロックサイクル数（HCLK）	書込み	AHB クロックサイクル（HCLK）	1	255

19.7.1 外部メモリインタフェース信号

次の表に、通常、NAND 型Flashメモリとのインタフェースに使用される信号を示します。

注： 接頭辞「N」は、アクティブローの信号を表します。

8 ビット NAND Flashメモリ

表 150. 8 ビット NAND 型Flash

FMC 信号名	I/O	機能
A[17]	O	NAND 型Flashアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型Flashコマンドラッチイネーブル（CLE）信号
D[7:0]	I/O	8 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMC への NAND 型Flashレディ /ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

16 ビット NAND Flashメモリ

表 151. 16 ビット NAND 型Flash

FMC 信号名	I/O	機能
A[17]	O	NAND 型Flashアドレスラッチイネーブル（ALE）信号
A[16]	O	NAND 型Flashコマンドラッチイネーブル（CLE）信号
D[15:0]	I/O	16 ビットマルチプレクス、双方向アドレス/データバス
NCE	O	チップセレクト
NOE(= NRE)	O	出力イネーブル（メモリ信号名：読出しイネーブル、NRE）
NWE	O	書込みイネーブル
NWAIT/INT	I	FMC への NAND 型Flashレディ /ビジー入力信号

FMC は必要なだけのアドレスサイクルを管理できるので、容量についての理論上の限界はありません。

19.7.2 NAND 型Flashによってサポートされるメモリとトランザクション

表 152 に、サポートされるデバイス、アクセスモード、およびトランザクションを示します。NAND Flashコントローラによって許可されない（またはサポートされない）トランザクションは、グレイで示されています。

表 152. サポートされるメモリおよびトランザクション

デバイス	モード	読出し／書込み	AHB データサイズ	メモリデータサイズ	設定許可／設定禁止	コメント
NAND 8 ビット	非同期	R	8	8	Y	-
	非同期	W	8	8	Y	-
	非同期	R	16	8	Y	2 つの FMC アクセスに分割
	非同期	W	16	8	Y	2 つの FMC アクセスに分割
	非同期	R	32	8	Y	4 つの FMC アクセスに分割
	非同期	W	32	8	Y	4 つの FMC アクセスに分割
NAND 16 ビット	非同期	R	8	16	Y	-
	非同期	W	8	16	N	-
	非同期	R	16	16	Y	-
	非同期	W	16	16	Y	-
	非同期	R	32	16	Y	2 つの FMC アクセスに分割
	非同期	W	32	16	Y	2 つの FMC アクセスに分割

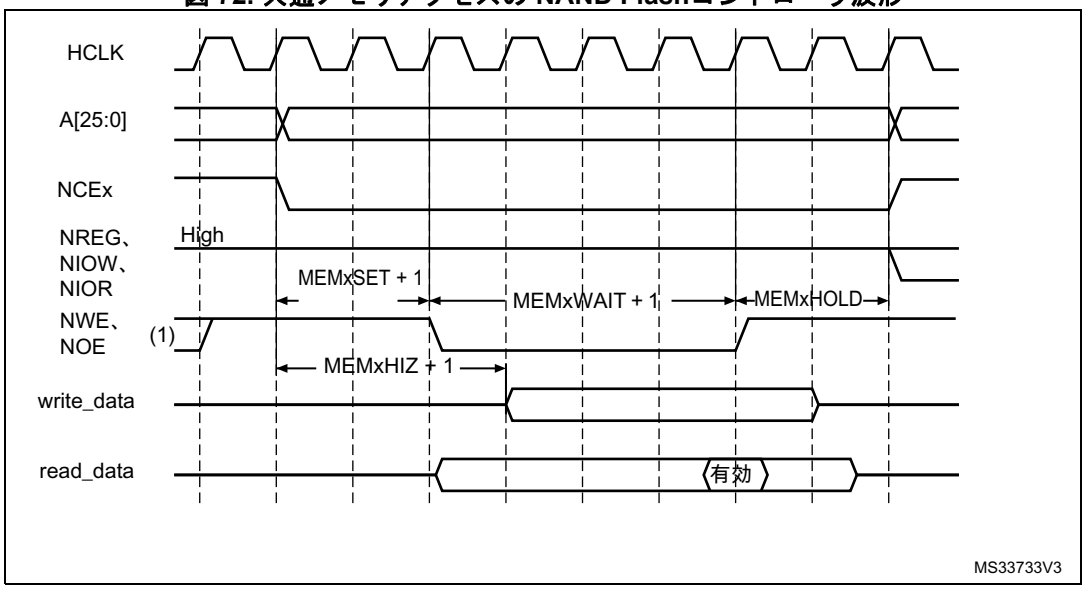
19.7.3 NAND Flashメモリのタイミング図

NAND 型Flashメモリのバンクは、以下のレジスタを通じて管理されます。

- 制御レジスタ：FMC_PCR
- 割込みステータスレジスタ：FMC_SR
- ECC レジスタ：FMC_ECCR
- 共通メモリ空間用タイミングレジスタ：FMC_PMEM
- 属性メモリ空間用タイミングレジスタ：FMC_PATT

各タイミング設定レジスタは、NAND Flashアクセスの 3 つのフェーズの HCLK サイクル数を定義するための 3 つのパラメータと、書込みアクセス時にデータバスの駆動を開始するタイミングを定義する 1 つのパラメータを格納します。図 72 に共通メモリアクセス用のタイミングパラメータの定義を示します。属性メモリ空間のアクセスタイミングが同様であることがわかります。

図 72. 共通メモリアクセスの NAND Flashコントローラ波形



1. 書込みアクセス時、NOE はハイ（非アクティブ）に保たれます。読出しアクセス時、NWE はハイ（非アクティブ）に保たれます。
2. 書込みアクセスの場合、ホールドフェーズ遅延は（MEMHOLD）HCLK サイクルとなり、読出しアクセスの場合、（MEMHOLD + 2）HCLK サイクルとなります。

19.7.4 NAND Flash動作

NAND Flashデバイスのコマンドラッチイネーブル（CLE）およびアドレスラッチイネーブル（ALE）信号は、FMC コントローラのアドレス信号によって駆動されます。これは、NAND Flashメモリにコマンドまたはアドレスを送信するには、CPU は、CPU のメモリ空間内の特定のアドレスに書込みを行う必要があることを意味します。

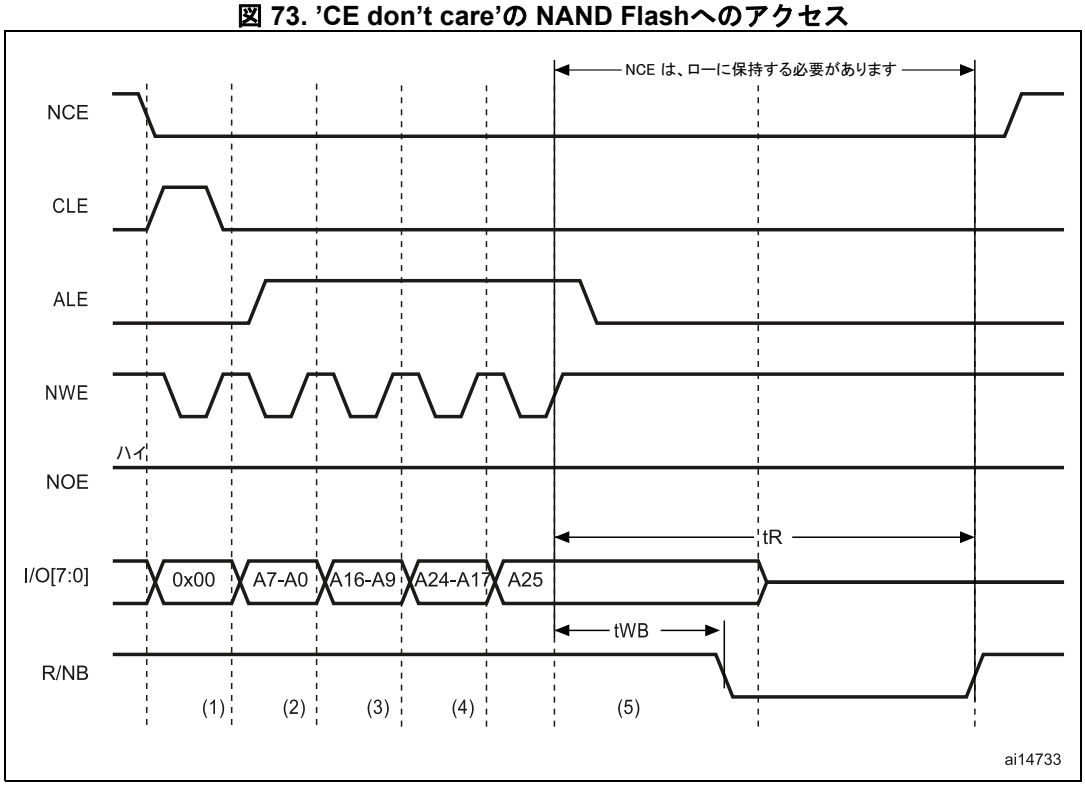
NAND Flashデバイスからの一般的なページ読出し動作には、次のステップが必要です。

1. FMC_PCR レジスタと FMC_PMEM レジスタ（デバイスによっては、FMC_PATT も。セクション 19.7.5 : NAND Flashのプリウェイト機能を参照）を、NAND Flashメモリの特性（NAND Flash のデータバス幅に対する PWID ビット、PTYP = 1、必要に応じて PWAITEN = 0 または 1、タイミング設定についてはセクション 19.5.2 : NAND Flashメモリ･アドレス･マッピングを参照）に従って設定することで、対応するメモリバンクのプログラムおよび有効化を行います。
2. CPU は、1 つのFlashコマンドバイトと等しいデータバイト（たとえば、Samsung の NAND Flashデバイスの場合は 0x00）で、共通メモリ空間にバイト書込みを行います。NAND Flash の LE 入力は、書込みストローブ（NWE 上のローパルス）の間はアクティブであり、書き込まれたバイトは、NAND Flashによってコマンドとして認識されます。コマンドがメモリデバイスにラッチされたら、次のページ読出し動作のために書き込む必要はありません。
3. CPU は、4 バイト（より小容量のデバイスの場合は 3 バイト）を共通メモリまたは属性空間に書き込むことによって、読出し操作の開始アドレス（STARTAD）を送信できます（STARTAD[7:0]、STARTAD[16:9]、STARTAD[24:17]、および 64 M ビット x 8 ビット NAND Flashの場合は最後に STARTAD[25]）。NAND 型Flashデバイスの ALE 入力は、書込みストローブの間アクティブであり（NWE 上のローパルス）、書き込まれたバイトは、読出し動作の開始アドレスとして解釈されます。属性メモリ空間を使用すると、FMC の異なるタイミング設定を使用でき、一部の NAND Flashメモリで必要とされるプリウェイト機能を実装できます（詳細については、セクション 19.7.5 : NAND Flashのプリウェイト機能を参照してください）。
4. コントローラは、NAND Flashメモリがレディ（R/NB 信号がハイ）になるのを待ってから、同じメモリバンクまたは別のメモリバンクへの新しいアクセスを開始します。待っている間、コントローラは NCE 信号をアクティブ（ロー）に保ちます。
5. CPU は、NAND Flashページ（データフィールド＋スペアフィールド）をバイト単位で読み出すために、共通メモリ空間からバイト読出し動作を行います。
6. 次の NAND Flashページは、CPU コマンドやアドレス書込み動作なしに読み出すことができます。これは次の 3 つの方法で行うことができます。
   - ステップ 5 に記述された動作を実行します。
   - ステップ 3 から動作をリスタートすることによって、新しいランダムアドレスにアクセスできます。
   - ステップ 2 からリスタートすることによって、NAND 型Flashデバイスに新しいコマンドを送信できます。

19.7.5 NAND Flashのプリウェイト機能

一部の NAND Flashデバイスでは、アドレスの最後の部分を書き込んだ後、コントローラは R/NB 信号を待つ必要があります。（図 73 を参照。）

図 73. 'CE don't care'の NAND Flashへのアクセス



1. CPU はアドレス 0x7001 0000 にバイト 0x00 を書き込みました。
2. CPU はアドレス 0x7002 0000 にバイト A7～A0 を書き込みました。
3. CPU はアドレス 0x7002 0000 にバイト A16～A9 を書き込みました。
4. CPU はアドレス 0x7002 0000 にバイト A24～A17 を書き込みました。
5. CPU はアドレス 0x7802 0000 にバイト A25 を書き込みました：FMC は FMC_PATT のタイミング定義を使用して書込みアクセスを実行します。ここで ATTHOLD ≥ 7（ただし、(7+1) * HCLK = 112 ns > t_{WB} max)。これにより、NCEは R/NB がローから再びハイになるまでローを維持します（NCE が無視されない NAND 型Flashメモリの場合のみ必要）。

この機能が必要な場合、MEMHOLD 値をプログラムすることにより t_{WB} タイミングを確実に満たすことができます。ただし、NAND Flash メモリに対するすべての CPU 読出しアクセスでは、（MEMHOLD+ 2）HCLK サイクル、CPU 書込みアクセスでは、（MEMHOLD）HCLK サイクルのホールド遅延が NWE 信号の立ち上がりエッジと次のアクセスとの間に挿入されます。

このタイミング制約に対処するためには、t_{WB} タイミングに応じた ATTHOLD 値でタイミングレジスタをプログラムし、MEMHOLD 値を最小値のままにしておくことで、属性メモリ空間を使用できます。この場合、CPU は NAND Flashのすべての読出しおよび書込みアクセスには共通メモリ空間を使用する必要があります。ただし、NAND Flashデバイスに最後のアドレスバイトを書き込むときには、属性メモリ空間に書き込む必要があります。

19.7.6 エラー訂正コード（ECC）の計算 NAND 型Flashメモリ

FMC NAND カードコントローラには、メモリバンクごとに 1 つずつ、2 つのエラー訂正コード計算ハードウェアブロックがあります。これらのハードウェアブロックは、ソフトウェアで ECC を処理する場合、ホスト CPU の負荷を削減します。

この 2 つの ECC ブロックは同一であり、それぞれバンク 2 とバンク 3 に接続されます。結果として、バンク 4 に接続されたメモリでは、ハードウェア ECC 計算を使用できません。

FMC に実装されたエラー訂正コード（ECC）のアルゴリズムでは、NAND Flashメモリに対する 256、512、1024、2048、4096、または 8192 バイトの読出しまたは書込みごとに、1 ビットのエラー訂正と 2 ビットのエラー検出を実行できます。ハミングコーディングアルゴリズムを採用し、行と列のパリティ計算を行います。

ECC モジュールは、NAND Flashメモリバンクがアクティブになるたびに、NAND Flashデータバスおよび読出し／書込み信号（NCE と NWE）を監視します。

ECC は次のように動作します。

- NAND Flashメモリのバンク 2 またはバンク 3 へアクセスするときは、D[15:0] バス上のデータがラッチされて、ECC 計算に使用されます。
- NAND Flashメモリの他のアドレスをアクセスするときは、ECC ロジックはアイドル状態となり、動作しません。このため、NAND Flashメモリに対するコマンドまたはアドレスを定義する書込み動作は、ECC 計算で考慮されません。

目的のバイト数がホスト CPU によって NAND Flashメモリから読み出されるか、NAND Flashメモリに書き込まれたら、FMC_ECCR レジスタを読み出して、計算値を取得する必要があります。読出し後は、ECCEN ビットを"0"にリセットすることによってクリアされます。新しいデータブロックを計算するには、FMC_PCR レジスタの ECCEN ビットを 1 にセットする必要があります。

ECC計算を行うときは :

1. FMC_PCR レジスタで ECCEN ビットを有効にします。
2. NAND 型Flashメモリページへデータを書き込みます。NAND ページへ書き込む際、ECC ブロックは ECC 値を計算します。
3. FMC_ECCR レジスタにある ECC 値を読み出して、変数内に格納します。
4. ECCEN ビットをクリアし、FMC_PCR レジスタで有効にした後、書き込んだデータを NAND ページから読み出します。NAND ページを読み出す際、ECC ブロックは ECC 値を計算します。
5. FMC_ECCR レジスタにある新しい ECC 値を読み出します。
6. 2 つの ECC 値が同じ場合、訂正は不要です。その他の場合、ECC エラーが存在するためソフトウェア訂正ルーチンがエラーが訂正できるか否かの情報を返します。

19.7.7 NAND Flashコントローラレジスタ

NAND Flash制御レジスタ（FMC_PCR）

アドレスオフセット：0x80

リセット値：0x0000 0018

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS[2:0]			TAR3
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAR[2:0]			TCLR[3:0]				Res.	Res.	ECCEN	PWID[1:0]		PTYP	PBKEN	PWAITEN	Res.
rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 **ECCPS[2:0]**：ECC ページ･サイズ

拡張 ECC のページ･サイズを定義します。

000：256 バイト

001：512 バイト

010：1024 バイト

011：2048 バイト

100：4096 バイト

101：8192 バイト

ビット 16:13 **TAR[3:0]**：ALE から RE 間の遅延

ALE ローから RE ローまでの時間を AHB クロック･サイクル（HCLK）数で設定します。
時間は、t_ar =（TAR + SET +2）× THCLK です。THCLK は HCLK クロック周期です。

0000：1 HCLK サイクル（デフォルト）

1111：16 HCLK サイクル

注： SET は、アドレス指定される空間に応じて、MEMSET または ATTSET です。

ビット 12:9 **TCLR[3:0]**：CLE から RE 間の遅延

CLE ローから RE ローまでの時間を AHB クロック･サイクル（HCLK）数で設定します。
時間は、t_clr = (TCLR + SET + 2) * THCLK です。THCLK は HCLK クロック周期です。

0000：1 HCLK サイクル（デフォルト）

1111：16 HCLK サイクル

注： SET は、アドレス指定される空間に応じて、MEMSET または ATTSET です。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **ECCEN**：ECC 計算ロジック･イネーブル･ビット

0：ECC ロジックは無効であり、リセットされています（リセット後のデフォルト）。

1：ECC ロジックは有効です。

ビット 5:4 **PWID[1:0]**：データバス幅

外部メモリ･デバイスの幅を定義します。

00：8 ビット

01：16 ビット（リセット後のデフォルト）

10：予約済み

11：予約済み

ビット 3 **PTYP**：メモリ･タイプ

対応するメモリバンクに接続されたデバイスのタイプを定義します。

0：予約済みであり、リセット値に保持する必要があります。

1：NAND 型 Flash（リセット後のデフォルト）

ビット 2 **PBKEN**：NAND Flash メモリ･バンク･イネーブル･ビット

メモリ･バンクを有効にします。無効なメモリ･バンクにアクセスすると、AHB バスでエラーが発生します。

0：対応するメモリバンクは無効です（リセット後のデフォルト）。

1：対応するメモリバンクは有効です。

ビット 1 **PWAITEN**：ウェイト機能イネーブル･ビット

NAND 型 Flash メモリ･バンクのウェイト機能を有効にします。

0：無効

1：有効

ビット 0 予約済みであり、リセット値に保持する必要があります。

FIFO ステータスおよび割込みレジスタ（FMC_SR）

アドレスオフセット：0x84

リセット値：0x0000 0040

このレジスタは、FIFO ステータスと割込みに関する情報を格納します。FMC には、AHB の 16 ワードまでのデータをメモリに書き込んで格納するときに使用される FIFO があります。

これは、FMC が FIFO をメモリへ転送すると同時に、FIFO に迅速に書き込んで、FMC 以外のペリフェラルへのトランザクションに AHB を解放するために使用されます。これらのレジスタビットの 1 つは、ECC のために、FIFO のステータスを示します。

ECC は、データのメモリへの書込み中に計算されます。正しい ECC を読み出すため、ソフトウェアは FIFO がエンプティになるまで待つ必要があります。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
									r	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FEMPT**：FIFO エンプティ

FIFO のステータスを示す読取り専用ビット

0：FIFO は空ではありません。

1：FIFO エンプティ

ビット 5 **IFEN**：割込み立ち下がりエッジ検出イネーブル･ビット

0：割込み立ち下がりエッジ検出リクエストは無効

1：割込み立ち下がりエッジ検出リクエストは有効

ビット 4 **ILEN**：割込みハイレベル検出イネーブル･ビット

0：割込みハイレベル検出リクエストは無効

1：割込みハイレベル検出リクエストは有効

ビット 3 **IREN**：割込み立ち上がりエッジ検出イネーブル･ビット

0：割込み立ち上がりエッジ検出リクエストは無効

1：割込み立ち上がりエッジ検出リクエストは有効

ビット 2 **IFS**：割込み立ち下がりエッジ･ステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立ち下がりエッジは発生しませんでした。

1：割込み立ち下がりエッジが発生しました。

注： ソフトウェアでこのビットに 1 を書き込むとセット状態になります。

ビット 1 **ILS**：割込みハイレベル･ステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込みハイレベルは発生しませんでした。

1：割込みハイレベルが発生しました。

ビット 0 **IRS**：割込み立ち上がりエッジ･ステータス

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。

0：割込み立ち上がりエッジは発生しませんでした。

1：割込み立ち上がりエッジが発生しました。

注： ソフトウェアでこのビットに 1 を書き込むとセット状態になります。

共通メモリ空間タイミング･レジスタ（FMC_PMEM）

アドレスオフセット：アドレス：0x88

リセット値：0xFCFC FCFC

FMC_PMEM 読出し／書込みレジスタは、NAND Flash メモリ･バンク のタイミング情報を格納します。この情報は、NAND Flashの共通メモリ空間に、コマンド、アドレスの書込みアクセス、およびデータ読出し／書込みアクセスするために使われます。

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

ビット 31:24 **MEMHIZ[7:0]**：共通メモリ x データ･バス･ハイ･インピーダンス時間

ソケット上の共通メモリ空間への NAND Flashによる書込みアクセスの開始後に、データバスをハイインピーダンスに保つ HCLK クロックサイクル数を定義します。これは、書込み トランザクションに対してのみ有効です。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ビット 23:16 **MEMHOLD[7:0]**：共通メモリホールド時間

ソケット x 上の共通メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書込みアクセス時はデータも）保持する書込みアクセス時の HCLK クロックサイクル数と読出しアクセス時の HCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：書込みアクセスの場合 1 HCLK サイクル／読出しアクセスの場合 3 HCLK サイクル

1111 1110：書込みアクセスの場合 254 HCLK サイクル／読出しアクセスの場合 256 HCLK サイクル

1111 1111：予約済み

ビット 15:8 **MEMWAIT[7:0]：**共通メモリウェイト時間

ソケット上の共通メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンド（NWE、NOE）をアサートするための HCLK（＋1）クロックサイクルの最小数を定義します。プログラムされた HCLK 値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1110：255 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **MEMSET[7:0]：**共通メモリ x セットアップ時間

ソケット x 上の共通メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンドアサーション（NWE、NOE）前にアドレスをセットアップするための HCLK（+1）クロックサイクル数を定義します。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

属性メモリ空間タイミング･レジスタ（FMC_PATT）

アドレスオフセット：0x8C

リセット値：0xFCFC FCFC

FMC_PATT 読出し／書込みレジスタは、NAND Flash メモリ･バンク のタイミング情報を格納します。この情報は、最後のアドレス書込みアクセスの際に、タイミングが以前のアクセスのタイミングと異なる必要がある場合に、NAND Flashの属性メモリ空間へ 8 ビットアクセスするために使用されます（レディ／ビジー管理については、セクション 19.7.5：NAND Flashのプリウェイト機能を参照）。

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

ビット 31:24 **ATTHIZ[7:0]：**属性メモリ･データ･バス･ハイ･インピーダンス時間

ソケット上の属性メモリ空間への NAND Flashによる書込みアクセスの開始後に、データバスをハイインピーダンスに保つ HCLK クロックサイクル数を定義します。これは、書込みトランザクションに限り有効です。

0000 0000：0 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ビット 23:16 **ATTHOLD[7:0]：**属性メモリホールド時間

ソケット上の属性メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンドのネゲート（NWE、NOE）後にアドレスを（書込みアクセス時はデータも）保持する書込みアクセス時の HCLK クロックサイクル数と読出しアクセス時の HCLK（+2）クロックサイクル数を定義します。

0000 0000：予約済み

0000 0001：書込みアクセスの場合 1 HCLK サイクル／読出しアクセスの場合 3 HCLK サイクル

1111 1110：書込みしアクセスの場合 254 HCLK サイクル／読出しアクセスの場合 256 HCLK サイクル

1111 1111：予約済み

ビット 15:8 **ATTWAIT[7:0]**：属性メモリウェイト時間

ソケット x 上の属性メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンド（NWE、NOE）をアサートするための HCLK（＋1）クロックサイクルの最小数を定義します。プログラムされた HCLK 値の後にウェイト信号（NWAIT）がアクティブ（ロー）の場合、コマンドアサーションの時間は延長されます。

0000 0000：予約済み

0000 0001：2 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1110：255 HCLK サイクル（＋NWAIT のデアサーションによって導入されるウェイトサイクル）

1111 1111：予約済み

ビット 7:0 **ATTSET[7:0]**：属性メモリ･セットアップ時間

ソケット上の属性メモリ空間への NAND Flashによる読出しまたは書込みアクセスの際に、コマンドアサーション（NWE、NOE）前にアドレスをセットアップするための HCLK（＋1）クロックサイクル数を定義します。

0000 0000：1 HCLK サイクル

1111 1110：255 HCLK サイクル

1111 1111：予約済み

ECC 結果レジスタ（FMC_ECCR）

アドレスオフセット：0x94

リセット値：0x0000 0000

このレジスタは、FMC NAND コントローラの ECC 計算モジュールによって計算された現在のエラー訂正コード値を格納します。CPU が、NAND Flashメモリページ上の正しいアドレスからデータを読み出すときには（セクション 19.7.6：エラー訂正コード（ECC）の計算 NAND 型Flashメモリを参照）、NAND Flashメモリに対して読み書きされたデータは、ECC 計算モジュールによって自動的に処理されます。X バイトを読み出した後（FMC_PCR レジスタの ECCPS フィールドに従って）、CPU は FMC_ECC レジスタから計算された ECC 値を読み出す必要があります。次に、これらの計算されたパリティデータがスペア領域に記録されているパリティ値と一致するか検証して、ページが有効か否か、訂正が必要か否かを確認します。FMC_ECCR レジスタは、読出し後、ECCEN ビットを "0" に設定することによってクリアする必要があります。新しいデータ･ブロックを計算するときは、ECCEN ビットを "1" に設定する必要があります。

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

ビット 31:0 **ECC[31:0]**：ECC 結果

このフィールドは、ECC 計算ロジックによって計算された値を格納します。表 153 に、これらのビットフィールドの値を示します。

表 153. ECC 結果関連ビット

ECCPS[2:0]	ページサイズ（バイト数）	ECC ビット
000	256	ECC[21:0]
001	512	ECC[23:0]
010	1024	ECC[25:0]

表 153. ECC 結果関連ビット （続き）

ECCPS[2:0]	ページサイズ（バイト数）	ECC ビット
011	2048	ECC[27:0]
100	4096	ECC[29:0]
101	8192	ECC[31:0]

19.7.8 FMC レジスタ･マップ

表 154. FMCレジスタ･マップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	FMC_BCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		WFDIS	CCLKEN	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値									0	0	0	0	0	0	0	0	0	0	1	1	0		0	0		1	0	1	1	0	1	1
0x08	FMC_BCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値									0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x10	FMC_BCR3	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値									0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x18	FMC_BCR4	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NBL SET [1:0]		Res.	Res.	CBURSTRW	CPSIZE [2:0]			ASYNCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	Res.	WAITPOL	BURSTEN	Res.	FACCEN	MWID [1:0]		MTYP [1:0]		MUXEN	MBKEN
	リセット値									0	0			0	0	0	0	0	0	1	1	0		0	0		1	0	1	0	0	1	0
0x04	FMC_BTR1	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0C	FMC_BTR2	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x14	FMC_BTR3	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x1C	FMC_BTR4	DATAHLD[1:0]		ACCMOD[1:0]		DATLAT[3:0]				CLKDIV[3:0]				BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x20	FMC_ PCSCNTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTB4EN	CNTB3EN	CNTB2EN	CNTB1EN	CSCOUNT[15:0]															
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 154. FMCレジスタ･マップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x104	**FMC_BWTR1**	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x10C	**FMC_BWTR2**	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x114	**FMC_BWTR3**	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x11C	**FMC_BWTR4**	DATAHLD[1:0]		ACCMOD[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSTURN [3:0]				DATAST[7:0]								ADDHLD[3:0]				ADDSET[3:0]			
	リセット値	0	0	0	0									1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x80	**FMC_PCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ECCPS [2:0]				TAR[3:0]				TCLR[3:0]				Res.	Res.	ECCEN	PWID [1:0]		PTYP	PBKEN	PWAITEN	Res.
	リセット値													0	0	0	0	0	0	0	0	0	0	0			0	0	1	1	0	0	
0x84	**FMC_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FEMPT	IFEN	ILEN	IREN	IFS	ILS	IRS
	リセット値																										1	0	0	0	0	0	0
0x88	**FMC_PMEM**	MEMHIZx[7:0]								MEMHOLDx[7:0]								MEMWAITx[7:0]								MEMSETx[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x8C	**FMC_PATT**	ATTHIZ[7:0]								ATTHOLD[7:0]								ATTWAIT[7:0]								ATTSET[7:0]							
	リセット値	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	1	0	0
0x94	**FMC_ECCR**	ECCx[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

20 Quad SPI インタフェース（QUADSPI）

20.1 概要

QUADSPI は、シングル、デュアル、またはQuad SPI Flashメモリを対象とする特殊な通信インタフェースです。このインタフェースは、次の 3 つのモードのいずれかで動作できます。

- インダイレクトモード：すべての動作は QUADSPI レジスタを使用して実行されます。
- ステータスポーリングモード：外部Flashメモリステータスレジスタが定期的に読み出され、フラグのセット時に割込みを生成することができます。
- メモリマップドモード:外部Flashメモリはデバイスのアドレス空間に配置され、システムによって内部メモリであるかのようにみなされます。

デュアルFlashモードを使用すると、スループットと容量の両方を 2 倍に増やすことができます。このモードでは、2 つのQuad SPI Flashメモリに同時にアクセスします。

20.2 QUADSPI の主な機能

- 3 つの機能モード：インダイレクト、ステータスポーリング、メモリマップド
- 2 つのFlashメモリに並列にアクセスすることで 8 ビットの同時送受信を可能にするデュアルFlashモード
- SDR および DDR サポート
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能な OP コード
- インダイレクトモードとメモリマップドモードの両方で完全にプログラム可能なフレームフォーマット
- 受信および送信用の内蔵 FIFO
- 8、16、32 ビットのデータアクセスを許可
- インダイレクトモードで動作するための DMA チャネル
- FIFO 閾値、タイムアウト、動作完了、アクセスエラーによる割込み生成

20.3 QUADSPI の機能説明

20.3.1 QUADSPI ブロック図

図 74. QUADSPI ブロック図（デュアルFlashモードが無効のとき）

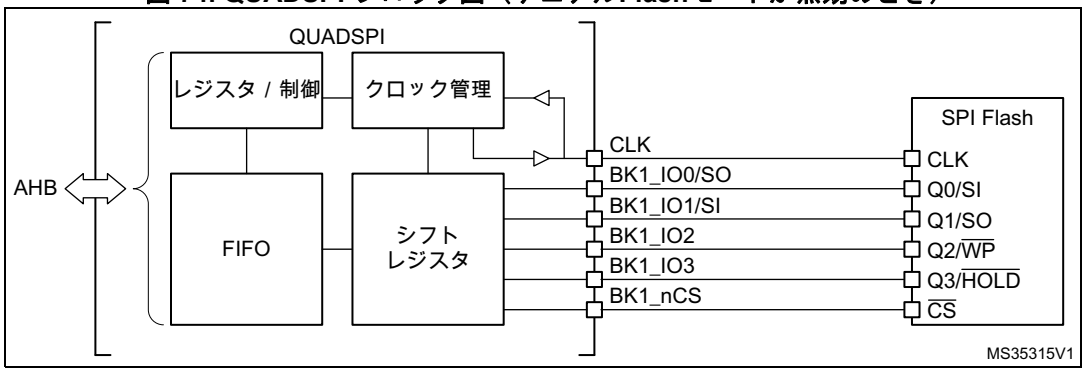
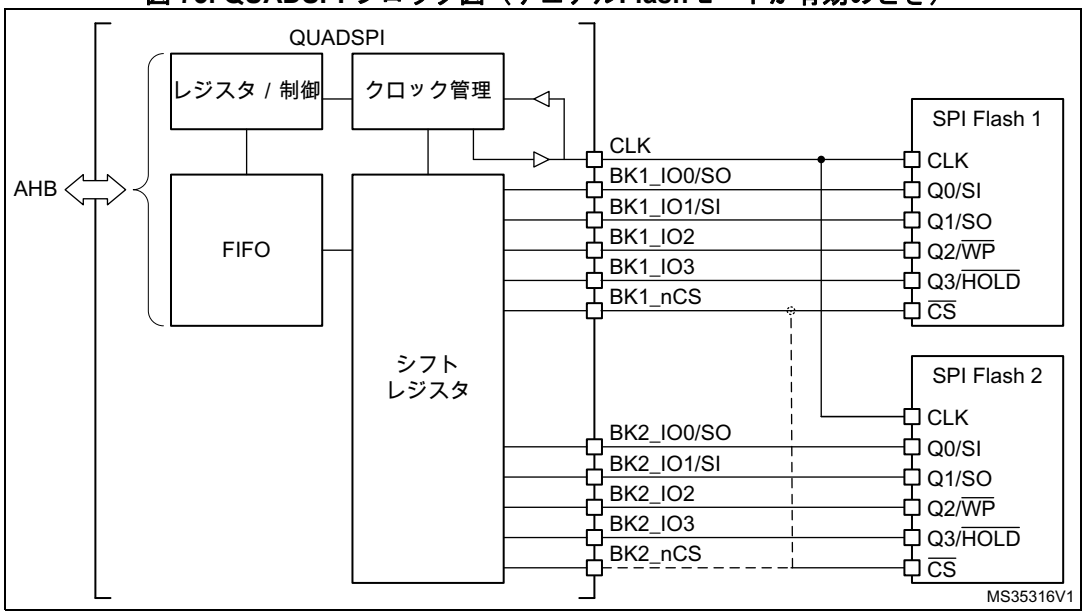


図 75. QUADSPI ブロック図（デュアルFlashモードが有効のとき）



20.3.2 QUADSPI ピン

下表は QUADSPI ピンの一覧で、そのうち 6 本 はシングル Flash メモリとのインタフェース用、10 または 11 本はデュアル Flash モードの 2 つの Flash メモリ（Flash 1 および Flash 2）とのインタフェース用です。

表 155. QUADSPI ピン

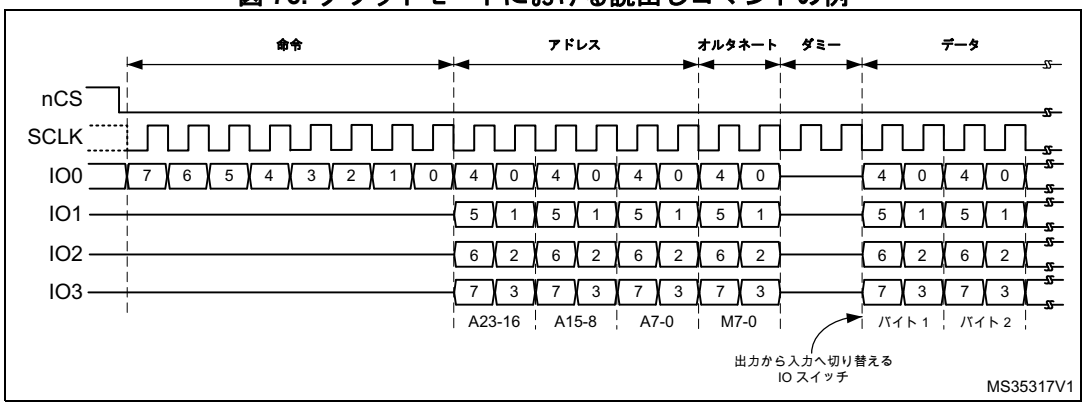
信号名	信号タイプ	説明
CLK	デジタル出力	Flash 1 および Flash 2 へのクロック
BK1_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（Flash 1 のみ）
BK1_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（Flash 1 のみ）
BK1_IO2	デジタル入出力	クワッドモードでの双方向入出力（Flash 1 のみ）
BK1_IO3	デジタル入出力	クワッドモードでの双方向入出力（Flash 1 のみ）
BK2_IO0/SO	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル出力（Flash 2 のみ）
BK2_IO1/SI	デジタル入出力	デュアル／クワッドモードでの双方向入出力またはシングルモードでのシリアル入力（Flash 2 のみ）
BK2_IO2	デジタル入出力	クワッドモードでの双方向入出力（Flash 2 のみ）
BK2_IO3	デジタル入出力	クワッドモードでの双方向入出力（Flash 2 のみ）
BK1_nCS	デジタル出力	チップセレクト出力（アクティブロー）（Flash 1 のみ）。デュアルFlashモードで QUADSPI が常時使用される場合は、Flash 2 にも使用可能
BK2_nCS	デジタル出力	チップセレクト出力（アクティブロー）（Flash 2 のみ）。デュアルFlashモードで QUADSPI が常時使用される場合は、Flash 1 にも使用可能

20.3.3 QUADSPI コマンドシーケンス

QUADSPI はコマンドを使用してFlashメモリと通信します。各コマンドには、命令、アドレス、オルタネート･バイト、ダミー、データの 5 つのフェーズが含まれます。これらのフェーズのいずれかをスキップするように設定することができますが、命令、アドレス、オルタネート･バイト、データの少なくとも 1 つのフェーズが必要です。

nCS は各コマンドの開始前に立ち下がり、各コマンドの終了後に再び立ち上がります。

図 76. クワッドモードにおける読出しコマンドの例



命令フェーズ

このフェーズでは、QUADSPI_CCR[7:0] レジスタの INSTRUCTION フィールドに設定された 8 ビットの命令がFlashメモリに送信され、実行すべき動作のタイプが指定されます。

ほとんどのFlashメモリは IO0/SO 信号（シングル SPI モード）から同時に 1 ビットの命令しか受信できませんが、命令フェーズでは、オプションで、同時に 2 ビット（デュアル SPI モードの IO0/IO1 経由）または 4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）を送信することができます。この設定には、QUADSPI_CCR[9:8] レジスタの IMODE[1:0] フィールドを使用することができます。

IMODE = 00 の場合、命令フェーズはスキップされますので、コマンドシーケンスはアドレスフェーズが存在する場合はそこから開始されます。

アドレスフェーズ

アドレスフェーズでは、動作のアドレスを示す 1～4 バイトがFlashメモリに送信されます。送信されるアドレスバイト数は、QUADSPI_CCR[13:12] レジスタの ADSIZE[1:0] フィールドに設定されます。インダイレクトモードと自動ポーリングモードでは、送信されるアドレスバイトは QUADSPI_AR レジスタの ADDRESS[31:0] フィールドに指定されます。一方、メモリマップドモードの場合は、アドレスはAHB（Cortex® から、または DMA から）経由で直接与えられます。

アドレスフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[11:10] レジスタの ADMODE[1:0] フィールドを使用することができます。

ADMODE = 00 の場合、アドレスフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

オルタネートバイトフェーズ

オルタネートバイトフェーズでは、一般に動作モードを制御する 1〜4 バイトがFlashメモリに送信されます。送信されるオルタネートバイト数は、QUADSPI_CCR[17:16] レジスタの ABSIZE[1:0] フィールドに設定されます。送信されるバイトは QUADSPI_ABR レジスタで指定されます。

オルタネートバイトフェーズで同時に送信できるのは、1 ビット（シングル SPI モードの SO 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

ABMODE = 00 の場合、オルタネートバイトフェーズはスキップされますので、コマンドシーケンスは次のフェーズが存在する場合は直接そこに進みます。

場合によっては、オルタネートバイトフェーズ中にフルバイトではなく 2 分の 1 バイト（4 ビット）を送信する必要があるかもしれません。たとえば、デュアルモード使用時で、オルタネートバイトに 2 サイクルだけが使われる場合などです。この場合、ファームウェアによって Quad モード（ABMODE=11）を使用し、ALTERNATE のビット 7 および 3 を“1”（IO3 ラインをハイで保持）に、ビット 6 および 2 を“0”（IO2 ラインをローで保持）にセットしてバイトを送信できます。送信される 4 ビットの上位 2 ビットは ALTERNATE のビット 4:3 にセットされ、下位 2 ビットはビット 1 および 0 にセットされます。例えば、4bitの値2（0010）が IO0/IO1 で送信される場合、ALTERNATE は 0x8A（1000_1010）にセットされている必要があります。

ダミーサイクルフェーズ

ダミーサイクルフェーズでは、データの送受信は一切なく、1〜31 サイクルが実行されます。これは、より高いクロック周波数が使われいてる場合に、データフェーズの準備をする時間をFlashメモリに与えるためです。このフェーズ中に実行されるサイクル数は、QUADSPI_CCR[22:18] レジスタの DCYC[4:0] フィールドで指定されます。SDR モードと DDR モードのどちらでも、この持続時間はフル CLK サイクル数として指定されます。

DCYC = 00 の場合、ダミーサイクルフェーズはスキップされますので、コマンドシーケンスはデータフェーズが存在する場合は直接そこに進みます。

ダミーサイクルフェーズの動作モードは DMODE によって決まります。

データ信号を出力モードから入力モードへ変更するのに十分な「ターンアラウンド」時間を確保するために、デュアルまたはクワッドモードを使用してFlashメモリからデータを受信する場合、少なくとも 1 ダミーサイクルが必要です。

データフェーズ

データフェーズ中は、Flashメモリに対して何バイトでも送受信することができます。

インダイレクトモードと自動ポーリングモードでは、送受信するバイト数は QUADSPI_DLR レジスタで指定されます。

インダイレクト書込みモードでは、Flashメモリに送信されるデータを QUADSPI_DR レジスタに書き込む必要があります。一方、インダイレクト読出しモードでは、Flashメモリから受信するデータは QUADSPI_DR レジスタから読み出して取得します。

メモリマップドモードでは、読み出されるデータは、直接 AHB を経由して Cortex または DMA へ送り戻されます。

データフェーズで同時に送受信できるのは、1 ビット（シングル SPI モードの SO/SI 経由）、2 ビット（デュアル SPI モードの IO0/IO1 経由）、または4 ビット（Quad SPI モードの IO0/IO1/IO2/IO3 経由）です。この設定には、QUADSPI_CCR[15:14] レジスタの ABMODE[1:0] フィールドを使用することができます。

DMODE = 00 の場合、データフェーズはスキップされますので、コマンドシーケンスは nCS を立ち上げることですぐに終了します。この設定はインダイレクト書込みモードのみで使用してください。

20.3.4 QUADSPI シングルインタフェースプロトコルモード

シングル SPI モード

レガシー SPI モードでは、1 ビットのシリアル送受信のみが可能です。このモードでは、データは SO 信号（I/O を IO0 と共有）を経由してFlashメモリへ送信されます。Flashメモリからのデータは、SI 信号（I/O を IO1 と共有）を経由して受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“01”にセットすることで、このシングルビットモードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをシングルモードに設定する場合

- IO0（SO）は出力モードです。
- IO1（SI）は入力モードです（ハイインピーダンス）。
- IO2 は出力モードで、強制的に 0 に設定されます。
- IO3 は出力モードで、強制的に 1 に設定されます（ホールド機能を無効にするため）。

これは、DMODE = 01 であれば、ダミー･フェーズにも該当します。

デュアル SPI モード

デュアル SPI モードでは、IO0/IO1 信号を経由して 2 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“10”にセットすることで、このデュアル SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

各フェーズをデュアルモードに設定する場合

- IO0/IO1 は、データフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。
- IO2 は出力モードで、強制的に 0 に設定されます。
- IO3 は出力モードで、強制的に 1 に設定されます。

ダミー･フェーズでは、DMODE = 01 の場合の IO0/IO1 は必ずハイインピーダンスです。

Quad SPI モード

Quad SPI モードでは、IO0/IO1/IO2/IO3 信号を経由して 4 ビットが同時に送受信されます。

QUADSPI_CCR レジスタのそれぞれのフェーズに対応するフィールド（IMODE/ADMODE/ABMODE/DMODE）を“11”にセットすることで、このQuad SPI モードを使用できるよう、フェーズごとに個別に設定することができます。

クワッドモードに設定されている各フェーズでは、IO0/IO1/IO2/IO3 はすべてデータフェーズの読出し動作時はハイインピーダンス（入力）ですが、他のすべての場合は出力です。

ダミーフェーズでは、DMODE = 11 の場合、IO0/IO1/IO2/IO3 はすべて常にハイインピーダンスです。

IO2 と IO3 はQuad SPI モードでのみ使用されます。Quad SPI モードを使用できるように設定されたフェーズが 1 つもない場合、IO2 および IO3 に対応するピンは QUADSPI がアクティブなときでも、他の機能に使用することができます。

SDR モード

DDRM ビット（QUADSPI_CCR[31]）は、デフォルトで 0 に設定されており、QUADSPI はシングルデータレート（SDR）モードで動作します。

SDR モードでは、QUADSPI が IO0/SO、IO1、IO2、IO3 信号を駆動している場合、これらの信号は CLK の立ち下がりエッジでのみ遷移します。

SDR モードでデータを受信する場合、QUADSPI はFlashメモリも CLK の立ち下がりエッジを使用してデータを送信するとみなします。デフォルトでは（SSHIFT = 0 の場合）、信号は次の CLK の（立ち上がり）エッジを使用してサンプリングされます。

DDR モード

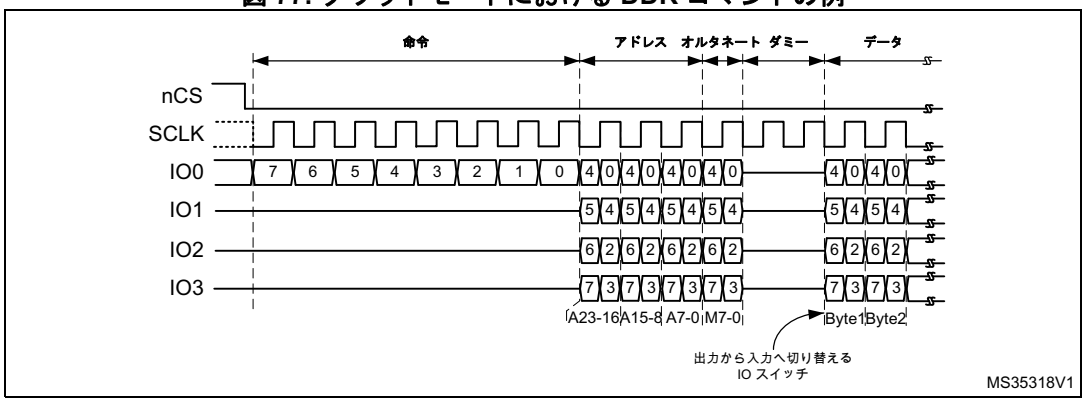
DDRM ビット（QUADSPI_CCR[31]）が 1 にセットされている場合、QUADSPI はダブルデータレート（DDR）モードで動作します。

DDR モードでは、QUADSPI がアドレス／オルタネートバイト／データフェーズで IO0/SO、IO1、IO2、IO3 信号を駆動している場合、CLK の立ち下がりエッジおよび立ち上がりエッジのそれぞれでビットを送信します。

命令フェーズは DDRM の影響は受けません。命令は常に CLK の立ち下がりエッジを使用して送信されます。

DDR モードでデータを受信する場合、QUADSPI はFlashメモリも CLK の立ち上がりエッジと立ち下がりエッジの両方を使用してデータを送信するとみなします。DDRM = 1 の場合、ファームウェアは SSHIFT ビット（QUADSPI_CR のビット 4）をクリアする必要があります。したがって、信号は 1/2 CLK サイクル後にサンプリングされます（下図の逆エッジで）。

図 77. クワッドモードにおける DDR コマンドの例



デュアルFlashモード

DFM ビット（QUADSPI_CR のビット 6）が 1 の場合、QUADSPI はデュアルFlashモードです。このモードでは、サイクルごとに 8ビットの送受信（DDR モードでは 16 ビット）を行うために 2 つの外部 Quad SPI Flashメモリ（Flash 1 と Flash 2）が使用され、スループットおよび容量を効果的に倍にします。

どちらのFlashメモリも同じ CLK を使用し、オプションで同じ nCS 信号を使用しますが、それぞれが個別の IO0、IO1、IO2、IO3 信号を有しています。

デュアルFlashモードは、SDR　モードまたは DDR モードと組み合わせて使用できるだけでなく、シングルビットモード、デュアルビットモード、クワッドビットモードとも組み合わせて使用できます。

FSIZE[4:0]（QUADSPI_DCR[20:16]）で指定されているFlashメモリのサイズは、Flashメモリの総容量（コンポーネント 1 個のサイズの 2 倍）を反映していなければなりません。

アドレス X が偶数の場合、QUADSPI はアドレス X に対してFlash 1 のアドレス X/2 にあるバイトを与え、アドレス X+1 に対してFlash 2 のアドレス X/2 にあるバイトを与えます。言い換えれば、偶数アドレスにあるバイトはすべてFlash 1 に、奇数アドレスにあるバイトはすべてFlash 2 に格納されます。

デュアルFlashモードでFlashメモリステータスレジスタを読み出す場合、シングルFlashモードで読み出すバイト数の2 倍のバイトを読み出す必要があります。つまり、各 Flash メモリがステータス･レジスタのフェッチ命令後に有効な 8 ビットを提供する場合、QUADSPI はデータ長 2 バイト（16 ビット）で設定する必要があり、それにより QUADSPI は各 Flash メモリから 1 バイトを受信します。各 Flash メモリが 16 ビットのステータスを提供する場合、QUADSPI は、4 バイトを読み出して、デュアル Flash モードの両方の Flash メモリのステータス･ビットをすべて取得できるように設定する必要があります。リザルト（データレジスタ内）の最下位バイトはFlash 1 ステータスレジスタの最下位バイト、次のバイトはFlash 2 ステータスレジスタの最下位バイトとなります。さらに、データレジスタの 3 番目のバイトはFlash 1 の 2 番目のバイト、4 番目のバイトはFlash 2 の 2 番目のバイトとなります（Flashメモリに 16 ビットステータスレジスタがある場合）。

偶数バイトは常にデュアルFlashモードでアクセスする必要があります。このため、DRM = 1 のとき、データ長フィールドのビット 0 （QUADSPI_DLR[0]）は 1 のままとなります。

デュアルFlashモードでは、Flash 1 インタフェース信号の動作は基本的に通常モードの動作と同じです。Flash 2 インタフェース信号の波形は、命令、アドレス、オルタネートバイト、およびダミーサイクルフェーズの間は、Flash 1 とまったく同じです。つまり、どちらのFlashメモリも常に同じ命令や同じアドレスを受信するということです。データフェーズ中は、2 つのバス BK1_IOx と BK2_IOx は並列してデータ転送を行いますが、Flash 1 に対する送信（または受信）データはFlash 2 のものとは異なります。

20.3.5 QUADSPI インダイレクトモード

インダイレクトモードでは、QUADSPI レジスタに書き込むことでコマンドが開始され、データレジスタへの書込み／読出しによってデータが転送されます。その方法はその他の通信ペリフェラルの場合と同様です。

FMODE = 00（QUADSPI_CCR[27:26]）の場合、QUADSPI はインダイレクト書込みモードです。このモードでは、データフェーズ中にFlashメモリへのバイトの送信が行われます。データは、データレジスタ（QUADSPI_DR）への書込みによって提供されます。

FMODE = 01 の場合、QUADSPI はインダイレクト読出しモードです。このモードでは、データフェーズ中にFlashメモリからのバイトの受信が行われます。データは、QUADSPI_DR の読出しによって復旧されます。

送受信するバイト数はデータ長レジスタ（QUADSPI_DLR）で指定されます。
QUADSPI_DLR = 0xFFFF_FFFF（すべて”1”）の場合、データ長は不定長と見なされ、QUADSPI はFlashメモリの終了（FSIZE によって定義される）までデータを転送し続けるのみです。転送されるバイトが無い場合は、DMODE（QUADSPI_CCR[25:24]）を”00”にセットする必要があります。

QUADSPI_DLR = 0xFFFF_FFFF かつ FSIZE = 0x1F（4 GB のFlashメモリを示す最大値）の場合、このような特殊なケースでは、転送は無限に続けられ、アボートリクエスト後または QUADSPI の無効化後にやっと停止します。最後のメモリアドレスが読み出された後も（アドレス 0xFFFF_FFFF で）、アドレス = 0x0000_0000 で読出しが続けられます。

プログラムされた送信または受信バイト数に達すると、TCF がセットされ、TCIE = 1 であれば割込みが生成されます。データ数が定義されていない場合、外部 SPI メモリの限界に達した時点で、QUADSPI_CR に定義されているFlashメモリのサイズに応じて TCF がセットされます。

コマンド開始のトリガ

コマンドは、基本的に、ファームウェアがそのコマンドに必要な最期の情報を提供するとすぐに開始されます。インダイレクトモードでコマンドを開始する方法は、QUADSPI の設定に応じて、3 通りあります。コマンドは以下の動作の直後に開始されます。

1. アドレスを必要としない場合（ADMODE = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、INSTRUCTION[7:0]（QUADSPI_CCR）への書込み動作直後
2. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要としない場合（FMODE = 01 または DMODE = 00）は、ADDRESS[31:0]（QUADSPI_AR）への書込み動作直後
3. アドレスを必要とする場合（ADMODE ! = 00）およびファームウェアによるデータの提供を必要とする場合（FMODE = 00 および DMODE ! = 00）は、DATA[31:0]（QUADSPI_DR）への書込み動作直後

オルタネートバイトレジスタ（QUADSPI_ABR）への書込みが通信開始をトリガすることはありません。オルタネートバイトが必要である場合、事前にプログラムしておく必要があります。

コマンドが開始されたらすぐに、BUSY ビット（QUADSPI_SR の ビット5）が自動的にセットされます。

FIFO およびデータの管理

インダイレクトモードでは、データは QUADSPI に内蔵されている 16バイト FIFO を通ります。FLEVEL[4:0]（QUADSPI_SR[12:8]）は、現在 FIFO に保持されているバイト数を示します。

インダイレクト書込みモード（FMODE = 00）では、ファームウェアは QUADSPI_DR に書き込む時点で FIFO にデータを追加します。書込みにより FIFO に追加されるバイト数は、ワード書込みで 4 バイト、ハーフワード書込みで 2 バイト、バイト書込みで 1 バイトです。ファームウェアが FIFO に追加するバイト数が多すぎる（DL[31:0] によって示される数より多い）場合、余分なバイトは、書込み動作の終了時（TCF がセットされるとき）に FIFO から一掃されます。

QUADSPI_DR へのバイト／ハーフワードアクセスは、32 ビットレジスタの最下位バイト／ハーフワードにのみ実行されます。

FTHRES[3:0] を使用して FIFO 閾値が定義されます。閾値に達すると、FTF（FIFO 閾値フラグ）がセットされます。インダイレクト読出しモードでは、FIFO から読み出される有効バイト数が閾値を超えると FTF がセットされます。また、FTHRES の設定に関わらず、Flashメモリから最後のバイトが読み出された後に FIFO 内にデータが残っている場合も FTF がセットされます。インダイレクト書込みモードでは、FIFO 内の空のバイト数が閾値を超えると FTF がセットされます。

FTIE = 1 の場合、FTF がセットされた時点で割込みが発生します。DMAEN = 1 の場合、FTF がセットされた時点で DMA 転送が開始されます。閾値条件が真でなくなる（CPU または DMA によって十分なデータが転送された後）とすぐに、ハードウェアによって FTF がクリアされます。

20.3.6 QUADSPI ステータス･フラグのポーリング･モード

自動ポーリングモードでは、QUADSPI は定期的にコマンドを開始させて、定義された数のステータスバイト（最大 4）を読み出します。受信バイトをマスクして、ステータスビットの一部を分離することができます。また、選択されたビットに値が定義されている場合は割込みを生成することができます。

Flashメモリへのアクセス方法はインダイレクト読出しモードの場合と同様です。アドレスを必要としない場合（AMODE = 00）は、QUADSPI_CCR が書き込まれるとすぐにアクセスが開始されます。反対にアドレスを必要とする場合は、QUADSPI_AR が書き込まれた時点で最初のアクセスが始まります。この時点で BUSY がハイになり、周期的なアクセスが実行される間もハイのままです。

MASK[31:0]（QUADSPI_PSMAR）の内容を使用して、自動ポーリングモードにおけるFlashメモリからのデータがマスクされます。MASK[n] = 0 の場合、リザルトのビット n はマスクされ、考慮されません。MASK[n] = 1 かつ　bit[n] の値が MATCH[n]（QUADSPI_PSMAR）の値と同じである場合、bit n との一致が見られます。

ポーリング一致モードビット（PMM、QUADSPI_CR のビット 23）が 0 の場合、「AND」一致モードがアクティブになります。つまり、ステータス一致フラグ（SMF）は、マスクされていないすべてのビットで一致が検出された場合にのみセットされます。

PMM = 1 の場合、「OR」一致モードがアクティブになります。つまり、SMF は、マスクされていない任意のビットで一致が検出された場合にセットされます。

SMIE = 1 の場合に SMF がセットされると、割込みが呼び出されます。

自動ポーリングモード停止（APMS）ビットがセットされている場合、一致が検出されるとすぐに動作は停止し、BUSY フラグが 0 になります。あるいは、BUSY は 1 のままで、アボートが検出される、または QUADSPI が無効化される（EN = 0）まで周期的なアクセスが継続します。

データレジスタ（QUADSPI_DR）には、最後に受信したステータスバイト（FIFO の無効化）が含まれます。データレジスタの内容は、一致ロジックで使用されるマスキングの影響を受けません。FTF ステータスビットは新たなステータスの読出しが完了するとすぐにセットされ、データが読み出された直後にクリアされます。

20.3.7 QUADSPI メモリマップドモード

メモリマップドモードに設定された場合、外部 SPI デバイスは内部メモリとみなされます。

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

Flashメモリの容量がもっと大きい場合でも、アドレス指定できるのは 256 MB までです。

FSIZE で規定されている範囲外のアドレスにアクセスすると、256 MB の範囲内であっても、バスエラーが発生します。このエラーの影響は、アクセスを試みたバスマスタに依存します。

- Cortex® CPU の場合は、有効時バスフォールト例外（またはバスフォールト無効時にハードフォールト例外）が発生します。
- DMA の場合は、DMA 転送エラーが発生し、対応する DMA チャネルが自動的に無効になります。

バイト、ハーフワード、およびワードアクセスはすべてサポートされます。

XIP（Execute In Place）動作のサポートが実装されています。この場合、QUADSPI は次のアクセスを予測し、あらかじめ次のアドレスのバイトをロードします。その後のアクセスが実際に連続したアドレスに対して行われる場合、値がすでにプリフェッチされているので、アクセスは高速で完了します。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、Flash メモリへのアクセスが長時間なくても、前の読出し動作はアクティブな状態を保ち、nCS もローに保持されます。nCS がローのときにFlashメモリはより多く電力を消費する傾向があるため、FIFO がプリフェッチデータでフルになっている場合は、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過したあとで nCS が開放されるように、アプリケーションはタイムアウトカウンタ（TCEN=1、QUADSPI_CR のビット 3）の起動を要求する場合があります。

最初のメモリマップドアクセスが発生するとすぐに、BUSY はハイになります。プリフェッチ動作の影響で、BUSY の立ち下がりは、タイムアウトが起こる、アボートが検出される、またはペリフェラルが無効化されるまで起こりません。

20.3.8 QUADSPI Flashメモリの設定

デバイス設定レジスタ（QUADSPI_DCR）を使用して、外部 SPI Flashメモリの特性を指定することができます。

FSIZE[4:0] フィールドは、次の式を使用して外部メモリのサイズを定義します。

Flash メモリ内のバイト数 = $2^{[FSIZE+1]}$

FSIZE+1 はFlashメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。Flashメモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE は 2 つのFlashメモリの総容量を示します。

QUADSPI が 2 つのコマンドを実行する場合、一方のコマンドはもう一方のコマンドの直後に実行され、デフォルトでは1 CLK サイクル中の 2 つのコマンドの間にハイレベルのチップセレクト信号（nCS）を立ち上げます。Flashメモリがコマンド間により長い時間を必要とする場合、チップセレクトハイ時間（CSHT）フィールドを使用して、nCS がハイに保たれなければならない CLK サイクルの回数（最大 8）を指定することができます。

クロックモード（CKMODE）ビットは、コマンド間の CLK 信号ロジックレベルを示します（nCS = 1 の場合）。

20.3.9 QUADSPI 遅延データサンプリング

デフォルトでは、QUADSPI がFlashメモリに駆動されているデータのサンプリングを行うタイミングは、Flashメモリが信号を駆動してから 1/2 CLK サイクル後です。

外部信号の遅延が発生した場合、それは後でデータをサンプリングする際に有用となるかもしれません。SSHIFT ビット（QUADSPI_CR のビット 4）を使用することで、データのサンプリングを 1/2 CLK サイクルだけシフトすることができます。

クロックのシフトは DDR モードではサポートされていませんので、DDRM ビットがセットされた時点で SSHIFT ビットをクリアする必要があります。

20.3.10 QUADSPI の設定

QUADSPI の設定は次の 2 つのフェーズで行われます。

- QUADSPI IP の設定
- QUADSPI Flashメモリの設定

QUADSPI は、設定および有効化が行われると、インダイレクトモード、ステータスポーリングモード、メモリマップドモードの 3 つの動作モードのいずれかで使用することができます。

QUADSPI の設定

QUADSPI IP は QUADSPI_CR を使用して設定します。ユーザは、受信データに対してクロック･プリスケーラ分周比とサンプリング･シフトを設定します。

DDR モードは DDRM ビットによって設定することができます。QUADSPI インタフェースを DDR モードに設定する場合、カーネル･クロックの内部分周器には、2 以上の分周比を設定する必要があります。このモードを有効にすると、アドレスおよびオルタネート･バイトが両方のクロック･エッジで送信され、両方のクロック･エッジでデータの送受信が行われます。DDRM ビットの設定にかかわらず、命令は常に SDR モードで送信されます。

DMA リクエストは DMAEN ビットを設定することで有効化されます。割込みを使用する場合、それぞれのイネーブルビットもこのフェーズ中にセットされます。

DMA リクエスト生成または割込み生成の FIFO レベルは FTHRES ビットで設定されます。

タイムアウトの発生が必要な場合、TCEN ビットをセットして、QUADSPI_LPTR レジスタにタイムアウト値を設定することができます。

デュアルFlashモードは DFM を “1”にセットすることで有効にできます。

QUADSPI Flashメモリの設定

対象の外部Flashメモリに関連するパラメータは QUADSPI_DCR レジスタによって設定されます。ユーザは FSIZE ビットでFlashメモリのサイズを、CSHT ビットでチップセレクトハイ時間の最小値を、MODE ビットで機能モード（モード 0 またはモード 3）を設定します。

20.3.11 QUADSPI の使用

動作モードは FMODE[1:0]（QUADSPI_CCR[27:26])を使用して選択されます。

インダイレクトモードの手順

FMODE が 00 に設定されている場合、インダイレクト書込みモードが選択され、データをFlashメモリに送信することができます。FMODE が 01 に設定されている場合、インダイレクト読出しモードが選択され、データをFlashメモリから読み出すことができます。

QUADSPI がインダイレクトモードで使用される場合、フレームは次のように構成されます。

1. QUADSPI_DLR に、読み出される／書き込まれるデータのバイト数を指定します。
2. QUADSPI_CCR に、フレームフォーマット、モード、および命令コードを指定します。
3. QUADSPI_ABR に、アドレスフェーズの直後に送信されるオプションのオルタネートバイトを指定します。
4. QUADSPI_CR に、動作モードを指定します。FMODE = 00（インダイレクト書込みモード）かつ DMAEN = 1 の場合､QUADSPI_AR は QUADSPI_CR より前に指定されなければなりません。そうしないと、QUADSPI_AR が更新される前に QUADSPI_DR が DMA によって書き込まれる可能性があるからです（DMA コントローラがすでに有効な場合）。
5. QUADSPI_AR に、ターゲットアドレスを指定します。
6. QUADSPI_DR により FIFO に対してデータの読出し／書込みを行います。

制御レジスタ（QUADSPI_CR）に書き込む場合は次の設定値を指定してください。

- イネーブルビット（EN）を“1”にセット
- RAM へのデータ転送用の DMA 有効ビット（DMAEN）
- タイムアウトカウンタイネーブルビット（TCEN）
- サンプリングシフト設定（SSHIFT）
- FTF フラグをセットするタイミングを示す FIFO 閾値レベル（FTRHES）
- 割込みイネーブル
- 自動ポーリングモードのパラメータ : 一致モードおよび STOP モード（FMODE=11 のとき有効）
- クロックプリスケーラ

通信設定レジスタ（QUADSPI_CCR）に書き込む場合は次のパラメータを指定してください。

- INSTRUCTION ビットによる命令バイト
- IMODE ビット（1、2、または4 ライン）により命令を送信する方法
- ADMODE ビット（なし、または1、2、4 ライン）によりアドレスを送信する方法
- ADSIZE ビットによるアドレスサイズ（8、16、24、または32 ビット）
- ABMODE ビット（なし、または1、2、4 ライン）によりオルタネートバイトを送信する方法
- ABSIZE ビットによるオルタネートバイト数（1、2、3、または4）
- DBMODE ビットによるダミーバイトの有無
- DCYC ビットによるダミーバイト数
- DMODE ビットによるデータの送受信方法（なし、または1、2、4 ライン）

アドレスレジスタ（QUADSPI_AR）とデータレジスタ（QUADSPI_DR）のどちらも特定のコマンドに関して更新する必要がない場合、QUADSPI_CCR が書き込まれるとすぐにコマンドシーケンスが開始されます。これは、ADMODE と DMODE がどちらも 00 である場合、またはインダイレクト読出しモード（FMODE=01）で ADMODEが 00 の場合に該当します。

アドレスを必要とする場合（ADMODE が 00 以外）で、データレジスタに書き込む必要がない場合（FMODE = 01 または DMODE = 00）、QUADSPI_AR が書き込まれてアドレスが更新されるとすぐにコマンドシーケンスが開始されます。

データ転送時（FMODE = 00 かつ DMODE! = 00）、QUADSPI_DR による FIFO への書込みによって通信の開始がトリガされます。

ステータスフラグポーリングモード

ステータスフラグポーリングモードは、FMODE フィールド（QUADSPI_CCR[27:26]）を 10 にセットすることで有効になります。このモードでは、プログラムされたフレームが送信され、定期的にデータが取得されます。

各フレームに読み込まれるデータの最大量は 4 バイトです。QUADSPI_DLR で、これより多くのデータを要求しても、そのリクエストは無視され、4 バイトだけが読み出されます。

QUADSPI_PISR レジスタには周期性が指定されます。

ステータス·データが取得されると、そのデータを以下の目的のために内部処理できます。

- ステータス一致フラグをセットし、割込みが有効になっていれば生成する。
- ステータスバイトの周期的な取得を自動的に停止する。

受信した値は QUADSPI_PSMKR に格納されている値でマスクし、QUADSPI_PSMAR に格納されている値で論理和または論理積をとることができます。

一致する場合、ステータス一致フラグがセットされ、有効になっていれば割込みが生成されます。また、AMPS ビットがセットされている場合、QUADSPI は自動的に停止する可能性があります。

いずれにしろ、一番最後に取得した値は QUADSPI_DR で使用可能です。

メモリマップドモード

メモリマップドモードでは、外部Flashメモリは内部メモリとみなされますが、アクセス中にいくらかの遅延があります。このモードでは外部Flashメモリに対して読出し動作のみが許可されます。

メモリマップドモードに移行するには、QUADSPI_CCR レジスタの FMODE を 11 に設定します。

プログラムされた命令およびフレームは、マスタがメモリマップド空間にアクセスしているときに送信されます。

FIFO は、リニア読出しを予測するプリフェッチバッファとして使用されます。このモードでの QUADSPI_DR へのアクセスではすべてゼロが返されます。

メモリマップドモードでは、データ長レジスタ（QUADSPI_DLR）は意味を持ちません。

20.3.12 1 回限りの命令の送信

一部のFlashメモリ（Winbound など）が提供するモードは、最初のコマンドシーケンスでのみ命令を送信し、その後のコマンドは直接アドレスから開始されるというものです。SIOO ビット（QUADSPI_CCR[28]）を使用して、そのような機能を活用することができます。

SIOO はすべての機能モードで有効です（インダイレクト、自動ポーリング、メモリマップド）。SIOO ビットがセットされている場合、QUADSPI_CCR への書込み後の最初のコマンドに対してのみ命令が送信されます。その後のコマンドシーケンスは、命令フェーズでは、QUADSPI_CCR への書込みがあるまでスキップされます。

IMODE = 00（命令なし）の場合、SIOO は効果がありません。

20.3.13 QUADSPI エラー管理

次の場合に、エラーが生成される可能性があります。

- インダイレクトモードまたはステータス･フラグ･ポーリング･モードでは、QUADSPI_AR に誤ったアドレスがプログラムされている場合（QUADSPI_DCR の FSIZE[4:0] で定義されている Flash メモリのサイズによる）、TEF がセットされ、割込みが有効になっていれば生成されます。
- また、インダイレクトモードでは、アドレスとデータの合計長さが Flash メモリのサイズを超えている場合、アクセスがトリガされるとすぐに TEF がセットされます。
- メモリマップドモードでは、マスタによる範囲外のアクセスが行われた場合または QUADSPI が無効な場合、誤ったバス･マスタ･リクエストへの応答としてバスエラーが生成されます。
- メモリマップドモードが無効なときにマスタがメモリマップド空間にアクセスすると、誤ったバス･マスタ･リクエストへの応答としてバスエラーが生成されます。

20.3.14 QUADSPI の BUSY ビットおよびアボート機能

QUADSPI がFlashメモリでの動作を開始すると、BUSY ビットが QUADSPI_SR に自動的にセットされます。

インダイレクトモードでは、QUADSPI が要求されたコマンドシーケンスを完了し、FIFO が空になると、BUSY ビットがリセットされます。

自動ポーリングモードでは、一致（APMS = 1 のとき）またはアボートの検出により、最後の周期的なアクセスが完了して初めて BUSY ビットはローになります。

メモリマップドモードでの最初のアクセス後、タイムアウトイベントまたはアボートの検出により、BUSY ビットはローになります。

QUADSPI_CR レジスタの ABORT ビットをセットすることによって、動作を中止することができます。アボートが完了すると、BUSY ビットおよび ABORT ビットは自動的にリセットされ、FIFO が一掃されます。

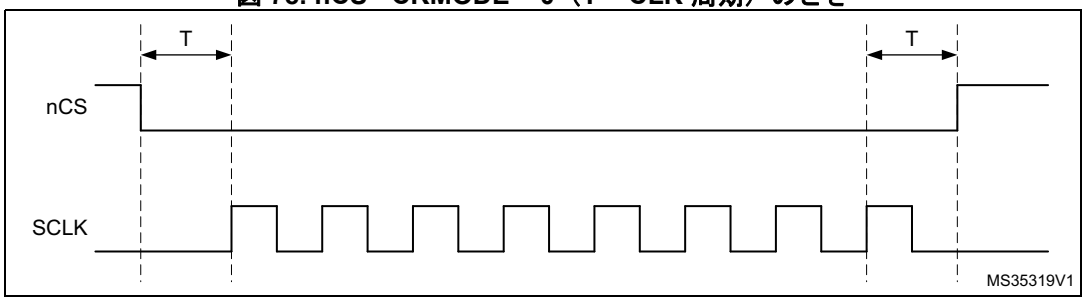
注： **Flashメモリには、ステータスレジスタへの書込み動作が中止された場合に誤動作するものもあります。**

20.3.15 nCS の動作

デフォルトでは nCS はハイで、外部Flashメモリは選択解除されています。nCS の立ち下がりは動作が始まる前に、立ち上がりは動作終了の直後に起こります。

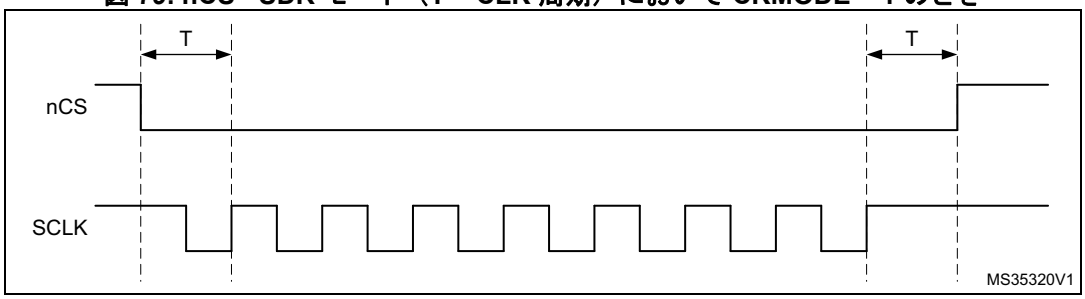
CKMODE = 0（「モード0」で、この状態では進行中の動作がない場合は CLK がローに保持されいてる）のとき、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（図 78 を参照）。

図 78. nCS - CKMODE = 0（T = CLK 周期）のとき



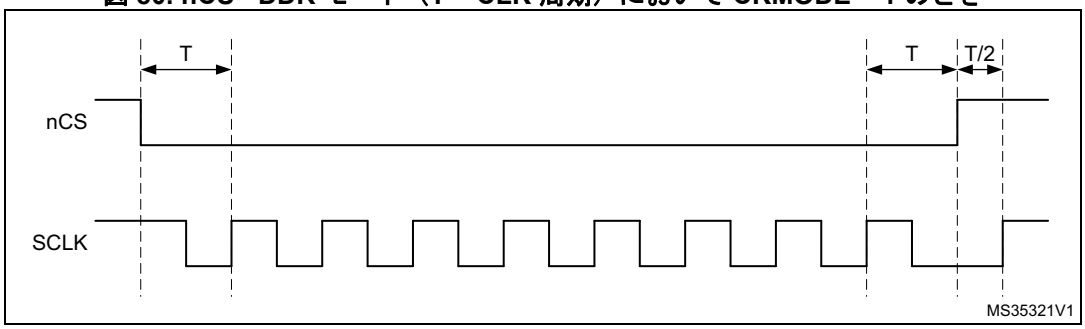
CKMODE = 1（「モード3」で、この状態では進行中の動作がない場合に CLK がハイになる）、かつ DDRM = 0（SDR モード）のとき、nCS は動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前でさらに立ち下がり、動作の最後の CLK エッジ立ち上がりから 1 CLK サイクル後に立ち上がります（図 79 を参照）。

図 79. nCS - SDR モード（T = CLK 周期）において CKMODE = 1 のとき



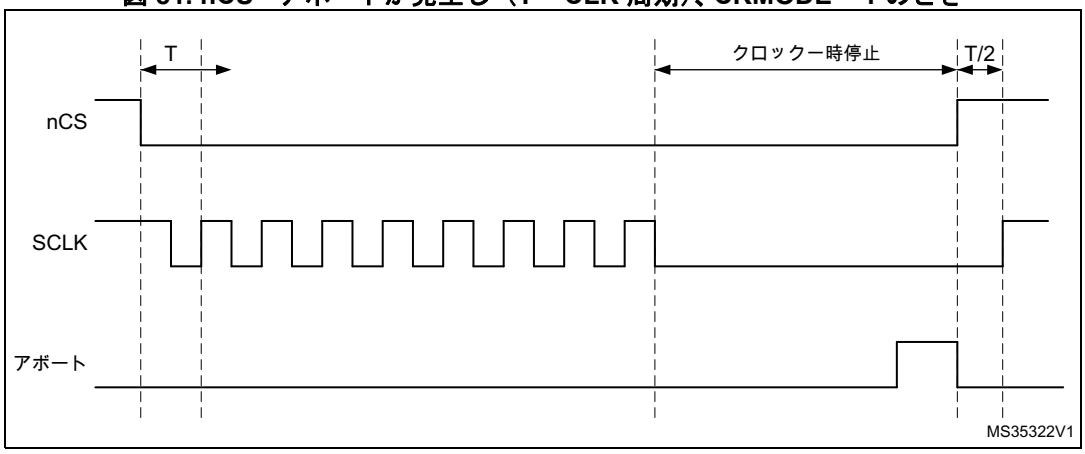
CKMODE = 1（モード3）かつ DDRM = 1（DDR モード）の場合、nCS の立ち下がりは動作の最初の CLK エッジ立ち上がりの 1 CLK サイクル前に、nCS の立ち上がりは動作の最後の アクティブ CLK エッジ立ち上がりから 1 CLK サイクル後に起こります（図 80 を参照）。DDR 動作は立ち下がりエッジで終わらなければならないため、nCS が立ち上がるとき CLK はローとなり、その後 CLK は 1/2 CLK サイクルだけ後ろ向きに立ちあがります。

図 80. nCS - DDR モード（T = CLK 周期）において CKMODE = 1 のとき



読出し動作で FIFO がフルのまま、または書込み動作で FIFO が空のままであれば、ファームウェアが FIFO を処理するまでその動作は停止し、CLK はローのままとなります。動作が停止した時点でアボートが発生した場合、アボートが要求された直後に nCS の立ち上がりが起こり、CLK は 1/2 CLK サイクル後に立ち上がります（図 81 を参照）。

図 81. nCS - アボートが発生し（T = CLK 周期）、CKMODE = 1 のとき



デュアル Flash 以外のモード（DFM = 0）で FSEL = 0 （デフォルト値）とすると、FLASH 1 がアクセスされるため BK2_nCS はハイに保たれ、FSEL = 1 とすると FLASH 2 がアクセスされるため、BK1_nCS がハイに保たれます。デュアル Flash モードでは、BK2_nCS は BK1_nCS とまったく同じように動作します。したがって、FLASH 2 が存在し、アプリケーションがデュアル Flash モード専用であれば、FLASH 2 にも BK1_nCS 信号を使用し、BK2_nCS 用のピンに他の機能を割り当てることができます。

20.4 QUADSPI 割込み

割込みは次のイベントによって生成できます。

- タイムアウト
- ステータス一致
- FIFO 閾値
- 転送完了
- 転送エラー

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 156. QUADSPI 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
タイムアウト	TOF	TOIE
ステータス一致	SMF	SMIE
FIFO 閾値	FTF	FTIE
転送完了	TCF	TCIE
転送エラー	TEF	TEIE

20.5 QUADSPI レジスタ

20.5.1 QUADSPI 制御レジスタ（QUADSPI_CR）

アドレスオフセット：0x0000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESCALER[7:0]								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	FTHRES[3:0]				FSEL	DFM	Res.	SSHIFT	TCEN	DMAEN	ABORT	EN
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:24 **PRESCALER[7:0]**：クロックプリスケーラ

このフィールドは、クロック（値 + 1）に基づいて CLK を生成するスケーラ係数を定義します。

0：F_{CLK} = F、クロックをそのまま QUADSPI CLK として使用します（プリスケーラを迂回）。

1：F_{CLK} = F/2

2：F_{CLK} = F/3

.....

255：F_{CLK} = F/256

クロック分周比が奇数の場合、CLK のデューティサイクルは 50% ではありません。クロック信号は、ハイに保持された時間より 1 サイクル分長く、ローに保持されます。

QUADSPI インタフェースを DDR モードに設定する場合、プリスケーラには、2 以上の分周比を設定する必要があります。

注： このフィールドは、BUSY が 0 のときにのみ変更できます。

ビット 23 **PMM**：ポーリング一致モード

このビットは、自動ポーリングモード時に「一致」を決定するために使うべき方法を示します。

0：AND 一致モードSMF は、Flashメモリから受信したマスクされていないすべてのビットが一致レジスタの対応するビットと一致している場合に、セットされます。

1：OR 一致モードSMF は、Flashメモリから受信したマスクされていないビットのいずれかが一致レジスタの対応するビットと一致している場合に、セットされます。

注： このビットは、BUSY が 0 のときにのみ修正できます。

ビット 22 **APMS**：自動ポーリング･モード停止

このビットは、一致後に自動ポーリングを停止するかどうかを決定します。

0：自動ポーリング･モードは、アボートまたは QUADSPI の無効化によってのみ停止されます。

1：自動ポーリング･モードは、一致が検出されるとすぐに停止されます。

注： このビットは、BUSY が 0 のときにのみ修正できます。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TOIE**：タイムアウト割込み有効化

このビットは、タイムアウト割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 19 **SMIE**：ステータス一致割込み有効化

このビットは、ステータス一致割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 18 **FTIE**：FIFO 閾値割込み有効化

このビットは、FIFO 閾値割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 17 **TCIE**：転送完了割込み有効

このビットは、転送完了割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 16 **TEIE**：転送エラー割込み有効

このビットは、転送エラー割込みを有効にします。

0：割込みは無効です。

1：割込みは有効です。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **FTHRES[3:0]**：FIFO 閾値レベル

このフィールドでは、FIFO 閾値フラグ（QUADSPI_SR レジスタの FTF ビット）がセットされる基準となる閾値（バイト数）をインダイレクト･モードで定義します。

0：インダイレクト書込みモード（FMODE = 00）で FIFO に 1 バイト以上の空きがある場合、またはインダイレクト読出しモード（FMODE = 01）で FIFO から読み出せる有効なデータが 1 バイト以上ある場合に FTF をセットします。

1：インダイレクト書込みモード（FMODE = 00）で FIFO に 2 バイト以上の空きがある場合、またはインダイレクト読出しモード（FMODE = 01）で FIFO から読み出せる有効なデータが 2 バイト以上ある場合に FTF をセットします。

.....

15：インダイレクト書込みモード（FMODE = 00）で FIFO に 16 バイトの空きがある場合、またはインダイレクト読出しモード（FMODE = 01）で FIFO から読み出せる有効なデータが 16 バイトある場合に FTF をセットします。

DMAEN = 1 の場合、FTHRES 値を変更する前に、対応するチャネル用の DMA コントローラを無効にする必要があります。

ビット 7 **FSEL**：Flash メモリの選択

このビットは、シングルFlashモード（DFM = 0）でアドレス指定するFlashメモリを選択します。

0：FLASH 1 が選択されます。

1：FLASH 2 が選択されます。

注： このビットは、BUSY が 0 のときにのみ修正できます。

このビットは、DFM = 1 のときは無視されます。

ビット 6 **DFM**：デュアル Flash モード

このビットはデュアルFlashモードをアクティブにします。このモードでは、2 つの外部Flashメモリを同時に使用して、スループットおよび容量を倍にします。

0：デュアル Flash モードは無効です。

1：デュアル Flash モードは有効です。

注： このビットは、BUSY が 0 のときにのみ修正できます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **SSHIFT**：サンプリングシフト

デフォルトでは、QUADSPI はFlashメモリによるデータ駆動の 1/2 CLK サイクル後にデータをサンプリングします。このビットを使用することで、外部信号の遅延を考慮するために、遅れてデータをサンプリングできます。

0：シフトなし

1：1/2 サイクルのシフト

ファームウェアは、DDR モード（DDRM = 1）では SSHIFT = 0 であることを予測する必要があります。

注： このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 3 **TCEN**：タイムアウトカウンタ有効化

このビットは、メモリマップドモードが選択されている（FMODE = 11)場合にのみ有効です。ある一定の時間アクセスがない場合（この時間は TIMEOUT[15:0]（QUADSPI_LPTR）によって定義されます)、このビットをアクティブにすることでチップセレクト（nCS）がリリースされます（これにより消費電力が削減されます)。

タイムアウトカウンタを有効にします。

デフォルトでは、QUADSPI がプリフェッチ動作を停止することはなく、Flashメモリへのアクセスが長時間なくても、前の読出し動作はアクティブな状態を保ち、nCs もローに保持されます。nCS がローのときにFlashメモリはより多く電力を消費する傾向があるため、FIFO がプリフェッチデータでフルになっている場合は、アクセスのない状態で TIMEOUT[15:0]（QUADSPI_LPTR）のサイクル周期が経過したあとで nCS が開放されるように、アプリケーションはタイムアウトカウンタ（TCEN=1、QUADSPI_CR のビット 3）の起動を要求する場合があります。

0：タイムアウトカウンタは無効です。したがって、チップセレクト（nCS）はメモリマップドモードにおける 1 回のアクセス後は無限にアクティブなままです。

1：タイムアウトカウンタは有効です。したがって、チップセレクトは、Flash メモリのインアクティブ期間が TIMEOUT[15:0] サイクル経過した後、メモリマップドモードでリリースされます。

注： このビットは、BUSY が 0 のときにのみ修正できます。

ビット 2 **DMAEN**：DMA 有効

インダイレクトモードでは、DMA を使用して、QUADSPI_DR レジスタ経由でデータを入出力することができます。DMA 転送は、FTF（FIFO 閾値フラグ）がセットされた時点で開始されます。

0：インダイレクト･モードでは DMA は無効です。

1：インダイレクト･モードでは DMA は有効です。

ビット 1 **ABORT**：アボート･リクエスト

このビットは進行中のコマンド･シーケンスを中止します。アボートが完了すると自動的にリセットされます。このビットは現在の転送を中止させます。

ポーリング･モードまたはメモリマップド･モードでは、このビットは APM ビットまたは DM ビットもリセットします。

0：アボートは要求されていません。

1：アボートが要求されています。

ビット 0 **EN**：イネーブル

QUADSPI. を有効にします。

0：QUADSPI. は無効です。

1：QUADSPI. は有効です。

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

アドレスオフセット：0x0004

リセット値：0x0000 0000

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

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **FSIZE[4:0]**：Flash メモリサイズ

このフィールドは、次の式を使用して外部メモリのサイズを定義します。

Flashメモリ内のバイト数 = $2^{[FSIZE+1]}$

FSIZE+1 はFlashメモリのアドレスを指定するために必要とされる有効なアドレスビット数です。Flash メモリの容量はインダイレクトモードでは最大 4 GB（32 ビットを使用してアドレス指定）ですが、メモリマップドモードにおけるアドレス指定可能な空間は 256 MB に制限されます。

DFM = 1 の場合、FSIZE には 2 つの Flash メモリの合計容量を指定します。

注： このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **CSHT[2:0]**：チップ・セレクト･ハイ時間

CSHT+1 は、Flashメモリに対して発行されるコマンド間でチップセレクト（nCS）がハイに保たれなければならない CLK サイクルの最小数を定義します。

0：nCS は、Flashメモリコマンド間の少なくとも 1 サイクルの間はハイに保持されます。

1：nCS は、Flashメモリコマンド間の少なくとも 2 サイクルの間はハイに保持されます。

.....

7：nCS は、Flashメモリコマンド間の少なくとも 8 サイクルの間はハイに保持されます。

注： このフィールドは、BUSY が 0 のときにのみ修正できます。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CKMODE**：モード 0／モード 3

このビットは、コマンド間（nCS = 1 の期間）における CLK のレベルを示します。

0：nCS がハイである（チップセレクトがリリースされている）間、CLK をローに保持する必要があります。これは、モード 0　と呼ばれます。

1：nCS がハイである（チップセレクトがリリースされている）間、CLK をハイに保持する必要があります。これは、モード 3　と呼ばれます。

注： このフィールドは、BUSY が 0 のときにのみ修正できます。

20.5.3 QUADSPI ステータスレジスタ（QUADSPI_SR）

アドレスオフセット：0x0008

リセット値：0x0000 0000

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **FLEVEL[4:0]**：FIFO レベル

このフィールドは FIFO 内に保持されている有効バイト数を示します。FIFO が空の場合、FLEVEL = 0、フルになっている場合、16です。メモリマップドモードおよび自動ステータス・ポーリング・モードでは、FLEVEL はゼロです。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **BUSY**：ビジー

このビットは動作が進行中のときにセットされます。このビットは、Flashメモリの動作が終わり、FIFOが空になると自動的にクリアされます。

ビット 4 **TOF**：タイムアウトフラグ

このビットは、タイムアウトが発生するとセットされます。CTOF に 1 を書き込むことによってクリアされます。

ビット 3 **SMF**：ステータス一致フラグ

このビットは、マスクされていない受信データが一致レジスタ（QUADSPI_PSMAR）の対応するビットと一致している場合に、自動ポーリングモードでセットされます。CSMF に 1 を書き込むことによってクリアされます。

ビット 2 **FTF**：FIFO 閾値フラグ

インダイレクトモードでは、このビットは FIFO 閾値に達した時点で、または Flash メモリからの読出し完了後に FIFO 内にデータが残っている場合にセットされます。閾値条件が真でなくなるとすぐに自動的にクリアされます。

自動ポーリングモードでは、このビットはステータスレジスタが読み出されるたびにセットされ、データレジスタが読み出された時点でクリアされます。

ビット 1 **TCF**：転送完了フラグ

インダイレクトモードではプログラムした数のデータが転送された時点、またどのモードでも転送が中止された時点で、このビットがセットされます。CTCF に 1 を書き込むことによってクリアされます。

ビット 0 **TEF**：転送エラー･フラグ

このビットは、インダイレクトモードで無効なアドレスにアクセスされているときに、インダイレクトモードでセットされます。CTEF に 1 を書き込むことによってクリアされます。

20.5.4 QUADSPI フラグ･クリア･レジスタ（QUADSPI_FCR）

アドレスオフセット：0x000C

リセット値：0x0000 0000

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

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CTOF**：タイムアウトフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TOF フラグがクリアされます。

ビット 3 **CSMF**：ステータス一致フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの SMF フラグがクリアされます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CTCF**：転送完了フラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TCF フラグがクリアされます。

ビット 0 **CTEF**：転送エラーフラグのクリア
このビットに 1 を書き込むと、QUADSPI_SR レジスタの TEF フラグがクリアされます。

20.5.5 QUADSPI データ長レジスタ（QUADSPI_DLR）

アドレスオフセット：0x0010

リセット値：0x0000 0000

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

ビット 31:0 **DL[31:0]**：データ長

インダイレクトモードおよびステータスポーリングモードで取得されるデータ数（値 + 1）ステータスポーリングモードの場合、3 より大きい値（4 バイトを示す）を使用する必要があります。

インダイレクトモードでは、すべて '1' は不定長を意味します。この場合、FSIZE で定義したメモリの終端まで QUADSPI が継続します。

0x0000_0000：1 バイトが転送されます。

0x0000_0001：2 バイトが転送されます。

0x0000_0002：3 バイトが転送されます。

0x0000_0003：4 バイトが転送されます。

.....

0xFFFF_FFFD：4,294,967,294（4G-2）バイトが転送されます。

0xFFFF_FFFE：4,294,967,295（4G-1）バイトが転送されます。

0xFFFF_FFFF：不定長 -- すべてのバイトがFlashメモリの終了（FSIZE によって定義）まで転送されます。FSIZE = 0x1F の場合、読出しが無限に続けられます。

デュアル Flash モード（DFM = 1）では、DL[0] ビットに '0' を書き込もうとしても、その値は '1' に固定されます。これによってアクセスごとに、必ず偶数バイトが転送されます。

このフィールドはメモリマップドモード（FMODE = 10）では効果がありません。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.6 QUADSPI 通信設定レジスタ（QUADSPI_CCR）

アドレスオフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DDRM	DHHC	Res.	SIOO	FMODE[1:0]		DMODE[1:0]		Res.	DCYC[4:0]					ABSIZE[1:0]	
rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABMODE[1:0]		ADSIZE[1:0]		ADMODE[1:0]		IMODE[1:0]		INSTRUCTION[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DDRM**：ダブル･データ･レート･モード

このビットは、アドレス、オルタネートバイト、およびデータフェーズの DDR モードをセットします。

0：DDR モード無効

1：DDR モード有効

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 30 **DHHC**：DDR ホールド

DDR モードで、データの出力を 1/4 QUADSPI 出力クロック･サイクルだけ遅延させます。

0：アナログ遅延を使用してデータ出力を遅延させます。

1：データ出力を 1/4 QUADSPI 出力クロック･サイクルだけ遅延させます。

この機能は、DDR モードでのみアクティブです。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

DHHC = 1 とする場合は、PRESCALER > 0 が必須になります。

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **SIOO**：命令の送信を 1 回のみとするモード

このビットは、IMODE = 00 のときには効果がありません。詳細は、セクション 20.3.12： を参照してください。

0：トランザクションごとに命令を送信します。

1：最初のコマンドでのみ命令を送信します。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 27:26 **FMODE[1:0]**：機能モード

このフィールドは、動作の QUADSPI 機能モードを定義します。

00：インダイレクト書込みモード

01：インダイレクト読出しモード

10：自動ポーリング･モード

11：メモリマップドモード

すでに DMAEN = 1 としている場合、FMODE 値を変更する前に、該当チャネルの DMA コントローラを無効にする必要があります。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 25:24 **DMODE[1:0]**：データ･モード

このフィールドは、動作のデータフェーズのモードを次のように定義します。

00：データはありません。

01：データは 1 本のライン上にあります。

10：データは 2 本のライン上にあります。

11：データは 4 本のライン上にあります。

また、このフィールドは動作のダミーフェーズモードも決定します。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **DCYC[4:0]**：ダミーサイクル数

このフィールドはダミーフェーズの処理時間を定義します。SDR モードと DDR モードの両方で、CLK サイクルの数を指定します（0 ~ 31）。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 17:16 **ABSIZE[1:0]**：オルタネートバイトサイズ

このビットはオルタネートバイトのサイズを定義します。

00：8 ビットのオルタネートバイト

01：16 ビットのオルタネートバイト

10：24 ビットのオルタネートバイト

11：32 ビットのオルタネートバイト

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 15:14 **ABMODE[1:0]**：オルタネート･バイト･モード

このフィールドは、動作のオルタネートバイトフェーズのモードを次のように定義します。

00：オルタネートバイトはありません。

01：オルタネートバイトは 1 本のライン上にあります。

10：オルタネートバイトは 2 本のライン上にあります。

11：オルタネートバイトは 4 本のライン上にあります。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 13:12 **ADSIZE[1:0]**：アドレスサイズ

このビットはアドレスサイズを定義します。

00：8 ビットアドレス

01：16 ビットアドレス

10：24 ビットアドレス

11：32 ビットアドレス

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 11:10 **ADMODE[1:0]**：アドレスモード

このフィールドは、動作のアドレスフェーズのモードを次のように定義します。

00：アドレスはありません。

01：アドレスは 1 本のライン上にあります。

10：アドレスは 2 本のライン上にあります。

11：アドレスは 4 本のライン上にあります。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 9:8 **IMODE[1:0]**：命令モード

このフィールドは、動作の命令フェーズのモードを次のように定義します。

00：命令はありません。

01：命令は 1 本のライン上にあります。

10：命令は 2 本のライン上にあります。

11：命令は 4 本のライン上にあります。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

ビット 7:0 **INSTRUCTION[7:0]**：命令

外部 SPI デバイスに送信される命令

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.7 QUADSPI アドレスレジスタ（QUADSPI_AR）

アドレスオフセット：0x0018

リセット値：0x0000 0000

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

ビット 31:0 **ADDRESS[31:0]**：アドレス

このフィールドには、外部 Flash メモリに送信するアドレスを格納します。

このフィールドへの書込みは、BUSY = 1 または FMODE = 11（メモリマップドモード）の場合は無視されます。

デュアル Flash モードでは、アドレスは常に偶数でなければならないので、ADDRESS[0] は自動的に 0 に固定されます。

20.5.8 QUADSPI オルタネート･バイト･レジスタ（QUADSPI_ABR）

アドレスオフセット：0x001C

リセット値：0x0000 0000

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

ビット 31:0 **ALTERNATE[31:0]**：オルタネートバイト

アドレスの直後に外部 SPI デバイスに送信されるオプションのデータ

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.9 QUADSPI データレジスタ（QUADSPI_DR）

アドレスオフセット：0x0020

リセット値：0x0000 0000

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

ビット 31:0 **DATA[31:0]**：データ

外部 SPI デバイスに対して送受信されるデータ

インダイレクト書込みモードでは、このレジスタに書き込まれたデータはいったん FIFO に格納されてから、データフェーズ時にFlashメモリに送信されます。FIFO がフルの場合、書き込まれるデータ量を受け入れるだけの十分な空間が FIFO にできるまで、書込み動作は停止されます。

インダイレクト読出しモードでは、このレジスタを読み出すことで（FIFO 経由）、Flashメモリから受信したデータが提供されます。FIFO のバイト数が読出し動作で要求したバイト数ほど多くない場合で、BUSY = 1 のとき、データ量が十分になるか、転送が完了するかのいずれか早く起こるほうまで、読出し動作は停止されます。

自動ポーリングモードでは、このレジスタは Flash メモリから読み出された最後のデータを格納します（マスキングなし）。

このレジスタへのワード、ハーフワード、バイトアクセスはサポートされます。インダイレクト書込みモードでの書込みにより FIFO に追加されるバイト数は、バイト書込みで 1 バイト、ハーフワード書込みで 2 バイト、ワード書込みで 4 バイトです。同様に、インダイレクト読出しモードでの読出しにより FIFO から削除されるバイト数は、バイト読出しで 1 バイト、ハーフワード読出しで 2 バイト、ワード読出しで 4 バイトです。インダイレクトモードでのアクセスは、このレジスタのボトムに整列されなければなりません。つまり、バイト読出しでは DATA[7:0]、ハーフワード読出しでは DATA[15:0] を読み出す必要があります。

20.5.10 QUADSPI ポーリング･ステータス･マスク･レジスタ（QUADSPI _PSMKR）

アドレスオフセット：0x0024

リセット値：0x0000 0000

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

ビット 31:0 **MASK[31:0]**：ステータスマスク

ポーリングモードで受信したステータスバイトに適用されるマスク

ビット n の場合

0：自動ポーリングモードで受信したデータのビット n はマスクされ、その値は一致ロジックでは考慮されません。

1：自動ポーリングモードで受信したデータのビット n はマスクされず、その値は一致ロジックで考慮されます。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

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

アドレスオフセット：0x0028

リセット値：0x0000 0000

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

ビット 31:0 **MATCH[31:0]**：ステータス一致

一致を得るためにマスクされたステータスレジスタと比較される値

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.12 QUADSPI ポーリング･インターバル･レジスタ（QUADSPI _PIR）

アドレスオフセット：0x002C

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **INTERVAL[15:0]**：ポーリングインターバル

自動ポーリングフェーズ時の読出し間の CLK サイクル数

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.13 QUADSPI 低消費電力タイムアウトレジスタ（QUADSPI_LPTR）

アドレスオフセット：0x0030

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TIMEOUT[15:0]**：タイムアウト期間

メモリマップドモードでのアクセス終了のたびに、QUADSPI はその後のバイトをプリフェッチし、それらを FIFO に保持します。このフィールドは、FIFO がフルになってから nCS を立ち上げるまでに QUADSPI が待つ CLK サイクルの数を示し、Flashメモリを消費電力がより少ない状態にします。

注： このフィールドは、BUSY が 0 のときにのみ書込みができます。

20.5.14 QUADSPI レジスタマップ

表 157. QUADSPI レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	**QUADSPI_CR**	PRESCALER[7:0]								PMM	APMS	Res.	TOIE	SMIE	FTIE	TCIE	TEIE	Res.	Res.	Res.	Res.	FTHRES [3:0]				FSEL	DFM	Res.	SSHIFT	TCEN	DMAEN	ABORT	EN
	リセット値	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0004	**QUADSPI_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSIZE[4:0]					Res.	Res.	Res.	Res.	Res.	CSHT			Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKMODE
	リセット値												0	0	0	0	0						0	0	0								0
0x0008	**QUADSPI_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLEVEL[4:0]					Res.	Res.	BUSY	TOF	SMF	FTF	TCF	TEF
	リセット値																				0	0	0	0	0			0	0	0	0	0	0
0x000C	**QUADSPI_FCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTOF	CSMF	Res.	CTCF	CTEF
	リセット値																												0	0		0	0
0x0010	**QUADSPI_DLR**	DL[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0014	**QUADSPI_CCR**	DDRM	DHHC	Res.	SIOO	FMODE[1:0]		DMODE[1:0]		Res.	DCYC[4:0]					ABSIZE[1:0]		ABMODE[1:0]		ADSIZE[1:0]		ADMODE[1:0]		IMODE[1:0]		INSTRUCTION[7:0]							
	リセット値	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	**QUADSPI_AR**	ADDRESS[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x001C	**QUADSPI_ABR**	ALTERNATE[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0020	**QUADSPI_DR**	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	**QUADSPI_ PSMKR**	MASK[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	**QUADSPI_ PSMAR**	MATCH[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	**QUADSPI_PIR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INTERVAL[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0030	**QUADSPI_ LPTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEOUT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2 : を参照してください。

21 A/D コンバータ（ADC）

21.1 概要

このセクションでは、最大 5 つ実装されている ADC について説明します。

- ADC1 および ADC2 は密接に結合されており、デュアルモードで動作できます（ADC1 はマスタです）。
- ADC3 および ADC4 は密接に結合されており、デュアルモードで動作できます（ADC3 はマスタです）。
- ADC5 は個別に制御されます。

各 ADC は、12 ビットの逐次比較型アナログデジタルコンバータで構成されています。

各 ADC には、最大 19 の多重化チャネルがあります。さまざまなチャネルの A/D 変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADC の結果は、左詰めまたは右詰めで 16 ビットのデータレジスタに格納されます。

ADC は AHB バスに配置され、データを高速で処理できます。

アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

組込みのハードウェア·オーバーサンプリング回路により、CPU の計算負荷を軽減しながら優れたアナログ性能が得られます。

低周波数できわめて少ない消費電力を実現するために、効率的な低電力モードが実装されています。

21.2 ADC の主な機能

- ハイパフォーマンス機能
  - 最大 5 つの ADC、うち 4 つ（ペア）はデュアルモードで動作可能：
    ADC1 は、14 個の外部チャネルと 4 つの内部チャネルに接続
    ADC2 は、16 個の外部チャネルと 2 つの内部チャネルに接続
    ADC3 は、15 個の外部チャネルと 3 つの内部チャネルに接続
    ADC4 は、16 個の外部チャネルと 2 つの内部チャネルに接続
    ADC5 は、13 個の外部チャネルと 5 つの内部チャネルに接続
  - 12、10、8、または 6 ビットに設定可能な分解能
  - ADC 変換時間は AHB バスクロック周波数から独立
  - 分解能を下げることで変換時間を短縮可
  - シングルエンドまたは差動入力を管理
  - 高速データ処理を可能にする AHB スレーブバスインタフェース
  - 自己較正
  - チャネル単位でプログラム可能なサンプリング時間
  - 柔軟なサンプリング時間制御
  - 最大 4 つのインジェクトチャネル（アナログ入力をレギュラまたはインジェクトチャネルへ完全に割り当て設定可能）
  - 高速コンテキスト切り替えを可能にするためにインジェクトチャネルの内容を準備するハードウェアによる支援
  - 組み込みのデータコヒーレンシによるデータ配置
  - データは、レギュラチャネル変換において、DMA で管理可能
  - 4 つのインジェクトチャネル専用のデータレジスタ
- オーバーサンプリング回路
  - 16 ビットデータレジスタ
  - 2 ~ 256 倍の範囲でオーバーサンプリング比を調整可能
  - プログラム可能な最大 8 ビットのデータ·シフト
- データプレコンディショニング
  - ゲイン補正
  - オフセット補正
- 低電力機能
  - 低周波数での動作時に ADC 消費電流を低減するための速度適応型低電力モード
  - 最適な ADC 性能を維持しながら、低いバス周波数を適用可能
  - 低 AHB バスクロック周波数アプリケーション（自動遅延モード）で ADC オーバーランを避けるための自動制御を提供
- A/D コンバータごとの外部アナログ入力チャネル数
  - GPIO パッドから最大 5 個の高速チャネル
  - GPIO パッドから最大 13 個の低速チャネル

- 複数の専用内部チャネルを別途用意
  - 内部基準電圧（V_{REFINT}）を ADC1、3、4、および 5 に接続
  - 内部温度センサ（V_{TS}）は ADC1 および 5 に接続
  - V_{BAT} 監視チャネル（$V_{BAT}/3$）を ADC1、3（カテゴリ 3 のデバイス専用）、および 5 に接続
  - OPAMP1 内部出力を ADC1 に接続
  - OPAMP2 および OPAMP3 内部出力は、ADC2 に接続
  - OPAMP3 内部出力を ADC3 に接続
  - OPAMP6 内部出力を ADC4（カテゴリ 3 のデバイスの場合）または ADC3（カテゴリ 4 のデバイスの場合）に接続
  - OPAMP4 および OPAMP5 内部出力を ADC5 に接続
- 変換開始は、次のように開始可能
  - レギュラ変換とインジェクト変換の両方について、ソフトウェアによって開始
  - レギュラ変換とインジェクト変換の両方について、極性が設定可能なハードウェアトリガによって開始（内部タイマイベントまたは GPIO 入力イベント）
- 変換モード
  - 各 ADC では単一チャネルの変換、または一連のチャネルをスキャン可能
  - シングルモードは、選択された入力をトリガごとに1回変換します。
  - 連続モードは、選択された入力を連続的に変換します。
  - 不連続モード
- ADC1、ADC2、ADC3、および ADC4 のデュアル ADC モード
- ADC レディ時、サンプリング終了時、（レギュラまたはインジェクト）変換終了時、（レギュラまたはインジェクト）シーケンス変換終了時、アナログウォッチドッグ 1、2、3、またはオーバーランイベント時に割込みを生成
- ADC ごとの 3 つのアナログウォッチドッグ
  - ウォッチドッグはフィルタリングを行って、範囲外のデータを無視できます。
- ADC 入力電圧範囲：$V_{REF-} \leq V_{IN} \leq V_{REF+}$

図 82 に、1 つの ADC のブロック図を示します。

オペアンプ出力電圧の変換で適用する A/D コンバータのサンプリング時間については、製品データシートでオペアンプの電気的特性のセクションを参照してください。

21.3 A/D コンバータの実装

表 158. A/D コンバータの機能

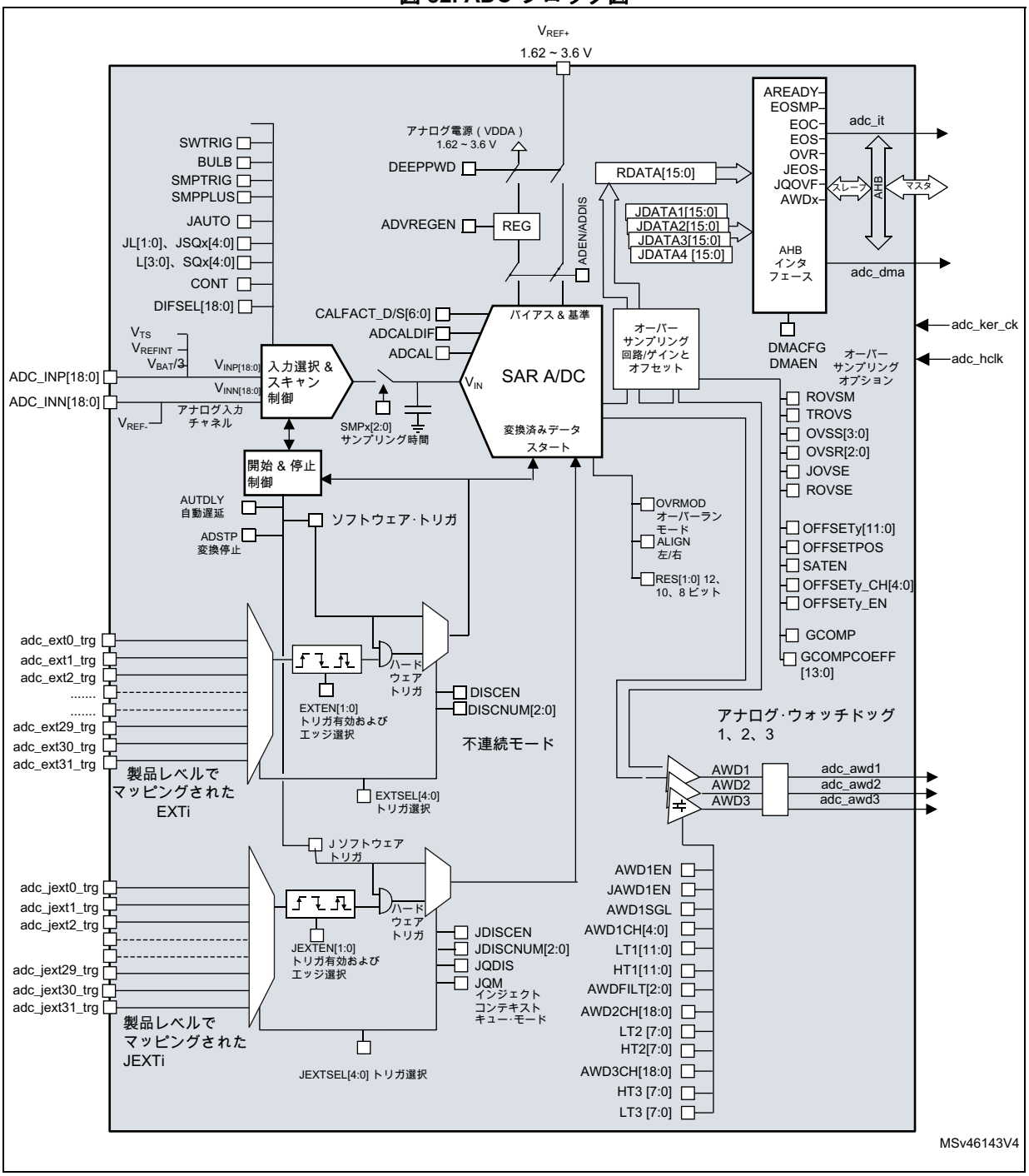
A/D コンバータのモード/機能	ADC1	ADC2	ADC3	ADC4	ADC5
デュアル·モード	x（カップル）		x（カップル）		-

21.4 ADC の機能詳細

21.4.1 ADC ブロック図

図 82 に A/D コンバータのブロック図、表 160 に A/D コンバータの各ピンの説明をそれぞれ示します。

図 82. ADC ブロック図



21.4.2 A/D コンバータのピンおよび内部信号

表 159. A/D コンバータの内部入力信号と内部出力信号

内部信号名	信号タイプ	説明
adc_ext_trg[31:0]	入力	レギュラ変換については、最大 32 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_jext_trg[31:0]	入力	インジェクト変換については、最大 31 個の外部トリガ入力（オンチップタイマに接続可能）。 これらの入力は、ADC マスタと ADC スレーブの間で共有されています。
adc_awdx_out	出力	オンチップ・タイマに接続された内部アナログ・ウォッチドッグの出力信号（x = アナログ・ウォッチドッグ番号 = 1、2、3）
adc_ker_ck	出力	A/D コンバータのカーネル・クロック
adc_hclk	入力	ADC ペリフェラルクロック
adc_it	出力	ADC 割込み
adc_dma	出力	ADC DMA リクエスト
V_{TS}	入力	内部温度センサからの出力電圧
V_{REFINT}	入力	内部基準電圧からの出力電圧
V_{BAT}	入力電源供給	外部バッテリ電圧供給

表 160. A/D コンバータの各入出力ピン

ピン名	信号タイプ	コメント
VREF+	入力、アナログ基準電圧正	A/D コンバータのハイレベル/正基準電圧
VDDA	入力、アナログ電源	V_{DDA} に等しいアナログ電源電圧
VREF−	入力、アナログ基準電圧負	ADC のローレベル／負基準電圧。V_{REF-} は内部的に V_{SSA} に接続されています。
VSSA	入力、アナログ電源グラウンド	アナログ電源のグランド。専用の V_{SSA} ピンがないデバイスパッケージでは、V_{SSA} は内部的に V_{SS} に接続されています。
$V_{INP}i$	各 A/D コンバータの 正側アナログ入力チャネル	外部チャネル ADCx_INP*i* または内部チャネルのいずれかに接続。この入力はシングルエンド・モードに変換されます。
$V_{INN}i$	各 A/D コンバータの 負側アナログ入力チャネル	V_{REF-} または外部チャネルのいずれかに接続。ADCx_INNi および ADCx_INP[i+1]
ADCx_INNi	負側外部アナログ入力信号	最大 19 本のアナログ入力チャネル（x = A/D コンバータの番号 = 1、2、3、4、5）： 詳細については、セクション 21.4.4：ADC1/2/3/4/5 のコネクティビティを参照してください。
ADCx_INPi	正側外部アナログ入力信号	最大 19 本のアナログ入力チャネル（x = A/D コンバータの番号 = 1、2、3、4、5）： 詳細については、セクション 21.4.4：ADC1/2/3/4/5 のコネクティビティを参照してください。

21.4.3 A/D コンバータのクロック

デュアル･クロック･ドメイン･アーキテクチャ

デュアルクロックドメインアーキテクチャは、ADC クロックが AHB バスクロックから独立していることを意味します。

入力クロックは、すべての A/D コンバータに共通で、2 種類のクロック･ソースから選択できます（図 83：A/D コンバータのクロック構成を参照）。

1. A/D コンバータのクロックには、以下のクロック･ソースから得られた固有のクロック･ソースを使用できます。
   - システム･クロック
   - PLL P クロック

   ADC 専用クロックの生成方法の詳細については、RCC のセクションを参照してください。この構成を選択するには、ADCx_CCR レジスタのビット CKMODE[1:0] をリセットする必要があります。
2. ADC クロックは、ADC バスインタフェースの AHB クロックから生成し、プログラム可能な係数（1、2、または 4）で分周することができます。このモードでは、プログラム可能な分周比を選択できます（ビット CKMODE[1:0] に従って /1、2、または 4）。

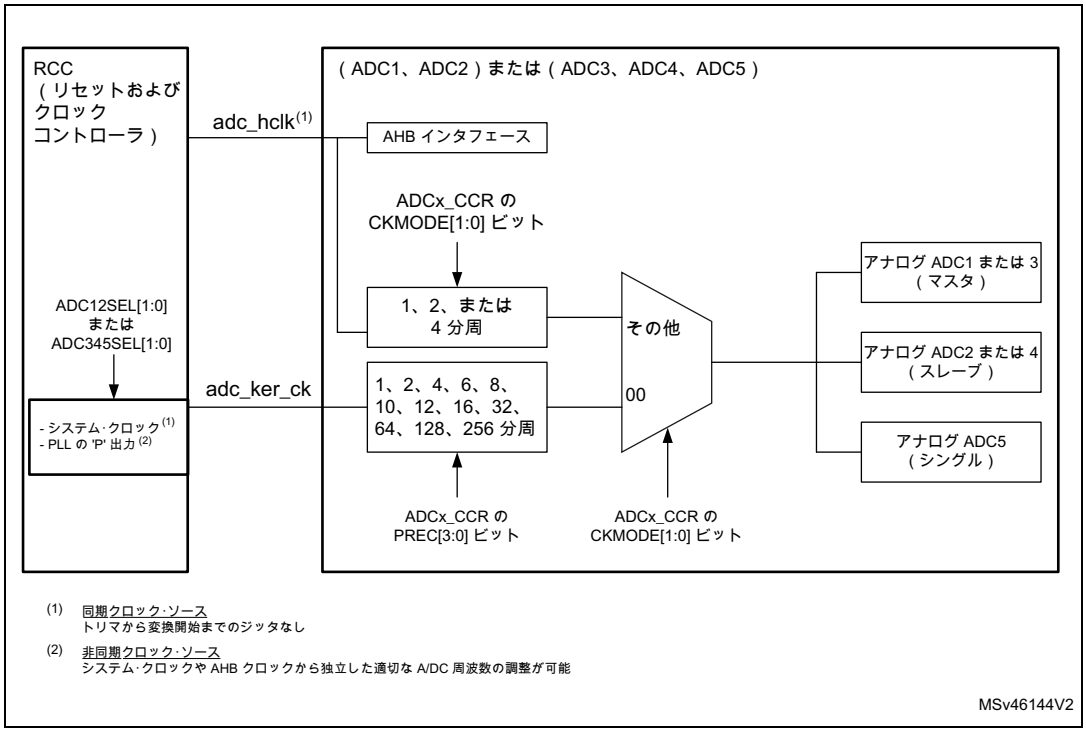
   この構成を選択するには、ADCx_CCR レジスタのビット CKMODE[1:0] が「00」以外の値である必要があります。

注： **オプション 2 では、AHB プリスケーラが設定されている（RCC_CFGR レジスタで HPRE[3:0] = 0xxx）場合のみ、プリスケーラ係数を 1（CKMODE[1:0] = 01）にできます。**

オプション 1 には、どの AHB クロック構成を選択していても、A/D コンバータの最大クロック周波数を使用できる利点があります。ADC クロックは、最終的には1、2、4、6、8、12、16、32、64、128、256 の比率で分周できます。このとき、ADCx_CCR レジスタの PRESC[3:0] ビットで設定されたプリスケーラを使用します。

オプション 2 には、クロックドメインの再同期を迂回するという利点があります。A/D コンバータをタイマでトリガする場合、不確実性がない正確なトリガを必要とするアプリケーションでは、このオプションが有利です（このオプションを使用しないと、2 つのクロック･ドメイン間の再同期によってトリガの発生タイミングが不確実になる可能性が高くなります）。

図 83. A/D コンバータのクロック構成



A/D コンバータのクロックと AHB のクロックとの比率に対する制約

通常、A/D コンバータのクロックと AHB のクロックの間に注意すべき制約はありませんが、インジェクト·チャネルをプログラムしている場合は例外です。この場合、次の比率に注意する必要があります。

- すべてのチャネルの分解能が 12 ビットまたは 10 ビットである場合、$F_{abc_hclk} >= F_{ADC}/4$ とする
- 分解能が 8 ビットのチャネルがあり、それを下回る分解能のチャネルがない場合、$F_{adc_hclk} >= F_{ADC}/3$ とする
- 分解能が 6 ビットのチャネルがある場合、$F_{adc_hclk} >= F_{ADC}/2$ とする

21.4.4 ADC1/2/3/4/5 のコネクティビティ

ADC1、ADC2、ADC3、ADC4、および ADC5 は密接に結合されており、以下の図に示すいくつかの外部チャネルを共有しています。

ADCy_INPx は、製品データシートで ADCy_INx と表記されているピンに相当します。

図 84. ADC1 接続性

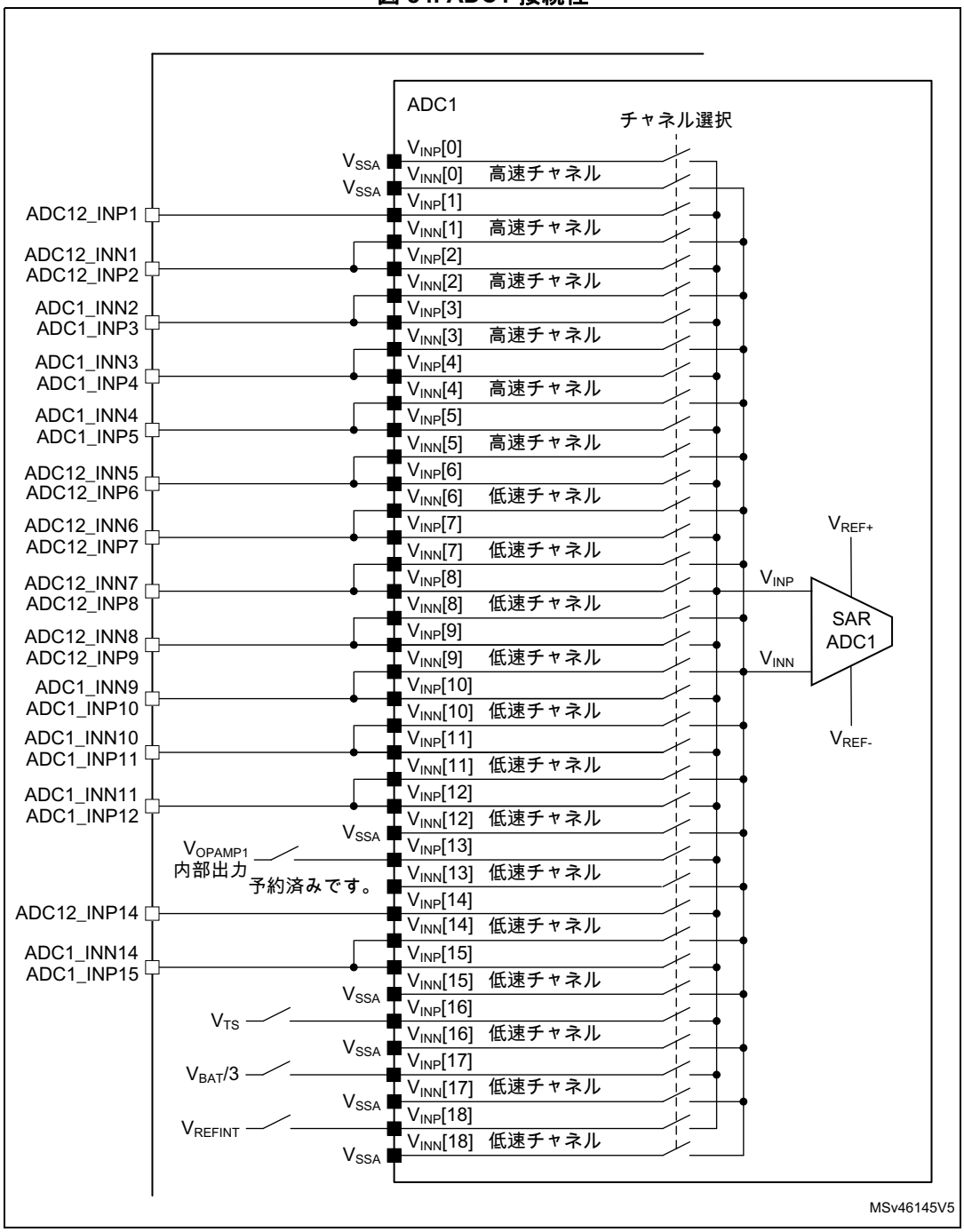


図 85. ADC2 接続性

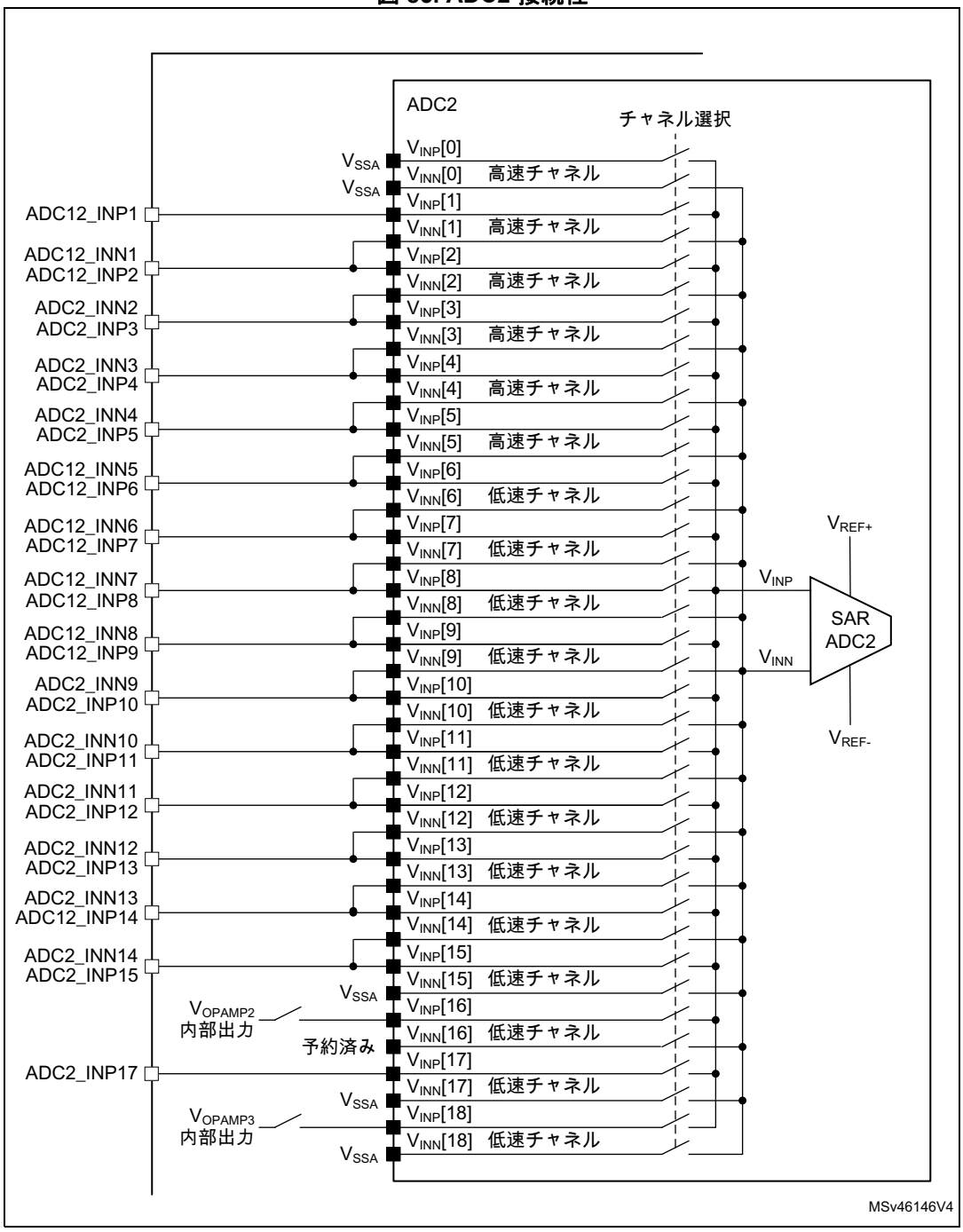

ADC2
チャネル選択
ADC12_INP1
ADC12_INN1
ADC12_INP2
ADC2_INN2
ADC2_INP3
ADC2_INN3
ADC2_INP4
ADC2_INN4
ADC2_INP5
ADC12_INN5
ADC12_INP6
ADC12_INN6
ADC12_INP7
ADC12_INN7
ADC12_INP8
ADC12_INN8
ADC12_INP9
ADC2_INN9
ADC2_INP10
ADC2_INN10
ADC2_INP11
ADC2_INN11
ADC2_INP12
ADC2_INN12
ADC2_INP13
ADC2_INN13
ADC12_INP14
ADC2_INN14
ADC2_INP15
ADC2_INP17
VSSA
VSSA
VINP[0]
VINN[0] 高速チャネル
VINP[1]
VINN[1] 高速チャネル
VINP[2]
VINN[2] 高速チャネル
VINP[3]
VINN[3] 高速チャネル
VINP[4]
VINN[4] 高速チャネル
VINP[5]
VINN[5] 高速チャネル
VINP[6]
VINN[6] 低速チャネル
VINP[7]
VINN[7] 低速チャネル
VINP[8]
VINN[8] 低速チャネル
VINP[9]
VINN[9] 低速チャネル
VINP[10]
VINN[10] 低速チャネル
VINP[11]
VINN[11] 低速チャネル
VINP[12]
VINN[12] 低速チャネル
VINP[13]
VINN[13] 低速チャネル
VINP[14]
VINN[14] 低速チャネル
VINP[15]
VINN[15] 低速チャネル
VSSA
VOPAMP2
内部出力
予約済み
VINP[16]
VINN[16] 低速チャネル
VINP[17]
VINN[17] 低速チャネル
VSSA
VOPAMP3
内部出力
VINP[18]
VINN[18] 低速チャネル
VSSA
VREF+
VINP
SAR
ADC2
VINN
VREF-
MSv46146V4

図 86. ADC3 接続性

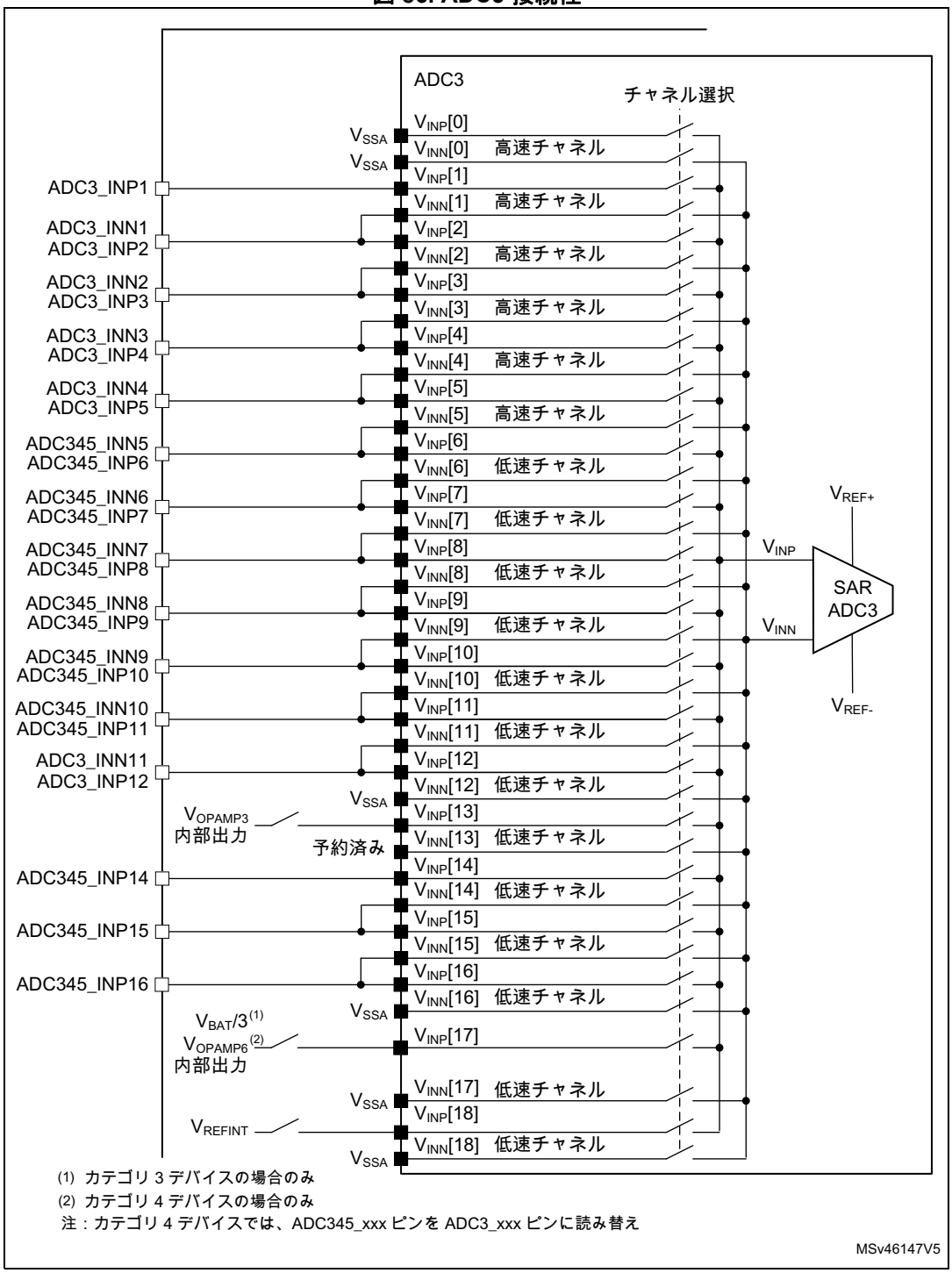

ADC3
チャネル選択
ADC3_INP1
ADC3_INN1
ADC3_INP2
ADC3_INN2
ADC3_INP3
ADC3_INN3
ADC3_INP4
ADC3_INN4
ADC3_INP5
ADC345_INN5
ADC345_INP6
ADC345_INN6
ADC345_INP7
ADC345_INN7
ADC345_INP8
ADC345_INN8
ADC345_INP9
ADC345_INN9
ADC345_INP10
ADC345_INN10
ADC345_INP11
ADC3_INN11
ADC3_INP12
ADC345_INP14
ADC345_INP15
ADC345_INP16
VSSA
VINP[0]
VINN[0] 高速チャネル
VINP[1]
VINN[1] 高速チャネル
VINP[2]
VINN[2] 高速チャネル
VINP[3]
VINN[3] 高速チャネル
VINP[4]
VINN[4] 高速チャネル
VINP[5]
VINN[5] 高速チャネル
VINP[6]
VINN[6] 低速チャネル
VINP[7]
VINN[7] 低速チャネル
VINP[8]
VINN[8] 低速チャネル
VINP[9]
VINN[9] 低速チャネル
VINP[10]
VINN[10] 低速チャネル
VINP[11]
VINN[11] 低速チャネル
VINP[12]
VINN[12] 低速チャネル
VOPAMP3 内部出力
予約済み
VINP[13]
VINN[13] 低速チャネル
VINP[14]
VINN[14] 低速チャネル
VINP[15]
VINN[15] 低速チャネル
VINP[16]
VINN[16] 低速チャネル
VBAT/3(1)
VOPAMP6(2) 内部出力
VINP[17]
VINN[17] 低速チャネル
VREFINT
VINP[18]
VINN[18] 低速チャネル
VREF+
VINP
VINN
SAR ADC3
VREF-
(1) カテゴリ 3 デバイスの場合のみ
(2) カテゴリ 4 デバイスの場合のみ
注：カテゴリ 4 デバイスでは、ADC345_xxx ピンを ADC3_xxx ピンに読み替え
MSv46147V5

図 87. ADC4 接続性

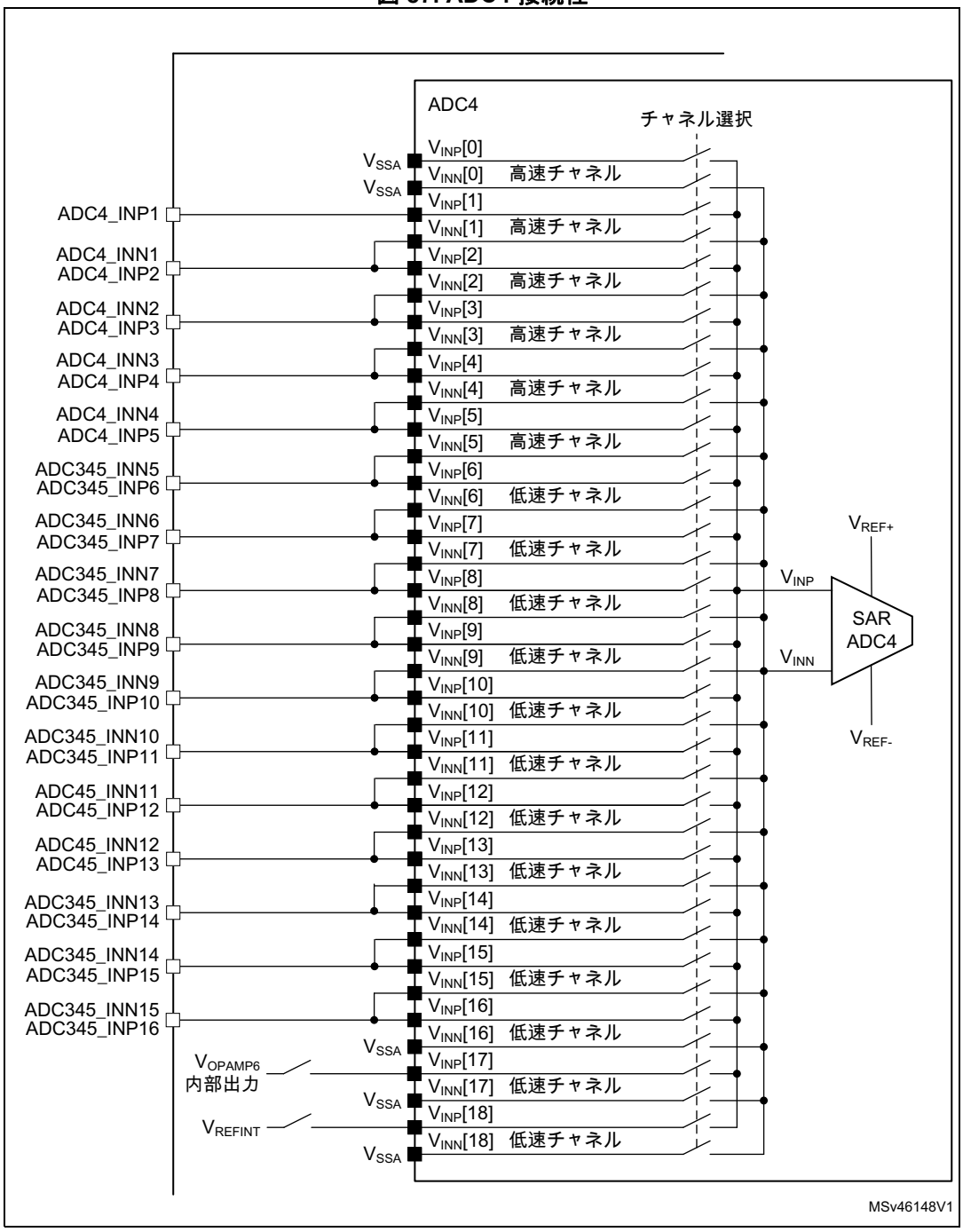
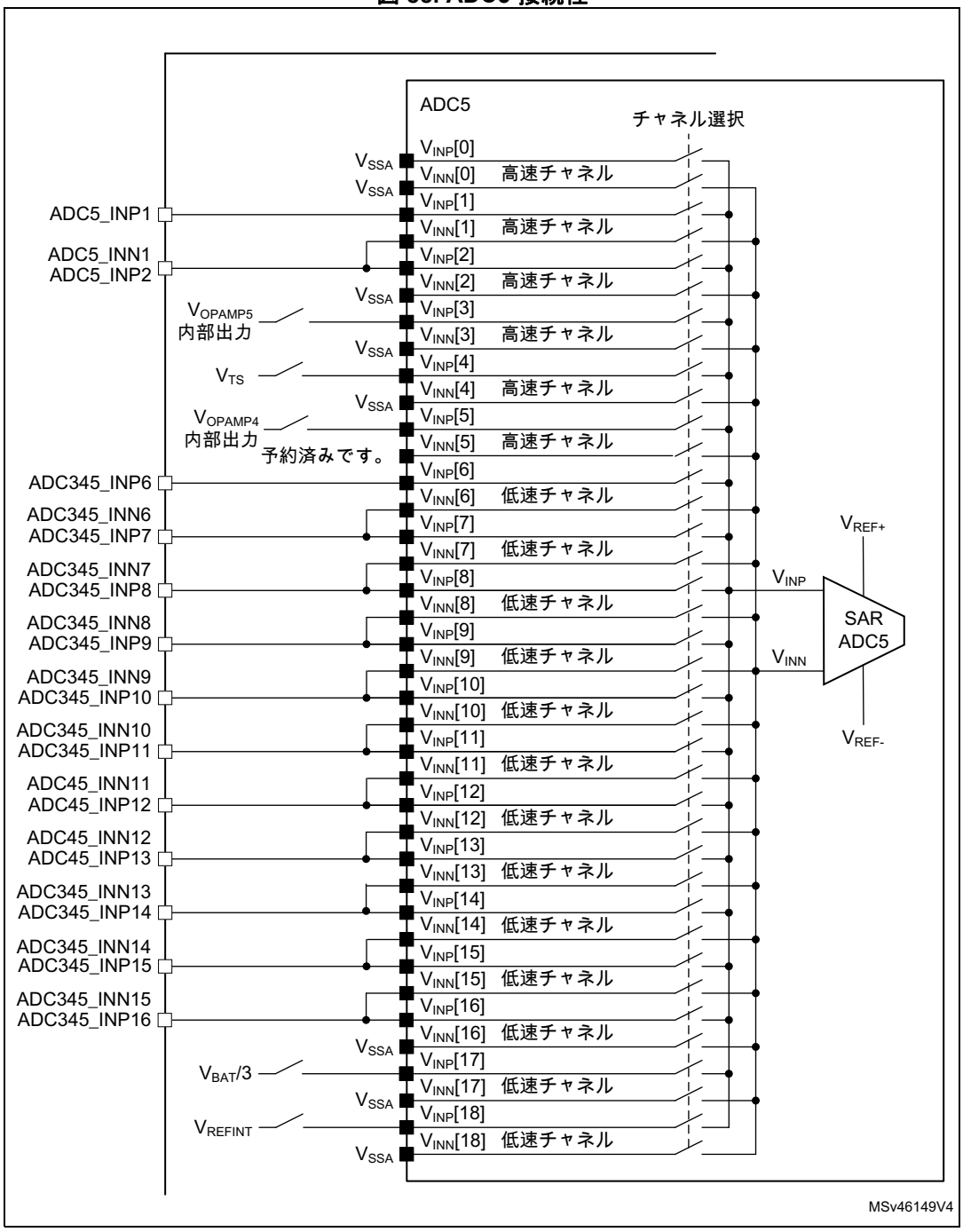

ADC4
チャネル選択
ADC4_INP1
ADC4_INN1
ADC4_INP2
ADC4_INN2
ADC4_INP3
ADC4_INN3
ADC4_INP4
ADC4_INN4
ADC4_INP5
ADC345_INN5
ADC345_INP6
ADC345_INN6
ADC345_INP7
ADC345_INN7
ADC345_INP8
ADC345_INN8
ADC345_INP9
ADC345_INN9
ADC345_INP10
ADC345_INN10
ADC345_INP11
ADC45_INN11
ADC45_INP12
ADC45_INN12
ADC45_INP13
ADC345_INN13
ADC345_INP14
ADC345_INN14
ADC345_INP15
ADC345_INN15
ADC345_INP16
VSSA
VSSA
VOPAMP6
内部出力
VREFINT
VSSA
VSSA
VINP[0]
VINN[0] 高速チャネル
VINP[1]
VINN[1] 高速チャネル
VINP[2]
VINN[2] 高速チャネル
VINP[3]
VINN[3] 高速チャネル
VINP[4]
VINN[4] 高速チャネル
VINP[5]
VINN[5] 高速チャネル
VINP[6]
VINN[6] 低速チャネル
VINP[7]
VINN[7] 低速チャネル
VINP[8]
VINN[8] 低速チャネル
VINP[9]
VINN[9] 低速チャネル
VINP[10]
VINN[10] 低速チャネル
VINP[11]
VINN[11] 低速チャネル
VINP[12]
VINN[12] 低速チャネル
VINP[13]
VINN[13] 低速チャネル
VINP[14]
VINN[14] 低速チャネル
VINP[15]
VINN[15] 低速チャネル
VINP[16]
VINN[16] 低速チャネル
VINP[17]
VINN[17] 低速チャネル
VINP[18]
VINN[18] 低速チャネル
VREF+
VINP
VINN
SAR
ADC4
VREF-
MSv46148V1

図 88. ADC5 接続性


ADC5
チャネル選択
ADC5_INP1
ADC5_INN1
ADC5_INP2
VOPAMP5
内部出力
VTS
VOPAMP4
内部出力
予約済みです。
ADC345_INP6
ADC345_INN6
ADC345_INP7
ADC345_INN7
ADC345_INP8
ADC345_INN8
ADC345_INP9
ADC345_INN9
ADC345_INP10
ADC345_INN10
ADC345_INP11
ADC45_INN11
ADC45_INP12
ADC45_INN12
ADC45_INP13
ADC345_INN13
ADC345_INP14
ADC345_INN14
ADC345_INP15
ADC345_INN15
ADC345_INP16
VBAT/3
VREFINT
VSSA
VINP[0]
VINN[0] 高速チャネル
VINP[1]
VINN[1] 高速チャネル
VINP[2]
VINN[2] 高速チャネル
VINP[3]
VINN[3] 高速チャネル
VINP[4]
VINN[4] 高速チャネル
VINP[5]
VINN[5] 高速チャネル
VINP[6]
VINN[6] 低速チャネル
VINP[7]
VINN[7] 低速チャネル
VINP[8]
VINN[8] 低速チャネル
VINP[9]
VINN[9] 低速チャネル
VINP[10]
VINN[10] 低速チャネル
VINP[11]
VINN[11] 低速チャネル
VINP[12]
VINN[12] 低速チャネル
VINP[13]
VINN[13] 低速チャネル
VINP[14]
VINN[14] 低速チャネル
VINP[15]
VINN[15] 低速チャネル
VINP[16]
VINN[16] 低速チャネル
VINP[17]
VINN[17] 低速チャネル
VINP[18]
VINN[18] 低速チャネル
VREF+
VINP
SAR
ADC5
VINN
VREF-
MSv46149V4

21.4.5 スレーブ AHB インタフェース

ADC は、制御／ステータスレジスタおよびデータアクセスのために AHB スレーブポートを実装しています。AHB インタフェースの機能を次に示します。

- ワード（32 ビット）アクセス
- シングルサイクルレスポンス
- レジスタへのすべての読出し／書込みアクセスに対してゼロウェイトステートの応答。

AHB スレーブインタフェースでは分割／再試行リクエストをサポートしておらず、また AHB エラーも発生しません。

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

デフォルトでは、リーク電流を低減するために電源が内部的にオフになっている（ADC_CR レジスタのビット DEEPPWD のリセット状態が 1 である）場合、ADC はディープパワーダウンモードになります。

A/D コンバータの動作を開始するには、以下のシーケンスを実行します。

1. DEEPPWD ビットをクリアしてディープ･パワーダウン･モードを終了します。
2. ADVREGEN ビットをセットして、A/D コンバータの電圧レギュレータを有効にします。
3. A/D コンバータを設定する起動時間の完了を待ちます（起動時間の値はデバイスのデータシートを参照してください）。

A/D コンバータの動作完了後は A/D コンバータ を無効化してもかまいません（ADEN = 0）。A/D コンバータの電圧レギュレータを無効にして電流消費を抑えることもできます。そのためには ビット ADVREGEN に 0 を書き込みます。

リーク電流を削減してさらに省電力化するには、ADC_CR レジスタのビット DEEPPWD を 1 に設定して A/D コンバータを再度ディープ･パワーダウン･モードに移行します。これは特に STOP モードに移行する前に効果的です。

注： **DEEPPWD = 1 を書き込むと、A/D コンバータの電圧レギュレータが自動的に無効になり、ビット ADVREGEN が自動的にクリアされます。**

内部電圧レギュレータが無効な状態（ADVREGEN = 0）でも、内部アナログ較正が保持されます。

A/D コンバータのディープ･パワーダウン･モード（DEEPPWD = 1）では内部アナログ較正が失われるので、較正を再実行するか、保存されている較正係数を再適用する必要があります（セクション 21.4.8：較正（ADCAL、ADCALDIF、ADC_CALFACT）を参照してください）。

21.4.7 シングルエンド入力および差動入力のチャネル

ADC_DIFSEL レジスタのビット DIFSEL[i] をプログラムすることで、チャネルをシングルエンド入力または差動入力のいずれかに設定できます。この設定は、A/D コンバータが無効（ADEN = 0）なときに書き込む必要があります。シングルエンド･チャネルに対応する DIFSEL[i] ビットは、必ず '0' にプログラムされています。

シングルエンド入力モードの場合、チャネル「i」で変換されるアナログ電圧は、$V_{INP[i]}$（正側入力）に等しい外部電圧 ADCy_INPx と V_{REF-}（負側入力）の差です。

差動入力モードの場合、チャネル「i」で変換されるアナログ電圧は、$V_{INP[i]}$ に等しい外部電圧の正側入力 ADCy_INPx と、$V_{INN[i]}$ に等しい負側入力 ADCy_INNx の差です。

差動モードの入力電圧範囲は V_{REF-} から V_{REF+} であるため、フル･スケール･レンジは 2 x V_{REF+} になります。$V_{INP[i]}$ が V_{REF-}、$V_{INN[i]}$ が V_{REF+} に等しい場合、負の最大入力差分電圧（V_{REF-}）は A/D コンバータ出力 0x000 に相当します。$V_{INP[i]}$ が V_{REF+}、$V_{INN[i]}$ が V_{REF-} に等しい場合、正の最大入力差分電圧（V_{REF+}）は A/D コンバータ出力 0xFFF に相当します。$V_{INP[i]}$ と $V_{INN[i]}$ を接続した場合の入力差分電圧 0 V は、A/D コンバータ出力 0x800 に相当します。

差動モードでの A/D コンバータの感度は、シングルエンド･モードの 1/2 に低下します。

A/D コンバータを差動モードに設定する場合、両方の入力に（V_{REF+}）/2 のバイアス電圧を適用する必要があります。許容コモン･モード入力電圧 V_{CMIN} の値はデバイスのデータシートを参照してください。

2 つの入力信号の差分が入力であると想定されます（コモン･モード電圧が変化しないことが必要です）。

内部チャネル（V_{TS} および V_{REFINT} など）はシングルエンドモードのみで使用されます。

各 ADC への入力チャネルの接続方法の詳細については、セクション 21.4.4 : ADC1/2/3/4/5 のコネクティビティを参照してください。

注意： **差動入力モードでチャネル「i」を設定すると、負側入力電圧 $V_{INN[i]}$ は別のチャネルに接続されます。その結果、そのチャネルはシングルエンド･モードと差動モードのどちらでも使用できなくなるので、変換入力として設定しないようにする必要があります。ADC1/ADC2/ADC3/ADC4/ADC5 間で共有されているチャネルもあります。これにより、これらの A/D コンバータの他方のチャネルが使用できなくなることがあります。唯一の例外は、ADC マスタ及びスレーブがインターリーブモードで動作するときです。**

21.4.8 較正（ADCAL、ADCALDIF、ADC_CALFACT）

各 ADC は、ADC のパワーオン／オフシーケンスを含むすべての較正シーケンスを駆動する、自動較正手順を提供しています。この処理時、ADC は、次の ADC のパワーオフまで ADC に内部で適用される、7 ビット幅の較正係数を計算します。較正処理時、アプリケーションでは ADC を使用してはならず、較正が完了するまで待つ必要があります。

較正は、すべての ADC 動作に先立って行われます。較正は、プロセスまたはバンドギャップのばらつきによりチップごとに異なるオフセットエラーを削除します。

シングルエンド入力変換に適用される較正係数は、差動入力変換に適用される係数とは異なります。

- シングルエンド入力変換に適用する較正を開始する前に、ADCALDIF = 0 を書き込みます。
- 差動入力変換に適用する較正を開始する前に、ADCALDIF = 1 を書き込みます。

つづいて、ソフトウェアでビット ADCAL = 1 を設定することで較正を開始します。A/D コンバータが無効のとき（ADEN = 0 のとき）にのみ較正を開始できます。すべての較正シーケンスで、ADCAL ビットは 1 のままです。較正が完了すると、ハードウェアによってクリアされます。このとき、関連する較正係数は、アナログ ADC と、ADC_CALFACT レジスタのビット CALFACT_S[6:0] または CALFACT_D[6:0] に内部的に格納されます（シングルエンドであるか差動入力較正であるかによって異なる）。

A/D コンバータが無効の場合（ADEN = 0）、内部アナログ較正が保持されます。ただし、A/D コンバータが長時間無効である場合は、A/D コンバータを再度有効にする前に、新しい較正サイクルを実行することを推奨します。

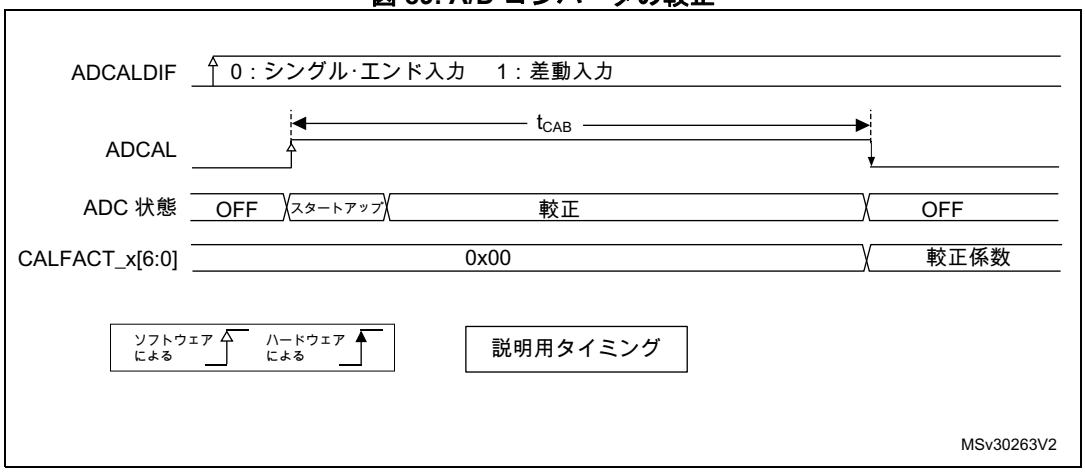
内部アナログ較正は A/D コンバータ の電源がオフになるたびに失われます（たとえば、製品が STANDBY または VBAT モードになった場合）。この場合、A/D コンバータの再較正に時間をかけるのを避けるためには、ソフトウェアが前回の較正時に較正係数を保存していた場合にかぎり、再較正せずに ADC_CALFACT レジスタに較正係数を再度書き込むことができます。

A/D コンバータが有効であり、変換中ではない場合（ADEN = 1、ADSTART = 0、および JADSTART = 0 である場合）に較正係数を書き込むことができます。その場合、次回の変換開始時に、較正係数がアナログ A/D コンバータに自動的にインジェクトされます。このローディングは透過的であり、変換開始のサイクル遅延は増加しません。V_{REF+} 電圧が 10% を超えて変化した場合、再較正することを推奨します。

ソフトウェアでの A/D コンバータの較正手順

1. DEEPPWD = 0、ADVREGEN = 1 であり、A/D コンバータの電圧レギュレータの起動時間が経過していることを確認します。
2. ADEN = 0 であることを確認します。
3. ADCALDIF = 0 （シングルエンド入力の場合）または ADCALDIF = 1 （差動入力の場合）を設定して、この較正の入力モードを選択します。
4. ADCAL = 1 を設定します。
5. ADCAL = 0 になるまで待ちます。
6. 較正係数は ADC_CALFACT レジスタから読み出すことができます。

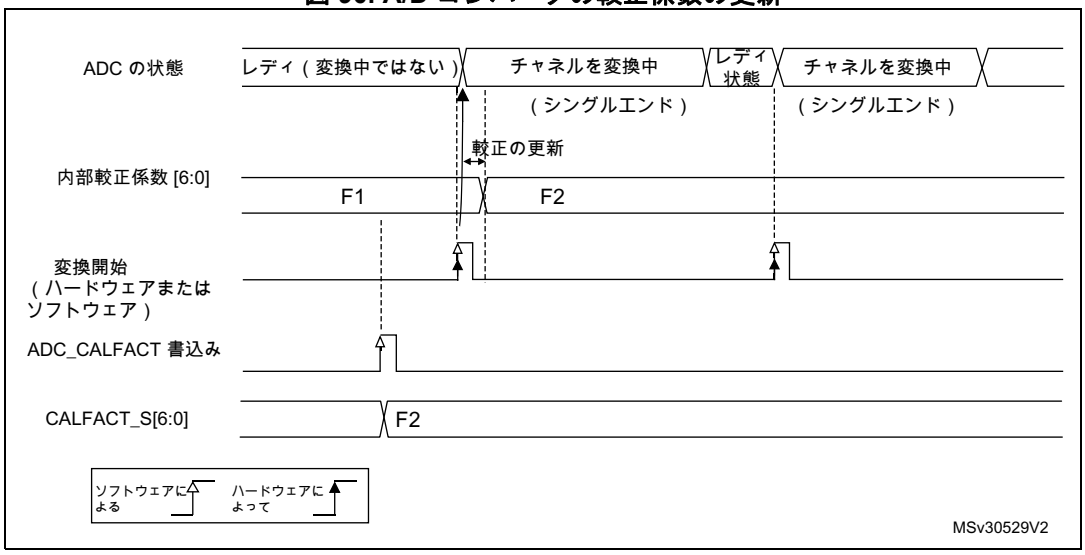
図 89. A/D コンバータの較正



ソフトウェアで A/D コンバータに較正係数を再インジェクトする手順

1. ADEN = 1、ADSTART = 0、および JADSTART = 0 である（A/D コンバータが有効で変換中でない）ことを確認します。
2. 新しい較正係数を CALFACT_S および CALFACT_D に書き込みます。
3. ビット CALFACT_S（シングルエンド入力チャネルの場合）またはビット CALFACT_D（差動入力チャネルの場合）に格納された較正係数と内部アナログ較正係数が異なる場合にのみ、変換を開始すると較正係数がアナログ A/D コンバータにインジェクトされます。

図 90. A/D コンバータの較正係数の更新

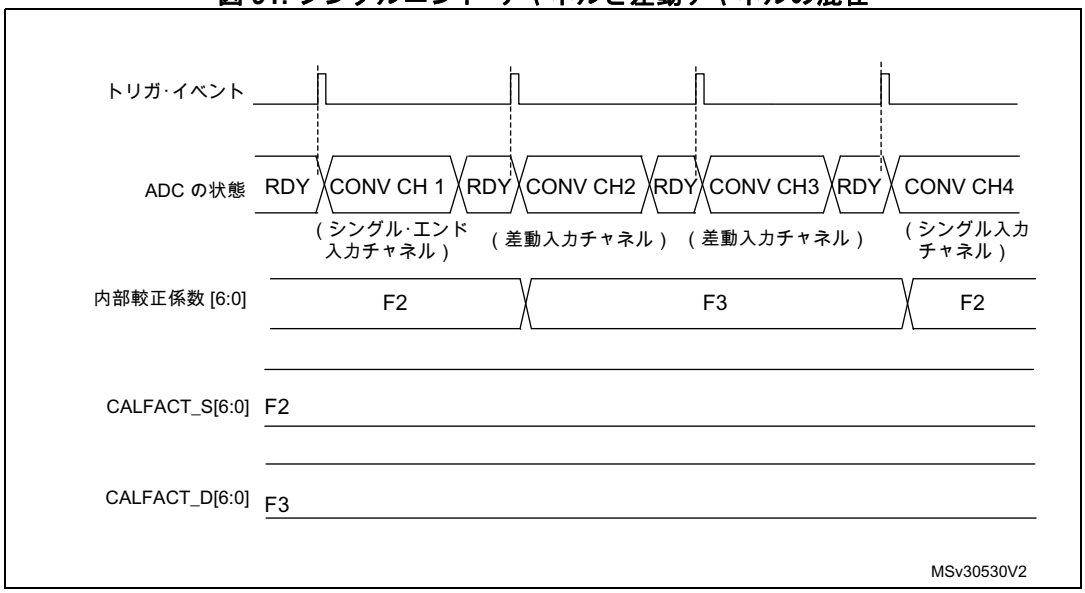


シングル ADC によるシングルエンドおよび差動アナログ入力の変換

差動入力とシングルエンド入力の両方を変換することを想定している A/D コンバータの場合、ADCALDIF = 0 とした較正と ADCALDIF= 1 とした較正の 2 つを実行する必要があります。手順は、次のとおりです。

1. ADC を無効にします。
2. シングルエンド入力モードで A/D コンバータを較正します（ADCALDIF = 0）。これにより、レジスタ CALFACT_S[6:0] が更新されます。
3. 差動入力モードで A/D コンバータを較正します（ADCALDIF = 1）。これにより、レジスタ CALFACT_D[6:0] が更新されます。
4. ADC を有効化し、チャネルを設定して変換を起動します。シングルエンド入力チャネルと差動入力チャネルを切り替えるたびに、アナログ A/D コンバータに自動的に較正値がインジェクトされます。

図 91. シングルエンド·チャネルと差動チャネルの混在



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

最初に、セクション 21.4.6 : ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）で説明している手順に従ってください。

DEEPPWD = 0 および ADVREGEN = 1 になれば A/D コンバータを有効化できますが、図 92 にあるように、正確な変換を開始するまでに安定化時間 t_{STAB} を必要とします。A/D コンバータは、次の 2 つの制御ビットによって有効または無効になります。

- ADEN = 1 とすることで A/D コンバータが有効になります。A/D コンバータの動作準備ができると、ADRDY フラグがセットされます。
- ADDIS = 1 とすることで A/D コンバータが無効になります。アナログ A/D コンバータが実質的に無効になると、ADEN および ADDIS はハードウェアによって自動的にクリアされます。

その後、ADSTART = 1 に設定するか（セクション 21.4.18 : 外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN、JEXTSEL、JEXTEN）を参照）、トリガが有効な場合に外部トリガ·イベントが発生したときにレギュラ変換を開始できます。

JADSTART = 1 に設定するか、インジェクト·トリガが有効な場合に外部インジェクト·トリガ·イベントが発生したときに、インジェクト変換が開始されます。

ソフトウェアで A/D コンバータを有効にする手順

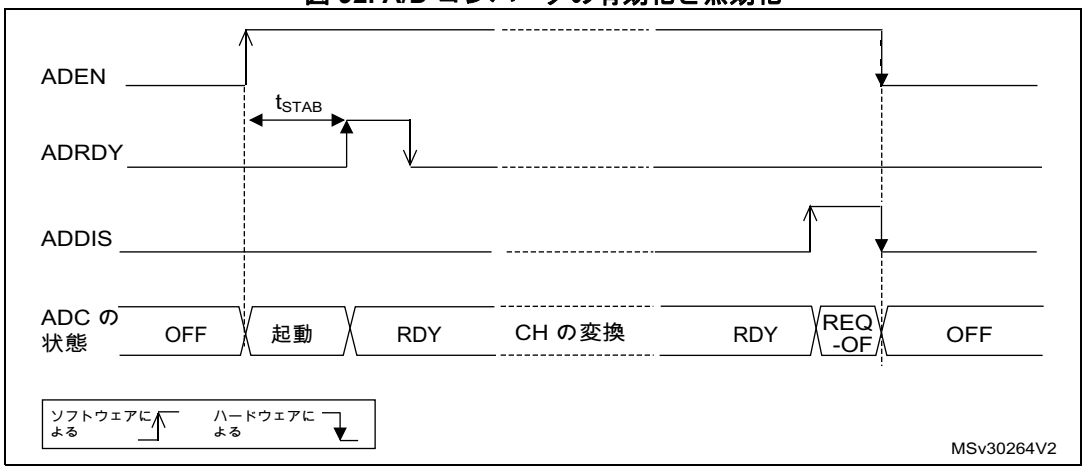
1. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします。
2. ADEN = 1 に設定します。
3. ADRDY = 1 になるまで待ちます（ADRDY は A/D コンバータの起動時間後にセットされます）。この時間を待機するには、該当の割込みを使用します（ADRDYIE = 1 に設定します）。
4. ADC_ISR レジスタの ADRDY ビットに「1」を書き込んでクリアします（オプション）。

注意： **ADCAL がセットされている場合、ADCAL ビットがハードウェアによってクリアされた（較正の終了）後の 4 回の A/D コンバータのクロック·サイクル中は、ADEN ビットをセットできません。**

ソフトウェアでの ADC の無効化手順

1. ADSTART = 0 および JADSTART = 0 であること、および変換が実行中でないことを確認します。必要な場合は、ADSTP = 1 および JADSTP= 1 に設定して、実行中のレギュラ変換とインジェクト変換を停止したうえで、ADSTP = 0 および JADSTP = 0 になるまで待ちます。
2. ADDIS = 1 に設定します。
3. アプリケーションで必要な場合、ADEN = 0 になり、アナログ A/D コンバータが実質的に無効になるまで待ちます（ADEN = 0 になると、ADDIS は自動的にリセットされます）。

図 92. A/D コンバータの有効化と無効化



21.4.10 A/D コンバータ制御ビット書込み時の制約

ソフトウェアでは、A/D コンバータが無効である場合のみ（ADEN は 0 である必要があります）、A/D コンバータのクロック（RCC のセクションを参照）、ADC_DIFSEL レジスタの制御ビット DIFSEL[i]、および ADC_CR レジスタの制御ビット ADCAL と ADEN を設定して有効化するために、RCC 制御ビットを書き込むことができます。

A/D コンバータが有効であり、A/D コンバータを無効にする保留中のリクエストがない場合のみ（ADEN = 1 および ADDIS = 0 である必要があります）、ADC_CR レジスタの制御ビット ADSTART、JADSTART、および ADDIS をソフトウェアで書き込むことができます。

ADC_CFGR、ADC_SMPRx、ADC_TRy、ADC_SQRy、ADC_JDRy、ADC_OFRy、ADC_OFCHRy、および ADC_IER レジスタのその他すべての制御ビットについては、次のとおりです。

- レギュラ変換の設定に関連する制御ビットは、A/D コンバータが有効（ADEN = 1）であり、実行中のレギュラ変換がない（ADSTART = 0）場合のみ、ソフトウェアで書き込むことができます。
- インジェクト変換の設定に関連する制御ビットは、A/D コンバータが有効（ADEN = 1）であり、実行中のインジェクト変換がない（JADSTART = 0）場合のみ、ソフトウェアで書き込むことができます。
- A/D 変換の実行中に ADC_TRy レジスタの内容を変更できます（詳細については、セクション 21.4.28 : アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照）。

ADC が有効で、変換可能の場合、および ADC を無効にするための保留中のリクエストがない場合のみ（ADSTART = 1 または JADSTART = 1、かつ ADDIS = 0）、ソフトウェアは ADC_CR レジスタの制御ビット ADSTP または JADSTP を書き込むことができます。

ソフトウェアは、A/D コンバータが有効である場合（ADEN = 1）、いつでもレジスタ ADC_JSQR に書き込むことができます。詳細については、セクション 21.7.16 : ADC インジェクトシーケンスレジスタ（ADC_JSQR）を参照してください。

注： **これらの禁止された書込みアクセスを防ぐハードウェア保護はありません。ADCの挙動は不明な状態になる場合があります。この状況から復帰するには、A/D コンバータを無効にする必要があります（ADEN と ADC_CR レジスタのすべてのビットをクリアします）。**

21.4.11 チャネルの選択（SQRx、JSQRx）

ADC ごとに最大 19 の多重化チャネルがあります。

- GPIO パッドからの最大 13個の低速アナログ入力（ADCx_INP/INN[6:18]）
  製品によっては、GPIO パッドを使用できないものがあります。
- A/D コンバータは、次の内部アナログ入力に接続されています。
  - 内部基準電圧（V_{REFINT}）は ADC1_INP18、ADC3_INP18、ADC4_INP18、ADC5_INP18 に接続されています。
  - 内部温度センサ（V_{TS}）は ADC1_INP16 および ADC5_INP4 に接続されています。
  - V_{BAT} 監視チャネル（$V_{BAT}/3$）は ADC1_INP17、ADC3_INP17、および ADC5_INP17 に接続されています。

注： **内部アナログチャネルのうちの 1 つを変換するには、最初に ADCx_CCR レジスタのビット VREFEN、VBATSEL、または VSENSESEL をプログラムして、対応するアナログソースを有効化する必要があります。**

変換は、レギュラとインジェクトの 2 つのグループに構成することができます。各グループは、任意のチャネルに対して任意の順序で行うことができる一連の変換で構成されます。たとえば、次のような順で変換順を設定することができます：ADCx_INP/INN3、ADCx_INP/INN8、ADCx_INP/INN2、ADCx_INN/INP2、ADCx_INP/INN0、ADCx_INP/INN2、ADCx_INP/INN2、ADCx_INP/INN15

- **レギュラグループ**は、最大 16 の変換で構成されます。レギュラチャネルと変換シーケンス内での順序は、ADC_SQRy レジスタで選択する必要があります。レギュラグループの合計変換数は、ADC_SQR1 レジスタの L[3:0] ビットに書き込む必要があります。
- **インジェクトグループ**は、最大 4 つの変換で構成されます。インジェクトチャネルと変換シーケンス内での順序は、ADC_JSQR レジスタで選択する必要があります。インジェクトグループの合計変換数は、ADC_JSQR レジスタの L[1:0] ビットに書き込む必要があります。

レギュラ変換が実行される可能性がある場合は、ADC_SQRy レジスタを変更しないでください。この場合、最初に ADSTP = 1 を書き込んで、A/D コンバータのレギュラ変換を停止する必要があります（セクション 21.4.17：実行中の変換の停止（ADSTP、JADSTP）を参照）。

JADSTART がセットされているとき（インジェクト変換実行中）にソフトウェアが ADC_JSQR レジスタを動作中に変更できるのは、コンテキスト･キューが有効になっている（ADC_CFGR レジスタの JQDIS = 0）場合だけです。セクション 21.4.21：インジェクト変換のコンテキストのキューを参照してください。

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

変換を開始する前に、ADC は測定する電圧ソースと ADC の内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースが内蔵コンデンサを入力電圧レベルまでチャージできるだけの十分な長さが必要です。

各チャネルは、ADC_SMPR1 および ADC レジスタの SMP[2:0] ビットを使用してプログラムすることで異なるサンプリング時間でサンプリングできます。したがって、次のサンプリング時間の値から選択することができます。

- SMP = 000：2.5 ADC クロックサイクル
- SMP = 001：6.5 ADC クロックサイクル
- SMP = 010：12.5 ADC クロックサイクル
- SMP = 011：24.5 ADC クロックサイクル
- SMP = 100：47.5 ADC クロックサイクル
- SMP = 101：92.5 ADC クロックサイクル
- SMP = 110：247.5 ADC クロックサイクル
- SMP = 111：640.5 ADC クロックサイクル

合計変換時間は、次のように計算されます。

T_{CONV} = サンプリング時間 + 12.5 ADC クロックサイクル

例：

$F_{adc_ker_ck}$ = 30 MHz、サンプリング時間 = 2.5 ADC クロックサイクル：

T_{CONV} = （2.5 + 12.5）ADC クロックサイクル = 15 ADC クロックサイクル = 500 ns

ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。

サンプリング時間の制約

各チャネルの SMP[2:0] ビットは、データシートの A/D コンバータの特性に関するセクションで指定された最小サンプリング時間を遵守してプログラムする必要があります。

バルブサンプリングモード

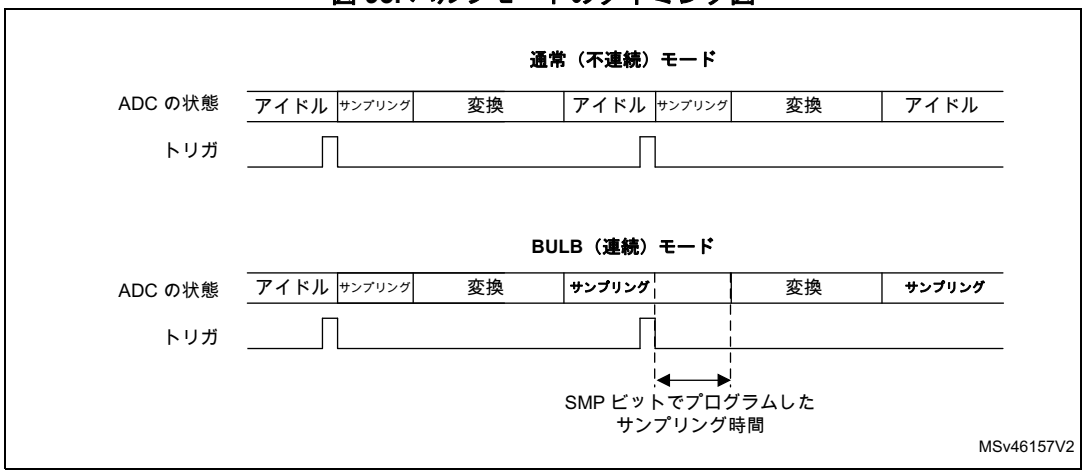
ADC レジスタで BULB ビットがセットされているとき、サンプリング期間は最後の ADC 変換の直後に開始します。ハードウェアまたはソフトウェアトリガは、サンプリング時間が ADC_SMPR1 レジスタでプログラムされた後、変換を開始します。最初の ADC 変換は、ADC が有効化された後、SMP ビットでプログラムされたサンプリング時間で実行されます。バルブモードは、2 番目の変換から開始するのに有効です。

最大サンプリング時間は制限されます（データシートの ADC 特性のセクションを参照）。

バルブモードは、連続変換モードとも、インジェクトチャネル変換とも互換性がありません。

BULB ビットがセットされているとき、ADC_CFGR2 の SMPTRIG ビットをセットすることはできません。

図 93. バルブモードのタイミング図



サンプリング時間制御トリガモード

SMPTRIG ビットがセットされているとき、SMPx ビットによってプログラムされたサンプリング時間は適用されません。サンプリング時間は、トリガ信号エッジによって制御されます。

ハードウェアトリガが選択されたとき、トリガ信号の各立ち上がりエッジによってサンプリング期間は開始します。立ち下がりエッジによってサンプリング期間は終了し、変換が開始されます。EXTEN[1:0] ビットは 01 に設定する必要があります。立ち上がりエッジと立ち下がりエッジが未定義のハードウェア･トリガ（1 パルス･イベント）はバルブ･モードでは使用できません。

ソフトウェア･トリガが選択されたとき、ソフトウェア･トリガは ADC_CR の ADSTART ビットではなく、SWTRIG ビットです。サンプリング期間を開始するには、SWTRIG ビットをセットする必要があり、サンプリング期間を終了して変換を開始するには、SWTRIG ビットをクリアする必要があります。EXTEN[1:0] ビットは 00 に設定する必要があります。

最大サンプリング時間は制限されます（データシートの A/D コンバータの特性に関するセクションを参照）。

このモードは、連続変換モードとも、インジェクトチャネル変換とも互換性がありません。

SMPTRIG ビットがセットされているとき、BULB ビットはセットできません。

I/O アナログスイッチ電圧ブースタ

V_{DDA} 電圧が低すぎる場合に、I/O アナログスイッチ抵抗が増加します。これには、状況に応じて適用されるサンプリング時間が必要になります（電気的特性に関するデータシート参照）。SYSCFG_CFGR1 レジスタの BOOSTEN ビットで内部電圧ブースタを有効にすることによって、低い V_{DDA} でもこの抵抗を最小化できます。

SMPPLUS 制御ビット

2.5 ADC クロックサイクルのサンプリング時間が選択されると、合計変換時間は 12 ビットモードで 15 サイクルになります。デュアルインタリーブモードが使用されている場合（セクション ：独立インジェクト変換に対応したインタリーブモード参照）、変換に偶数のサイクル数が必要になるため、サンプリング間隔は指定された値と同じものにできません。SMPPLUS ビットは、サンプリング時間 2.5 ADC クロックサイクルを 3.5 ADC クロックサイクルに変更するのに使用できます。この方法では、合計変換時間は 16 クロックサイクルになるため、各 8 サイクルのインタリーブが可能になります。

21.4.13 シングル変換モード（CONT = 0）

シングル変換モードでは、A/D コンバータはチャネルのすべての変換を一度実行します。CONT ビットが 0 の時、次のいずれかによってこのモードが開始します。

- ADC_CR レジスタの ADSTART ビットをセット（レギュラチャネル）
- ADC_CR レジスタの JADSTART ビットをセット（インジェクトチャネル）
- 外部ハードウェアトリガイベント（レギュラまたはインジェクトチャネル）

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC （レギュラ変換の完了）フラグがセットされます。
- EOCIE ビットがセットされている場合、割込みが生成されます。

インジェクトシーケンス内で、各変換の完了後、

- 変換データは 4 つの 16 ビット ADC_JDRy レジスタのうちの 1 つに格納されます。
- JEOC （インジェクト変換の完了）フラグがセットされます。
- JEOCIE ビットがセットされている場合、割込みが生成されます。

レギュラシーケンスの完了後、

- EOS （レギュラシーケンスの完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割込みが生成されます。

インジェクトシーケンスの完了後、

- JEOS （インジェクトシーケンスの完了）フラグがセットされます。
- JEOSIE ビットがセットされている場合、割込みが生成されます。

次に、ADC は、新しい外部レギュラトリガまたは外部インジェクトトリガが発生するか、ADSTART または JADSTART ビットが再びセットされるまで停止します。

注： **単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。**

21.4.14 連続変換モード（CONT = 1）

このモードはレギュラ･チャネルにのみ適用されます。

連続変換モードでは、ソフトウェアまたはハードウェアのレギュラトリガイベントが発生すると、ADC はチャネルのレギュラ変換を一度すべて実行し、自動的に再起動して、シーケンスの各変換を連続的に実行します。CONT ビットが 1 の時､外部トリガまたは ADC_CR レジスタの ADSTART ビットをセットすることによって、このモードが開始されます。

レギュラシーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC（end of conversion）フラグがセットされます。
- EOCIE ビットがセットされている場合、割込みが生成されます。

変換シーケンスの完了後、

- EOS （シーケンス完了）フラグがセットされます。
- EOSIE ビットがセットされている場合、割込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADC は変換シーケンスを連続的に繰り返します。

注： **単一チャネルを変換するには、長さが 1 のシーケンスをプログラムします。**

不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1 と CONT = 1 の両方を設定することは禁じられています。

インジェクト･チャネルを連続して変換することはできません。唯一の例外は、連続モードに設定されたレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTO ビットを使用）場合です。自動インジェクションモードのセクションを参照してください。

21.4.15 変換の開始（ADSTART、JADSTART）

ソフトウェアは、ADSTART = 1 に設定することで A/D コンバータのレギュラ変換を開始します。

ADSTART をセットすると変換の開始は次のようになります。

- EXTEN[1:0] = 00（ソフトウェア･トリガ）の場合、すぐに開始します。
- EXTEN[1:0] ≠ 00 の場合、選択されたレギュラ･ハードウェア･トリガの次のアクティブ･エッジで開始します。

ソフトウェアは、JADSTART = 1 に設定することで A/D コンバータのインジェクト変換を開始します。

JADSTART をセットすると変換の開始は次のようになります。

- JEXTEN[1:0] = 00（ソフトウェア･トリガ）の場合、すぐに開始します。
- JEXTEN[1:0] ≠ 00 の場合、選択されたインジェクト･ハードウェア･トリガの次のアクティブ･エッジで開始します。

注： **自動インジェクション･モード（JAUTO = 1）では、ADSTART ビットを使用してレギュラ変換を開始し、つづいて自動インジェクト変換を実行します（JADSTART はクリア状態に維持する必要があります）。**

ADSTART および JADSTART は、A/D コンバータの動作が実行中かどうかを示す情報も提供します。ADSTART = 0 と JADSTART = 0 が両方とも真で A/D コンバータがアイドルであることを示しているときには、A/D コンバータを再設定できます。

次のときに、ADSTART はハードウェアによってクリアされます。

- シングル･モード時のソフトウェアのレギュラ･トリガ（CONT= 0、EXTSEL = 0x0）
  - レギュラ変換シーケンスの終了時（EOS のアサート）、または DISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（CONT=x、EXTSEL=x）
  - ソフトウェアによってアサートされた ADSTP 手順の実行後。

注： **連続モード（CONT = 1）では、シーケンスは自動的に再起動されるので、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。**

シングル･モードでハードウェア･トリガを選択すると（CONT = 0 かつ EXTSEL ≠ 0x00）、次のハードウェア･トリガ･イベントで ADSTART の再リセットを必要としないソフトウェアを支援するために、EOS がアサートされても、ADSTART はハードウェアによってクリアされません。これにより、以降のハードウェア･トリガの見落としを防ぐことができます。

次のときに、JADSTART はハードウェアによってクリアされます。

- シングル･モード時のソフトウェアのインジェクト･トリガ（JEXTSEL = 0x0）
  - インジェクト変換シーケンスの終了時（JEOS のアサート）、または JDISCEN = 1 の場合のサブグループ処理の終了時
- すべての場合に（JEXTSEL=x）
  - ソフトウェアによってアサートされた JADSTP 手順の実行後。

注： **ソフトウェアトリガが選択されているとき、EOC フラグがまだハイの場合に ADSTART ビットをセットしてはいけません。**

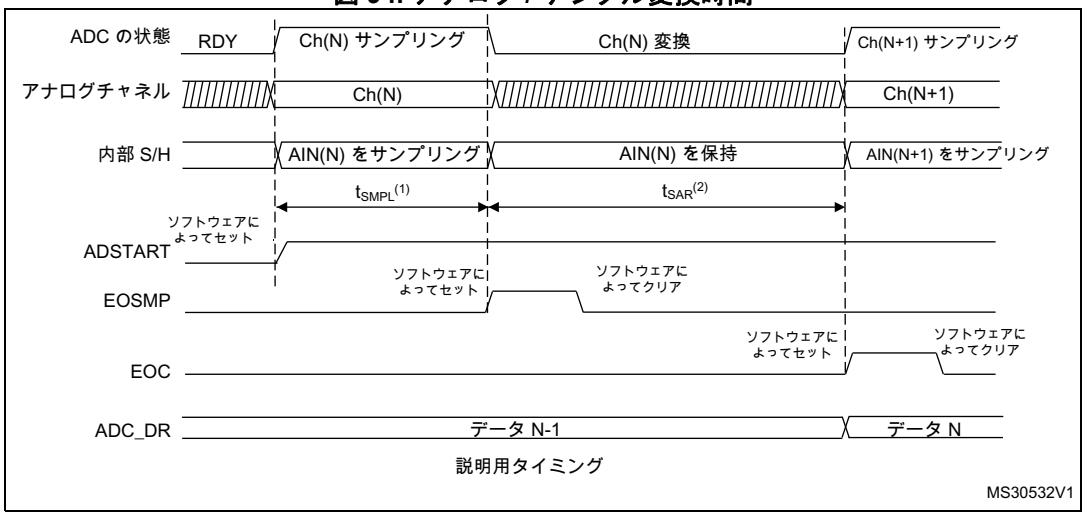
21.4.16 ADC のタイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間（データ分解能に依存）を加えた合計です。

$T_{CONV} = T_{SMPL} + T_{SAR} = [2.5\ _{|min} + 12.5\ _{|12bit}] \times T_{ADC_CLK}$

$T_{CONV} = T_{SMPL} + T_{SAR} = 83.33\ ns\ _{|min} + 416.67\ ns\ _{|12bit} = 500.0\ ns$（$F_{ADC_CLK} = 30\ MHz$ の場合）

図 94. アナログ / デジタル変換時間



1. T_{SMPL} は SMP[2:0] に依存します。
2. T_{SAR} は RES[2:0] に依存します。

21.4.17 実行中の変換の停止（ADSTP、JADSTP）

ソフトウェアは、ADSTP = 1 に設定することで実行中のレギュラ変換を停止でき、JADSTP = 1 に設定することで実行中のインジェクト変換を停止できます。

変換を停止すると、A/D コンバータで実行中の動作がリセットされます。つづいて、A/D コンバータを再設定して（チャネル選択やトリガの変更など）、新しい動作に備えることができます。

レギュラ変換の実行中に、インジェクト変換を停止することもできる点に注意してください（その逆も可能）。これにより、たとえば、レギュラ変換の実行中にインジェクト変換シーケンスやトリガを再設定することができます（その逆も可能）。

ADSTP ビットがソフトウェアによってセットされると、実行中のレギュラ変換は中止され、部分的な結果は破棄されます（ADC_DR レジスタは現在の変換では更新されません）。

JADSTP ビットがソフトウェアによってセットされると、実行中のインジェクト変換は中止され、部分的な結果は破棄されます（ADC_JDRy レジスタは現在の変換では更新されません）。スキャンシーケンスも中止され、リセットされます（ADC を再起動すると、新しいシーケンスが再開されることを意味します）。

この手順が完了すると、ADSTP/ADSTART ビット（レギュラ変換の場合）、または JADSTP/JADSTART ビット（インジェクト変換の場合）がハードウェアによってクリアされ、ソフトウェアは ADC が完全に停止する前にビットがリセットされるまで ADSTART（または JADSTART）をポーリングする必要があります。

注： **自動インジェクション・モード（JAUTO = 1）では、ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方が中止されます（JADSTP を使用しないでください）。**

図 95. 実行中のレギュラ変換の停止

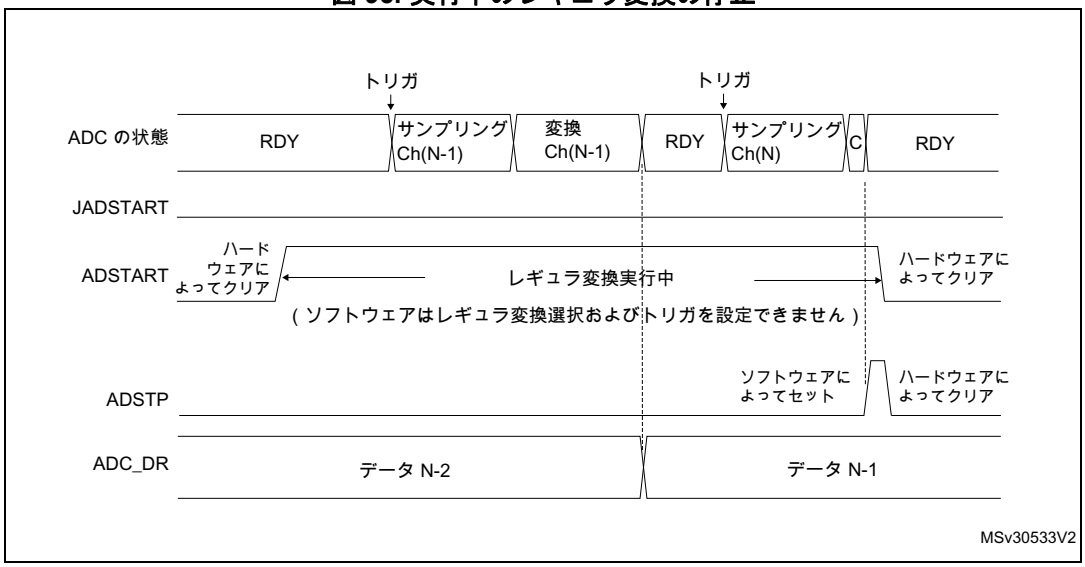
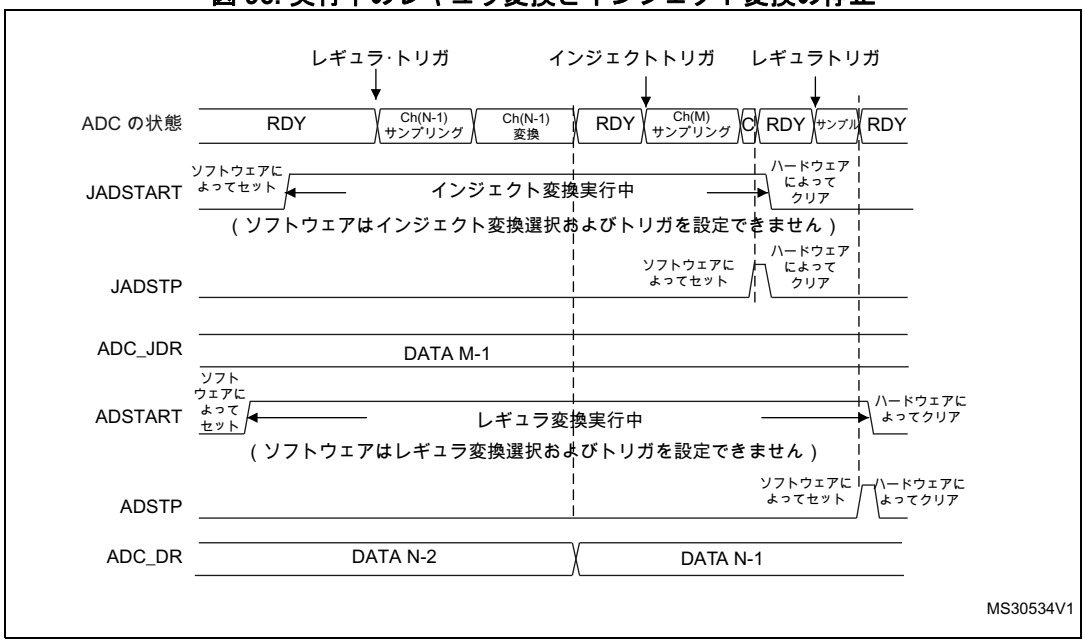


図 96. 実行中のレギュラ変換とインジェクト変換の停止



21.4.18 外部トリガによる変換およびトリガ極性（EXTSEL、EXTEN、JEXTSEL、JEXTEN）

変換または変換シーケンスは、ソフトウェアまたは外部イベント（タイマ･キャプチャや入力ピンなど）によってトリガできます。EXTEN[1:0] 制御ビット（レギュラ変換の場合）または JEXTEN[1:0] ビット（インジェクト変換の場合）が 00 以外の場合、外部イベントは選択した極性で変換をトリガできます。

インジェクト･キューが有効である場合（ビット JQDIS = 0）、インジェクト･ソフトウェア･トリガは使用できません。

レギュラ･トリガの選択は、ソフトウェアでビット ADSTART=1 に設定すると有効になります。インジェクト･トリガの選択は、ソフトウェアでビット JADSTART = 1 に設定すると有効になります。

変換中に発生したハードウェア･トリガは無視されます。

- ビット ADSTART = 0 の場合、発生したレギュラ･ハードウェア･トリガは無視されます。
- ビット JADSTART = 0 の場合、発生したインジェクト･ハードウェア･トリガは無視されます。

表 161 に EXTEN[1:0] と JEXTEN[1:0] の値とトリガ極性の対応を示します。

表 161. レギュラ外部トリガのトリガ極性の設定

EXTEN[1:0]	転送元
00	ハードウェアトリガ検出が無効になり、ソフトウェアトリガ検出が有効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： **レギュラトリガの極性は動作中に変更することはできません。**

表 162. インジェクト外部トリガのトリガ極性の設定

JEXTEN[1:0]	転送元
00	– JQDIS = 1（キュー無効）の場合：ハードウェア･トリガ検出が無効になり、ソフトウェア･トリガ検出が有効になります。 – JQDIS = 0（キュー有効）の場合、ハードウェアおよびソフトウェア･トリガ検出は無効になります。
01	立ち上がりエッジで検出するハードウェアトリガ
10	立ち下がりエッジで検出するハードウェアトリガ
11	立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： **キューが有効（JQDIS = 0）になっている場合、インジェクト･トリガの極性を動作中に予測して変更できます。セクション 21.4.21：インジェクト変換のコンテキストのキューを参照してください。**

EXTSEL および JEXTSEL 制御ビットでは、レギュラおよびインジェクトグループの変換をトリガできるイベントを 32 のイベントから選択します。

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： **レギュラトリガの選択は動作中に変更することはできません。**
インジェクトトリガの選択は動作中に予測して変更できます。セクション 21.4.21：623 ページのインジェクト変換のコンテキストのキューを参照してください。

図 97 に示すように、各 ADC マスタは ADC スレーブと同じ入力トリガを共有しています。

図 97. ADC マスタと ADC スレーブ間でのトリガの共有

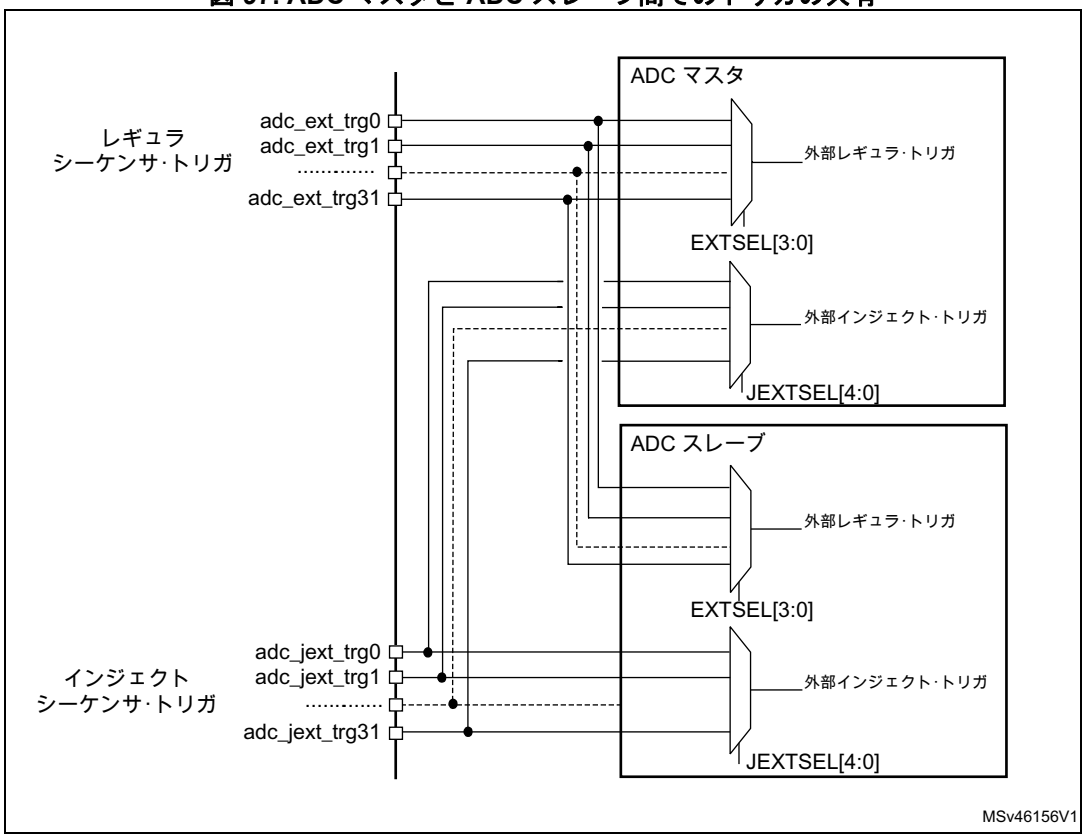


表 163 から表 166 に、レギュラ変換とインジェクト変換に使用できる 3 つの A/D コンバータのすべての外部トリガを示します。

表 163. ADC1/2 - レギュラ·チャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg	TIM1_CC1	オンチップ·タイマからの内部信号	00000
adc_ext_trg1	TIM1_CC2	オンチップ·タイマからの内部信号	00001
adc_ext_trg2	TIM1_CC3	オンチップ·タイマからの内部信号	00010
adc_ext_trg3	TIM2_CC2	オンチップ·タイマからの内部信号	00011
adc_ext_trg4	TIM3_TRGO	オンチップ·タイマからの内部信号	00100
adc_ext_trg5	TIM4_CC4	オンチップ·タイマからの内部信号	00101
adc_ext_trg6	EXTI ライン 11	外部ピン	00110
adc_ext_trg7	TIM8_TRGO	オンチップ·タイマからの内部信号	00111
adc_ext_trg8	TIM8_TRGO2	オンチップ·タイマからの内部信号	01000
adc_ext_trg9	TIM1_TRGO	オンチップ·タイマからの内部信号	01001
adc_ext_trg10	TIM1_TRGO2	オンチップ·タイマからの内部信号	01010
adc_ext_trg11	TIM2_TRGO	オンチップ·タイマからの内部信号	01011
adc_ext_trg12	TIM4_TRGO	オンチップ·タイマからの内部信号	01100

表 163. ADC1/2 - レギュラ･チャネルの外部トリガ （続き）

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg13	TIM6_TRGO	オンチップ･タイマからの内部信号	01101
adc_ext_trg14	TIM15_TRGO	オンチップ･タイマからの内部信号	01110
adc_ext_trg15	TIM3_CC4	オンチップ･タイマからの内部信号	01111
adc_ext_trg16	TIM20_TRGO	オンチップ･タイマからの内部信号	10000
adc_ext_trg17	TIM20_TRGO2	オンチップ･タイマからの内部信号	10001
adc_ext_trg18	TIM20_CC1	オンチップ･タイマからの内部信号	10010
adc_ext_trg19	TIM20_CC2	オンチップ･タイマからの内部信号	10011
adc_ext_trg20	TIM20_CC3	オンチップ･タイマからの内部信号	10100
adc_ext_trg21	hrtim_adc_trg1	オンチップ･タイマからの内部信号	10101
adc_ext_trg22	hrtim_adc_trg3	オンチップ･タイマからの内部信号	10110
adc_ext_trg23	hrtim_adc_trg5	オンチップ･タイマからの内部信号	10111
adc_ext_trg24	hrtim_adc_trg6	オンチップ･タイマからの内部信号	11000
adc_ext_trg25	hrtim_adc_trg7	オンチップ･タイマからの内部信号	11001
adc_ext_trg26	hrtim_adc_trg8	オンチップ･タイマからの内部信号	11010
adc_ext_trg27	hrtim_adc_trg9	オンチップ･タイマからの内部信号	11011
adc_ext_trg28	hrtim_adc_trg10	オンチップ･タイマからの内部信号	11100
adc_ext_trg29	LPTIMOUT	オンチップタイマからの内部信号	11101
adc_ext_trg30	TIM7_TRGO	オンチップ･タイマからの内部信号	11110
adc_ext_trg31	予約済み	-	11111

表 164. ADC1/2 - インジェクトチャネルの外部トリガ

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg0	TIM1_TRGO	オンチップ･タイマからの内部信号	00000
adc_jext_trg1	TIM1_CC4	オンチップ･タイマからの内部信号	00001
adc_jext_trg2	TIM2_TRGO	オンチップ･タイマからの内部信号	00010
adc_jext_trg3	TIM2_CC1	オンチップ･タイマからの内部信号	00011
adc_jext_trg4	TIM3_CC4	オンチップ･タイマからの内部信号	00100
adc_jext_trg5	TIM4_TRGO	オンチップ･タイマからの内部信号	00101
adc_jext_trg6	EXTI ライン 15	外部ピン	00110
adc_jext_trg7	TIM8_CC4	オンチップ･タイマからの内部信号	00111
adc_jext_trg8	TIM1_TRGO2	オンチップ･タイマからの内部信号	01000
adc_jext_trg9	TIM8_TRGO	オンチップ･タイマからの内部信号	01001
adc_jext_trg10	TIM8_TRGO2	オンチップ･タイマからの内部信号	01010
adc_jext_trg11	TIM3_CC3	オンチップ･タイマからの内部信号	01011
adc_jext_trg12	TIM3_TRGO	オンチップ･タイマからの内部信号	01100
adc_jext_trg13	TIM3_CC1	オンチップ･タイマからの内部信号	01101
adc_jext_trg14	TIM6_TRGO	オンチップ･タイマからの内部信号	01110

表 164. ADC1/2 - インジェクトチャネルの外部トリガ （続き）

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg15	TIM15_TRGO	オンチップ･タイマからの内部信号	01111
adc_jext_trg16	TIM20_TRGO	オンチップ･タイマからの内部信号	10000
adc_jext_trg17	TIM20_TRGO2	オンチップ･タイマからの内部信号	10001
adc_jext_trg18	TIM20_CC4	オンチップ･タイマからの内部信号	10010
adc_jext_trg19	hrtim_adc_trg2	オンチップ･タイマからの内部信号	10011
adc_jext_trg20	hrtim_adc_trg4	オンチップ･タイマからの内部信号	10100
adc_jext_trg21	hrtim_adc_trg5	オンチップ･タイマからの内部信号	10101
adc_jext_trg22	hrtim_adc_trg6	オンチップ･タイマからの内部信号	10110
adc_jext_trg23	hrtim_adc_trg7	オンチップ･タイマからの内部信号	10111
adc_jext_trg24	hrtim_adc_trg8	オンチップ･タイマからの内部信号	11000
adc_jext_trg25	hrtim_adc_trg9	オンチップ･タイマからの内部信号	11001
adc_jext_trg26	hrtim_adc_trg10	オンチップ･タイマからの内部信号	11010
adc_jext_trg27	TIM16_CC1	オンチップ･タイマからの内部信号	11011
adc_jext_trg28	予約済み	-	11100
adc_jext_trg29	LPTIMOUT	オンチップタイマからの内部信号	11101
adc_jext_trg30	TIM7_TRGO	オンチップ･タイマからの内部信号	11110
adc_jext_trg31	予約済み	-	11111

表 165. ADC3/4/5 - レギュラチャネルの外部トリガ

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg0	TIM3_CC1	オンチップ･タイマからの内部信号	00000
adc_ext_trg1	TIM2_CC3	オンチップ･タイマからの内部信号	00001
adc_ext_trg2	TIM1_CC3	オンチップ･タイマからの内部信号	00010
adc_ext_trg3	TIM8_CC1	オンチップ･タイマからの内部信号	00011
adc_ext_trg4	TIM3_TRGO	オンチップ･タイマからの内部信号	00100
adc_ext_trg5	EXTI ライン 2	外部ピン	00101
adc_ext_trg6	TIM4_CC1	オンチップ･タイマからの内部信号	00110
adc_ext_trg7	TIM8_TRGO	オンチップ･タイマからの内部信号	00111
adc_ext_trg8	TIM8_TRGO2	オンチップ･タイマからの内部信号	01000
adc_ext_trg9	TIM1_TRGO	オンチップ･タイマからの内部信号	01001
adc_ext_trg10	TIM1_TRGO2	オンチップ･タイマからの内部信号	01010
adc_ext_trg11	TIM2_TRGO	オンチップ･タイマからの内部信号	01011
adc_ext_trg12	TIM4_TRGO	オンチップ･タイマからの内部信号	01100
adc_ext_trg13	TIM6_TRGO	オンチップ･タイマからの内部信号	01101
adc_ext_trg14	TIM15_TRGO	オンチップ･タイマからの内部信号	01110
adc_ext_trg15	TIM2_CC1	オンチップ･タイマからの内部信号	01111

表 165. ADC3/4/5 - レギュラチャネルの外部トリガ （続き）

名前	転送元	タイプ	EXTSEL[4:0]
adc_ext_trg16	TIM20_TRGO	オンチップ･タイマからの内部信号	10000
adc_ext_trg17	TIM20_TRGO2	オンチップ･タイマからの内部信号	10001
adc_ext_trg18	TIM20_CC1	オンチップ･タイマからの内部信号	10010
adc_ext_trg19	hrtim_adc_trg2	オンチップ･タイマからの内部信号	10011
adc_ext_trg20	hrtim_adc_trg4	オンチップ･タイマからの内部信号	10100
adc_ext_trg21	hrtim_adc_trg1	オンチップ･タイマからの内部信号	10101
adc_ext_trg22	hrtim_adc_trg3	オンチップ･タイマからの内部信号	10110
adc_ext_trg23	hrtim_adc_trg5	オンチップ･タイマからの内部信号	10111
adc_ext_trg24	hrtim_adc_trg6	オンチップ･タイマからの内部信号	11000
adc_ext_trg25	hrtim_adc_trg7	オンチップ･タイマからの内部信号	11001
adc_ext_trg26	hrtim_adc_trg8	オンチップ･タイマからの内部信号	11010
adc_ext_trg27	hrtim_adc_trg9	オンチップ･タイマからの内部信号	11011
adc_ext_trg28	hrtim_adc_trg10	オンチップ･タイマからの内部信号	11100
adc_ext_trg29	LPTIMOUT	オンチップタイマからの内部信号	11101
adc_ext_trg30	TIM7_TRGO	オンチップ･タイマからの内部信号	11110
adc_ext_trg31	予約済み	-	11111

表 166. ADC3/4/5 - インジェクトチャネルの外部トリガ

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg0	TIM1_TRGO	オンチップ･タイマからの内部信号	00000
adc_jext_trg1	TIM1_CC4	オンチップ･タイマからの内部信号	00001
adc_jext_trg2	TIM2_TRGO	オンチップ･タイマからの内部信号	00010
adc_jext_trg3	TIM8_CC2	オンチップ･タイマからの内部信号	00011
adc_jext_trg4	TIM4_CC3	オンチップ･タイマからの内部信号	00100
adc_jext_trg5	TIM4_TRGO	オンチップ･タイマからの内部信号	00101
adc_jext_trg6	TIM4_CC4	オンチップ･タイマからの内部信号	00110
adc_jext_trg7	TIM8_CC4	オンチップ･タイマからの内部信号	00111
adc_jext_trg8	TIM1_TRGO2	オンチップ･タイマからの内部信号	01000
adc_jext_trg9	TIM8_TRGO	オンチップ･タイマからの内部信号	01001
adc_jext_trg10	TIM8_TRGO2	オンチップ･タイマからの内部信号	01010
adc_jext_trg11	TIM1_CC3	オンチップ･タイマからの内部信号	01011
adc_jext_trg12	TIM3_TRGO	オンチップ･タイマからの内部信号	01100
adc_jext_trg13	EXTI ライン 3	外部ピン	01101
adc_jext_trg14	TIM6_TRGO	オンチップ･タイマからの内部信号	01110
adc_jext_trg15	TIM15_TRGO	オンチップ･タイマからの内部信号	01111
adc_jext_trg16	TIM20_TRGO	オンチップ･タイマからの内部信号	10000
adc_jext_trg17	TIM20_TRGO2	オンチップ･タイマからの内部信号	10001

表 166. ADC3/4/5 - インジェクトチャネルの外部トリガ （続き）

名前	転送元	タイプ	JEXTSEL[4:0]
adc_jext_trg18	TIM20_CC2	オンチップ･タイマからの内部信号	10010
adc_jext_trg19	hrtim_adc_trg2	オンチップ･タイマからの内部信号	10011
adc_jext_trg20	hrtim_adc_trg4	オンチップ･タイマからの内部信号	10100
adc_jext_trg21	hrtim_adc_trg5	オンチップ･タイマからの内部信号	10101
adc_jext_trg22	hrtim_adc_trg6	オンチップ･タイマからの内部信号	10110
adc_jext_trg23	hrtim_adc_trg7	オンチップ･タイマからの内部信号	10111
adc_jext_trg24	hrtim_adc_trg8	オンチップ･タイマからの内部信号	11000
adc_jext_trg25	hrtim_adc_trg9	オンチップ･タイマからの内部信号	11001
adc_jext_trg26	hrtim_adc_trg10	オンチップ･タイマからの内部信号	11010
adc_jext_trg27	hrtim_adc_trg1	オンチップ･タイマからの内部信号	11011
adc_jext_trg28	hrtim_adc_trg3	オンチップ･タイマからの内部信号	11100
adc_jext_trg29	LPTIMOUT	オンチップタイマからの内部信号	11101
adc_jext_trg30	TIM7_TRGO	オンチップ･タイマからの内部信号	11110
adc_jext_trg31	予約済み	-	11111

21.4.19 インジェクトチャネルの管理

トリガ･インジェクション･モード

トリガ･インジェクションを使用するには、ADC_CFGR レジスタの JAUTO ビットをクリアしておく必要があります。

1. 外部トリガによって、または、ADC_CR レジスタの ADSTART ビットをセットすることによって、レギュラチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合、または ADC_CR レジスタの JADSTART ビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンススイッチが起動されます（すべてのインジェクトチャネルが一度変換されます）。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。
4. インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。図 98 に対応するタイミング図を示します。

注： **トリガインジェクションを使用する場合は、トリガイベントの間隔がインジェクションシーケンスより長くなるようにしなければなりません。たとえば、シーケンスの長さが 30 ADC クロックサイクルの場合（2.5 クロック周期のサンプリング時間を持つ 2 つの変換）、トリガの最小間隔は 31 ADC クロックサイクルでなければなりません。**

自動インジェクションモード

ADC_CFGR レジスタの JAUTO ビットがセットされている場合、レギュラグループチャネルの変換後、インジェクトグループチャネルが自動的に変換されます。これを使用して、ADC_SQRy レジスタおよび ADC_JSQR レジスタにプログラミングされた最大 20 回の変換シーケンスを実行することができます。

このモードでは、レギュラ変換を開始し、続いてインジェクト変換を開始するために ADC_CR レジスタの ADSTART ビットをセットする必要があります（JADSTART はクリアされたままでなければなりません）。ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方が中止されます（JADSTP ビットを使用することはできません）。

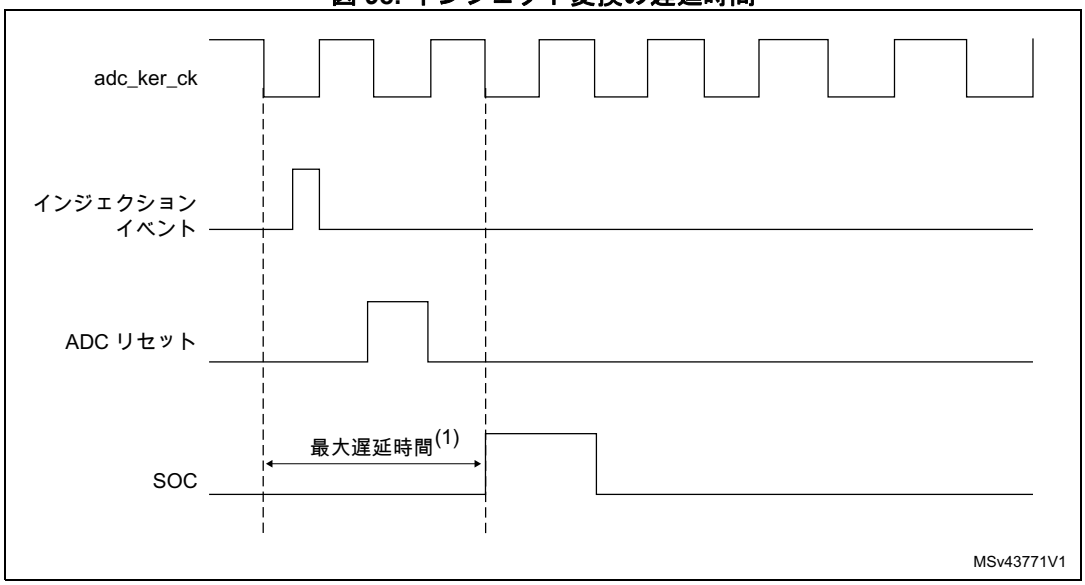
このモードでは、インジェクトチャネルの外部トリガは無効でなければなりません。

JAUTO ビットに加えて CONT ビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： **自動インジェクトモードと不連続モードを同時に使用することはできません。**

JAUTO モードで、レギュラシーケンスのデータをエクスポートする際に DMA を使用する場合は、サーキュラモードでプログラムする必要があります（DMA_CCRx レジスタの CIRC ビットをセット）。CIRC ビットがリセットされると（シングルショット･モード）、DMA 転送完了イベントによって JAUTO シーケンスが停止されます。

図 98. インジェクト変換の遅延時間



1. 最大遅延時間の値は、デバイスのデータシートの電気特性に記載されています。

21.4.20 不連続モード（DISCEN、DISCNUM、JDISCEN）

レギュラ･グループ･モード

このモードは、ADC_CFGR レジスタの DISCEN ビットをセットすることによって有効になります。

これを使用して、ADC_SQRy レジスタで選択された変換シーケンスの一部である短い n 回（n ≤ 8）の変換シーケンス（サブグループ）を実行できます。n の値は、ADC_CFGR レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADC_SQRy レジスタで選択された次の n 回の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_SQR1 レジスタの L[3:0] ビットによって定義されます。

例：

- DISCEN = 1、n = 3、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。

- 2 番目のトリガ：変換されるチャネルは 6、7、8 です（各変換で EOC イベントが生成されます）。
- 3 番目のトリガ：変換されるチャネルは 9、10、11 です（各変換で EOC イベントが生成されます）。EOS イベントはチャネル 11 の変換後に生成されます。
- 4 番目のトリガ：変換されるチャネルは 1、2、3 です（各変換で EOC イベントが生成されます）。
- .....

- DISCEN = 0、変換されるチャネル = 1、2、3、6、7、8、9、10、11
  - 1 番目のトリガ：シーケンス全体、すなわち、チャネル 1、次いで 2、3、6、7、8、9、10、および 11 が変換されます。各変換後に EOC イベントが生成され、最後の変換後には EOS イベントも生成されます。
  - 後続のすべてのトリガ･イベントによって、シーケンス全体が再起動されます。

注： **一部のマイクロコントローラでは、上記の例に示したチャネル番号の一部を使用できない場合があります。**

レギュラ･グループが不連続モードで変換されるときには、ロールオーバーは発生しません（シーケンスの最後のサブグループの変換が n 回未満になる場合があります）。

すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。上の例では、4 番目のトリガで最初のサブグループのチャネル 1、2、および 3 が再変換されます。

不連続モードと連続モードの両方を有効にすることはできません。この場合（DISCEN = 1、CONT = 1 の場合）、A/D コンバータは連続モードが無効化されたかのように動作します。

インジェクト･グループ･モード

このモードは、ADC_CFGR レジスタの JDISCEN ビットをセットすることによって有効になります。ADC_JSQR レジスタで選択されたシーケンスを、外部インジェクトトリガイベント後に 1 チャネルずつ変換します。これは、「n」が 1 に固定されたレギュラチャネルの不連続モードに相当します。

外部トリガが発生すると、ADC_JSQR レジスタで選択された次のチャネル変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_JSQR レジスタの JL[1:0] ビットによって定義されます。

例：

- JDISCEN = 1、変換されるチャネル = 1、2、3
  - 1 番目のトリガ：チャネル 1 が変換されます（JEOC イベントが生成されます）。
  - 2 番目のトリガ：チャネル 2 が変換されます（JEOC イベントが生成されます）。
  - 3 番目のトリガ: チャネル 3 が変換され、JEOC イベントと JEOS イベントが生成されます。
  - .....

注： **一部のマイクロコントローラでは、上記の例に示したチャネル番号の一部を使用できない場合があります。**

すべてのインジェクト･チャネルが変換されると、次のトリガでは、最初のインジェクト･チャネルの変換が開始されます。上の例では、4 番目のトリガで最初のインジェクトチャネル 1 が再変換されます。

自動インジェクトモードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

21.4.21 インジェクト変換のコンテキストのキュー

コンテキストのキューは、次のインジェクト変換シーケンスに最大 2 つのコンテキストを予測して実装されます。この機能を有効にするには、ADC_CFGR レジスタの JQDIS ビットをリセットする必要があります。コンテキストキューが有効になっている場合、ハードウェアトリガによる変換のみ可能です。

このコンテキストの構成は次のとおりです。

- インジェクトトリガの設定（ADC_JSQR レジスタの JEXTEN[1:0] ビット および JEXTSEL ビット）
- インジェクトシーケンスの定義（ADC_JSQR レジスタの JSQx[4:0] ビット および JL[1:0] ビット）

コンテキストのすべてのパラメータは 1 つの ADC_JSQR レジスタ 内に定義され、このレジスタには最大 2 セットのパラメータのバッファ化を可能にする 2 つのバッファのキューが実装されます。

- JSQR レジスタは、インジェクト変換の実行中など、いつでも書き込むことができます。
- JSQR レジスタに書き込まれる各データは、コンテキストのキューに格納されます。
- 最初に、キューは空であり、JSQR レジスタへの最初の書込みアクセスによってコンテキストがすぐに変更され、ADC ではインジェクトトリガを受信する準備ができます。
- 一度インジェクトシーケンスが完了すると、キューは消費され、キューに格納された次の JSQR パラメータに従ってコンテキストが変化します。この新しいコンテキストは、次のインジェクト変換シーケンスに適用されます。
- キューのオーバーフローは、キューがフル状態のときにJSQR レジスタに書き込むと発生します。このオーバーフローは、フラグ JQOVF のアサーションによって示されます。オーバーフローが発生すると、オーバーフローを作成した JSQR レジスタの書込みアクセスは無視され、コンテキストのキューは変更されません。JQOVFIE ビットがセットされている場合は、割込みを生成することができます。
- キューが空になると、ADC_CFGR レジスタの制御ビット JQM の値に応じて、2 つの挙動が考えられます。
  - JQM = 0 の場合、キューは A/D コンバータを有効化した直後は空ですが、その後の実行動作中に空になることはありません。キューは常に最後のアクティブ･コンテキストを維持し、その後の有効なインジェクト･シーケンスの開始を最後のアクティブ･コンテキストに応じて処理します。
  - JQM = 1 の場合、キューはインジェクト･シーケンスが終了した後、またはキューを一掃した後に空になる場合があります。このとき、キュー内に残されたコンテキストはなく、ハードウェア･トリガは無効になります。そのため、ソフトウェアが新しいインジェクトコンテキストを JSQR レジスタに再度書き込むまで、さらなるハードウェアインジェクトトリガは無視されます。
- JSQR レジスタを読み出すと、その時点でアクティブな現在の JSQR コンテキストが返されます。JSQR コンテキストが空である場合、JSQR は 0x0000 として読み出されます。
- キューは、JADSTP = 1 に設定してインジェクト変換を停止した場合、または ADDIS = 1 に設定して A/D コンバータを無効化した場合、一掃されます。
  - JQM = 0 の場合、キューには最後のアクティブ･コンテキストが保持されます。
  - JQM = 1 の場合、キューは空になり、トリガは無視されます。

注： **不連続モードに設定された場合（ビット JDISCEN = 1）、コンテキストを変更し、キューを消費するのはインジェクト･シーケンスの最後のトリガのみです。1 番目のトリガではキューを消費するだけですが、次の不連続モードの例に示すように、その他のトリガは引き続き有効なトリガとなります（両方のコンテキストにおいて長さは 3）。**

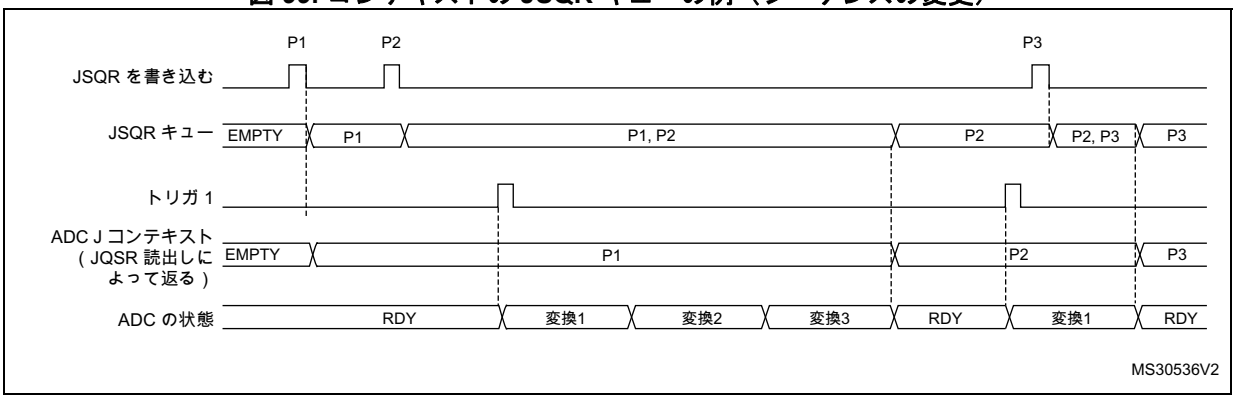
- 1 番目のトリガ、不連続。シーケンス 1：コンテキスト 1 を消費、1 番目の変換を実行
- 2 番目のトリガ、不連続。シーケンス 1：2 番目の変換
- 3 番目のトリガ、不連続。シーケンス 1：3 番目の変換

- 4 番目のトリガ、不連続。シーケンス 2：コンテキスト 2 を消費、1 番目の変換を実行
- 5 番目のトリガ、不連続。シーケンス 2：2 番目の変換
- 6 番目のトリガ、不連続。シーケンス 2：3 番目の変換

トリガまたはシーケンスのコンテキストを変更する場合の挙動

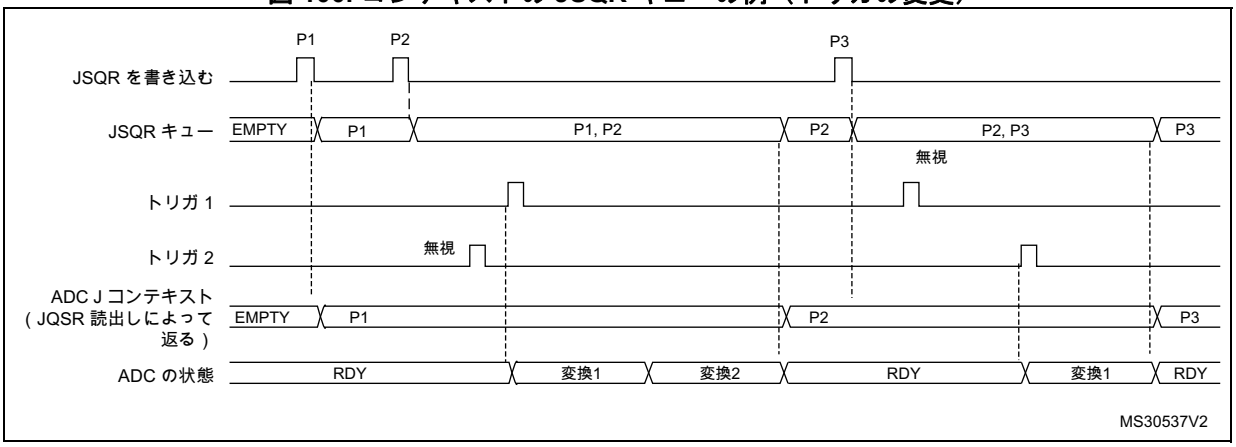
図 99 と図 100 に、シーケンスまたはトリガを変更する際のコンテキストのキューの挙動を示します。

図 99. コンテキストの JSQR キューの例（シーケンスの変更）



1. パラメータ：
   P1：3 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

図 100. コンテキストの JSQR キューの例（トリガの変更）

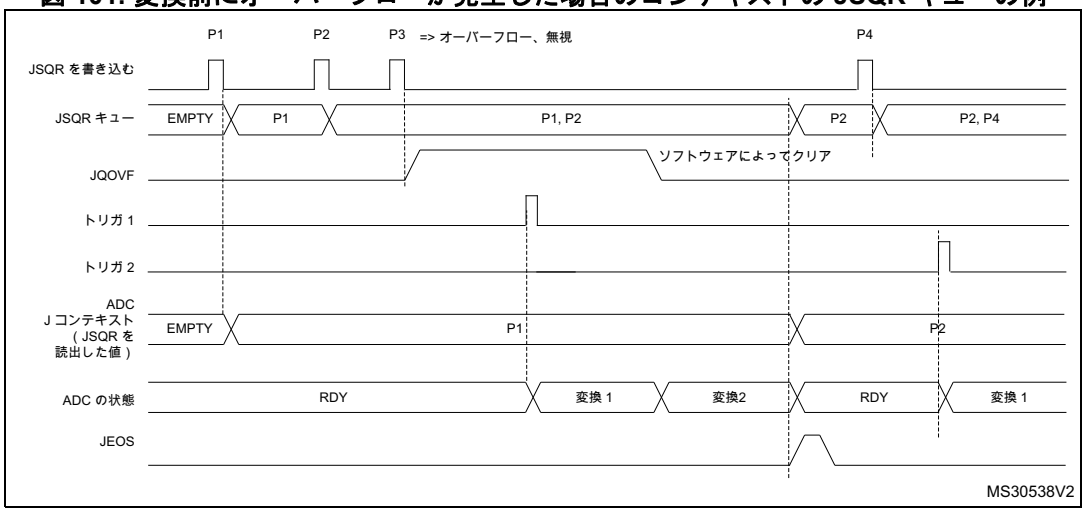


1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：4 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：キューのオーバーフローが発生した場合の挙動

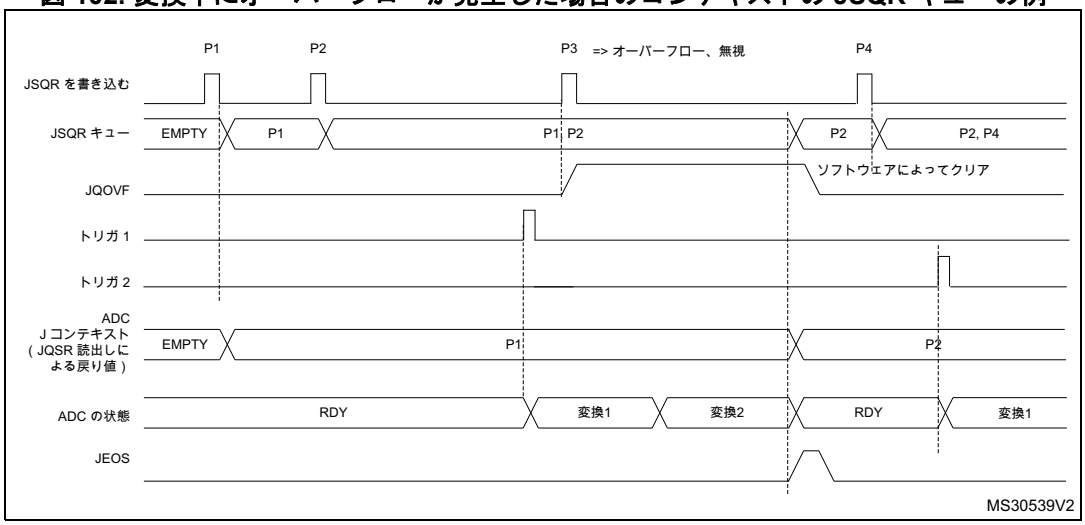
図 101 と図 102 に、変換前または変換中にオーバーフローが発生した場合のコンテキストのキューの挙動を示します。

図 101. 変換前にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

図 102. 変換中にオーバーフローが発生した場合のコンテキストの JSQR キューの例



1. パラメータ：
   P1：2 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 2
   P3：3 つの変換のシーケンス、ハードウェアトリガ 1
   P4：4 つの変換のシーケンス、ハードウェアトリガ 1

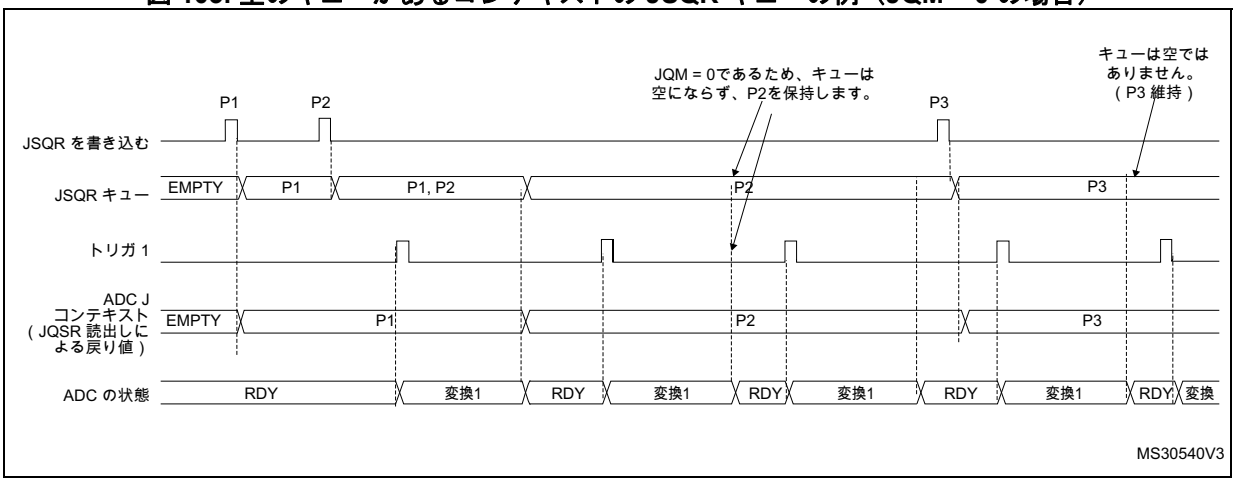
次のようにキューのオーバーフローを管理することを推奨します。

- JSQR レジスタへの各 P コンテキストの書込み後、JQOVFフラグ は書込みが無視されたかどうかを示します（割込みを生成することができます）。
- 一度前のコンテキスト P2 のフラグ JEOS がセットされた場合のみ、3 番目のコンテキスト（P3）を書き込んでキューのオーバーフローを避けます。これにより、前のコンテキストが消費され、キューがフル状態ではないことを確認できます。

コンテキストのキュー：キューが空になった場合の挙動

図 103 と図 104 に、JQM = 0 または 1 の場合にキューが空になる際のコンテキストのキューの挙動を示します。

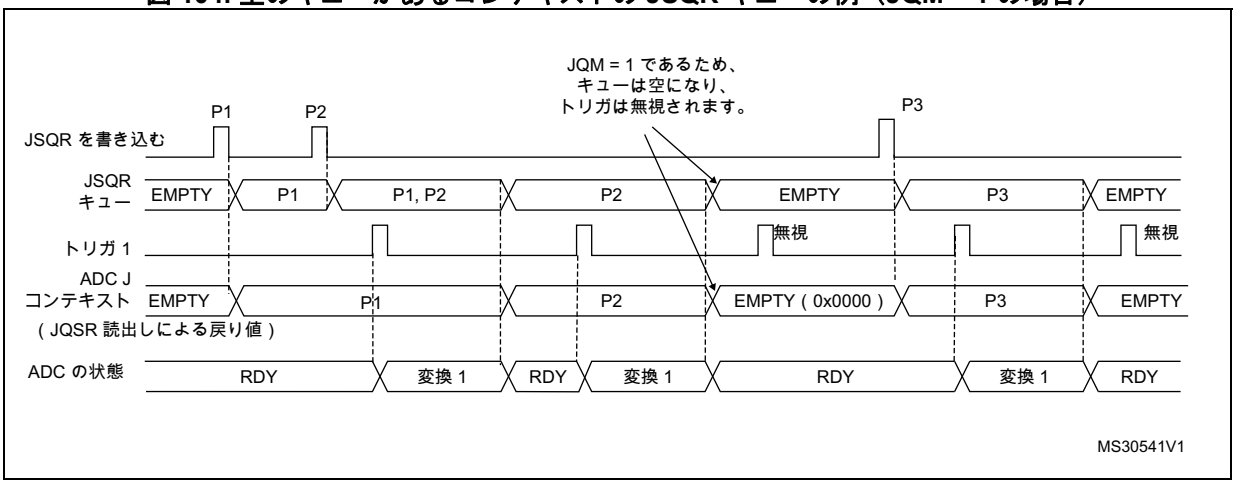
図 103. 空のキューがあるコンテキストの JSQR キューの例（JQM = 0 の場合）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

注： **P3 を書き込む際、コンテキストはすぐに変更されます。ただし、内部での再同期のため、遅延が発生します。また、P3 を書き込んだ後または前にトリガが発生すると、コンテキスト P2 を考慮して変換が発生する可能性があります。この状況を避けるために、ユーザはすぐに適用される新しいコンテキストを書き込む際に ADC トリガが発生していないことを確認する必要があります。**

図 104. 空のキューがあるコンテキストの JSQR キューの例（JQM = 1 の場合）

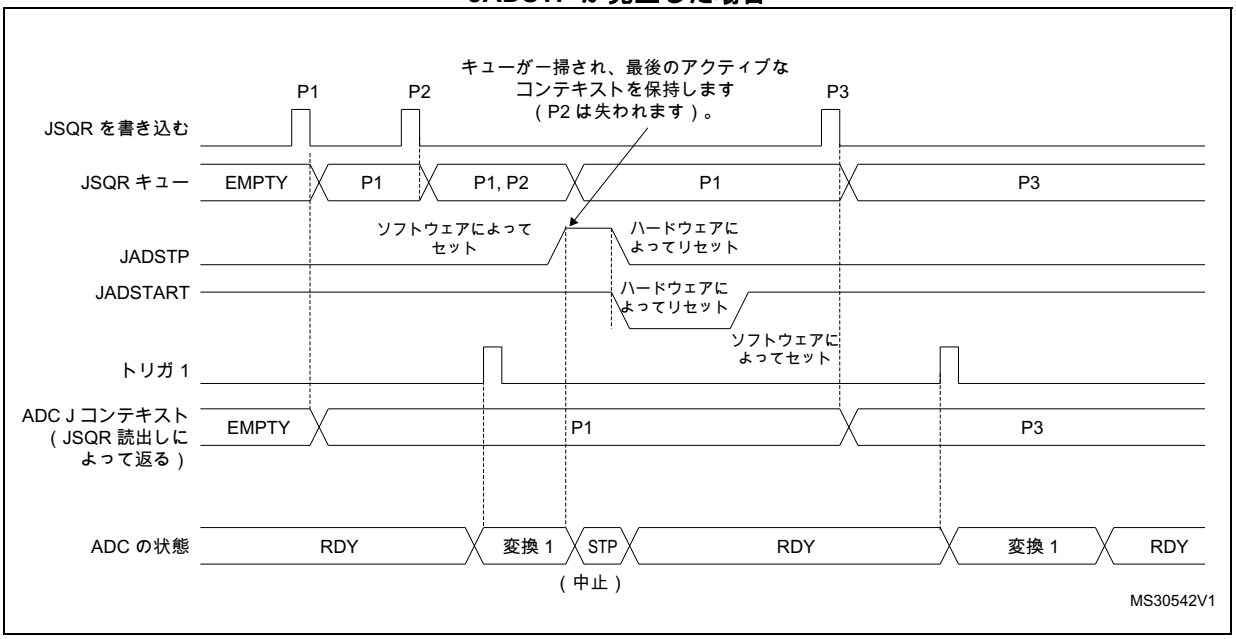


1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキューのFlash

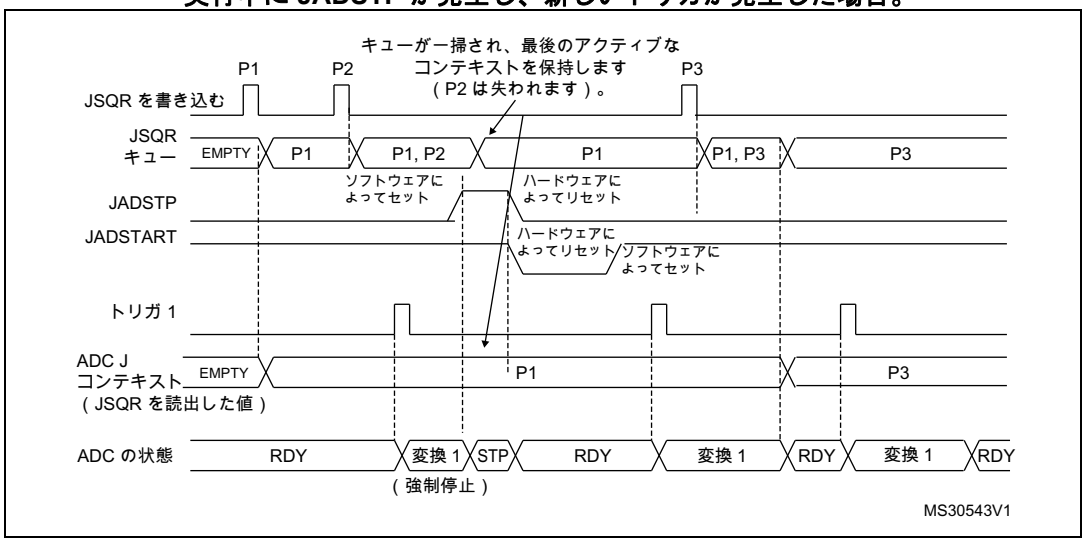
次の図に、キューがFlashされた場合のさまざまな状況におけるコンテキストのキューの挙動を示します。

図 105. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に JADSTP が発生した場合



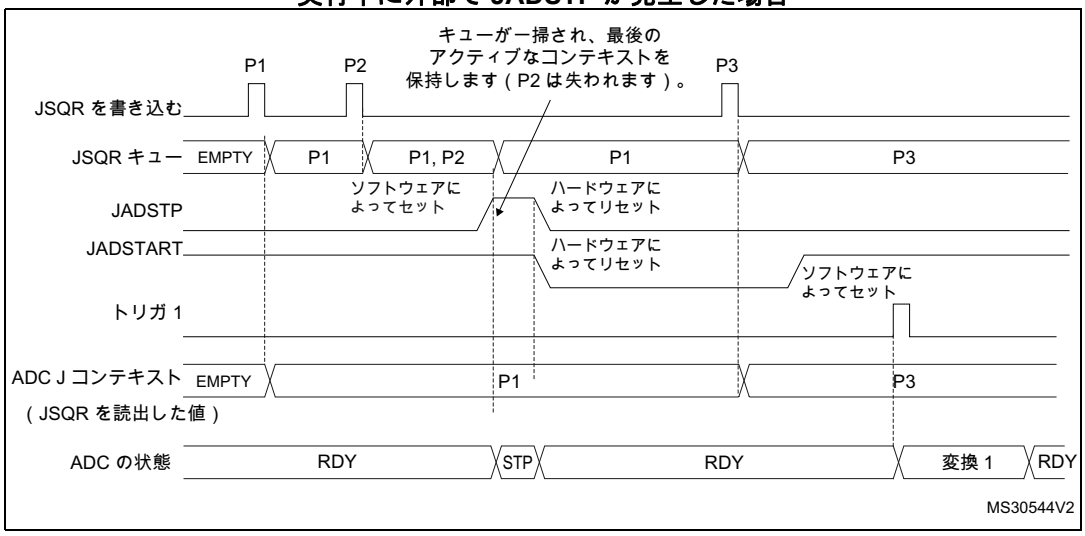
1. パラメータ：
P1：1 つの変換のシーケンス、ハードウェアトリガ 1
P2：1 つの変換のシーケンス、ハードウェアトリガ 1
P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 106. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に JADSTP が発生し、新しいトリガが発生した場合。



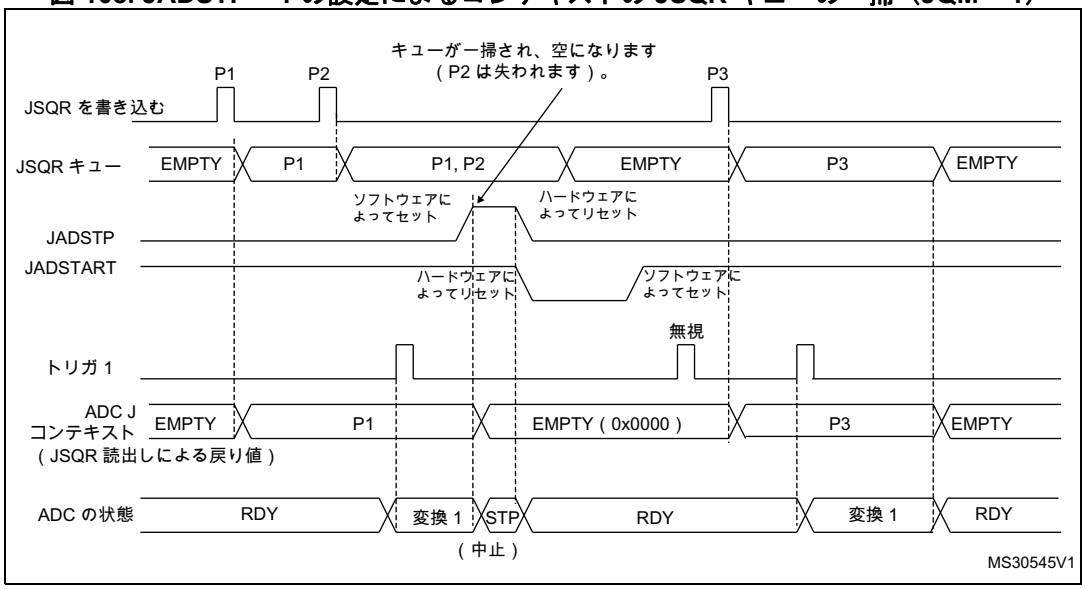
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 107. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）変換の実行中に外部で JADSTP が発生した場合



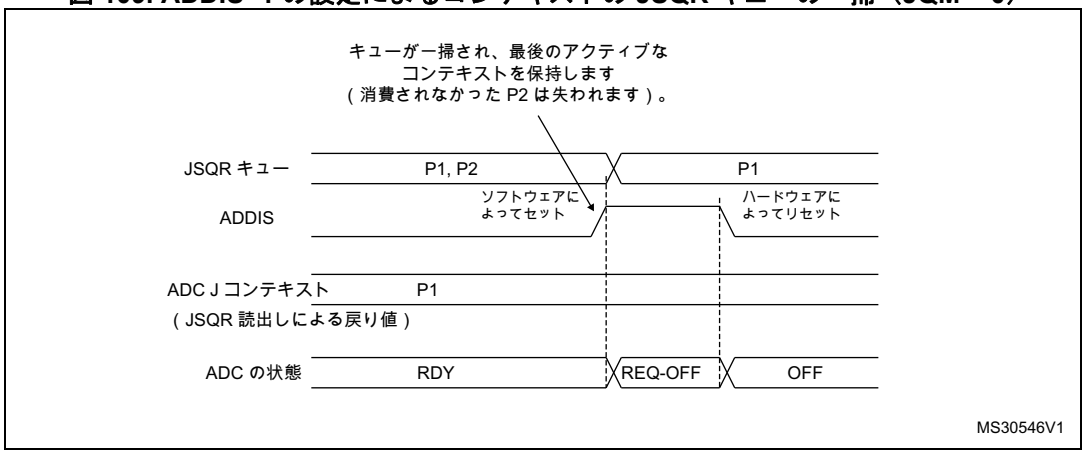
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 108. JADSTP = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 1）



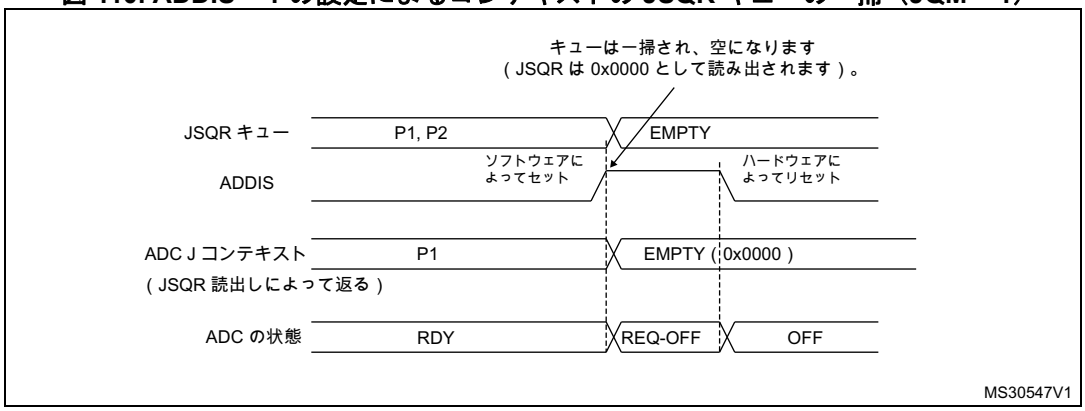
1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 109. ADDIS=1 の設定によるコンテキストの JSQR キューの一掃（JQM = 0）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

図 110. ADDIS = 1 の設定によるコンテキストの JSQR キューの一掃（JQM = 1）



1. パラメータ：
   P1：1 つの変換のシーケンス、ハードウェアトリガ 1
   P2：1 つの変換のシーケンス、ハードウェアトリガ 1
   P3：1 つの変換のシーケンス、ハードウェアトリガ 1

コンテキストのキュー：空のキューでの ADC 開始

最初のコンテキストが ADC 初期化の時点で不明である場合、空のキューで ADC 操作を開始するには次の手順に従う必要があります。この手順は、JQM ビットがリセットされているときのみ適用可能です。

5. 00 ではない JEXTEN[1:0] でダミーの JSQR を書き込みます（00 を書き込むとソフトウェア変換がトリガされます）。
6. JADSTART をセットします。
7. JADSTP をセットします。
8. JADSTART がリセットされるまで待ちます。
9. JADSTART をセットします。

キューの無効化

ADC_CFGR レジスタのビット JQDIS = 1 に設定することで、キューを無効化できます。

21.4.22 プログラム可能な分解能（RES）- 高速変換モード

A/D コンバータの分解能を下げることによって、高速変換を行うことができます。

分解能は、制御ビット RES[1:0] をプログラムすることによって、12、10、8、または 6 ビットに設定できます。図 115、図 116、図 117、および図 118 に、分解能とデータの整列に対する変換結果のフォーマットを示します。

分解能を下げることによって変換時間を高速にでき、高いデータ精度を必要としないアプリケーションに有効です。表 167 のように、これによって逐次比較ステップの間に変換時間は短縮されます。

表 167. T_{SAR} タイミングは分解能に依存

RES（ビット）	T_{SAR}（ADC クロックサイクル）	T_{SAR}（ns）（F_{ADC}=30 MHz 時）	T_{CONV}（ADC クロックサイクル）（サンプリング時間 = 2.5 ADC クロックサイクル）	T_{CONV} (ns)（F_{ADC}= 30 MHz 時）
12	12.5 ADC クロックサイクル	416.67 ns	15 ADC クロックサイクル	500.0 ns
10	10.5 ADC クロックサイクル	350.0 ns	13 ADC クロックサイクル	433.33 ns
8	8.5 ADC クロックサイクル	203.33 ns	11 ADC クロックサイクル	366.67 ns
6	6.5 ADC クロックサイクル	216.67 ns	9 ADC クロックサイクル	300.0 ns

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

ADC は各レギュラ変換の終了（EOC）イベントと各インジェクト変換の終了（JEOC）イベントをアプリケーションに通知します。

ADC は、新しいレギュラ変換データが ADC_DR レジスタで使用可能になるとすぐに EOC フラグをセットします。EOCIE ビットがセットされている場合は、割込みを生成することができます。EOC フラグは、ソフトウェアによって 1 を書き込むことによって、または ADC_DR を読み出すことによってクリアされます。

ADC は、新しいインジェクト変換データが ADC_JDRy レジスタの 1 つで使用可能になるとすぐに JEOC フラグをセットします。JEOCIE ビットがセットされている場合は、割込みを生成することができます。JEOC フラグは、ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

さらに、ADC は、ステータスビット EOSMP をセットすることによって、サンプリングフェーズの終了を示します（レギュラ変換のみ）。EOSMP フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。EOSMPIE ビットがセットされている場合は、割込みを生成することができます。

21.4.24 変換シーケンスの終了（EOS、JEOS）

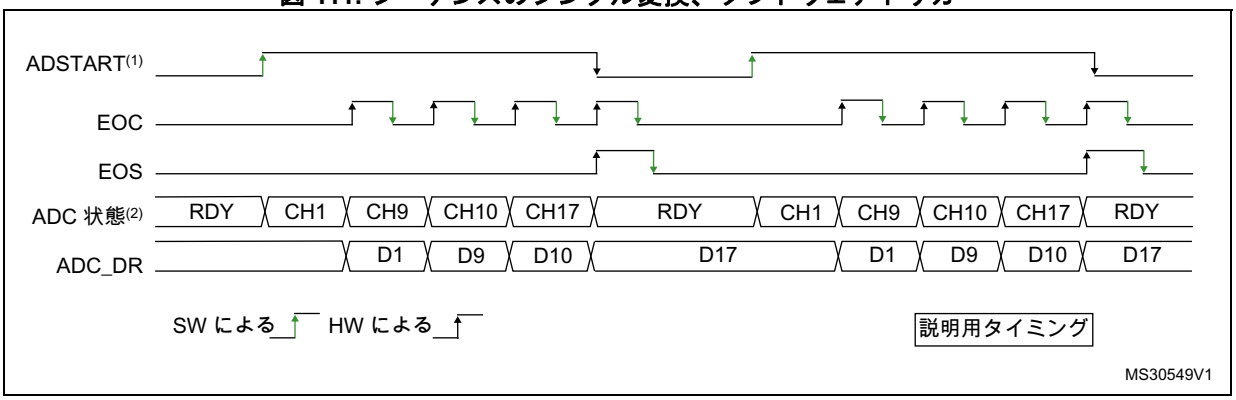
ADC は各レギュラシーケンスの終了（EOS）と各インジェクトシーケンスの終了（JEOS）イベントをアプリケーションに通知します。

ADC は、レギュラ変換シーケンスの最後のデータが ADC_DR レジスタで使用可能になるとすぐに、EOS フラグをセットします。EOSIE ビットがセットされている場合は、割込みを生成することができます。EOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC は、インジェクト変換シーケンスの最後のデータが完了するとすぐに、JEOS フラグをセットします。JEOSIE ビットがセットされている場合は、割込みを生成することができます。JEOS フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

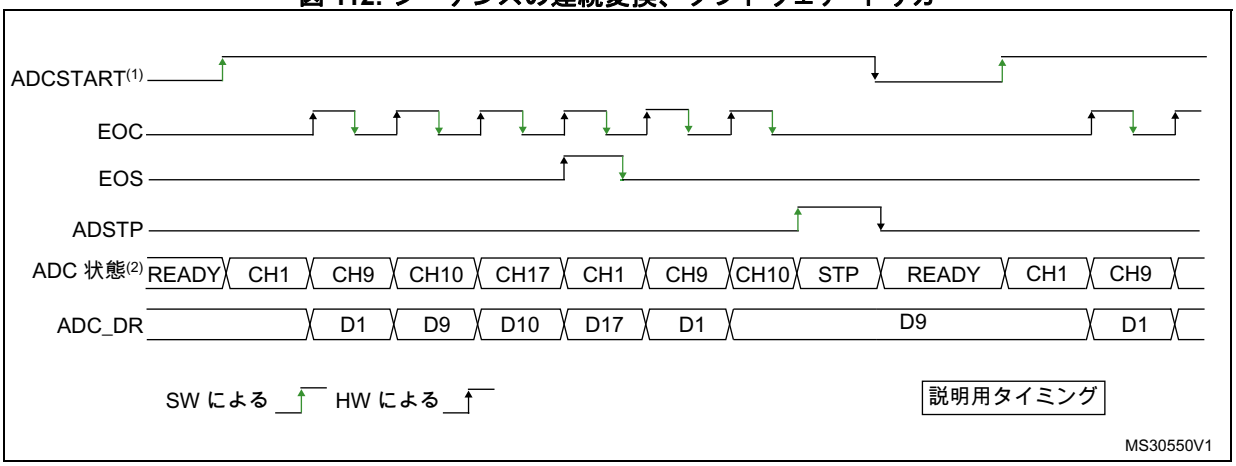
21.4.25 タイミング図の例（シングル / 連続モード、ハードウェア / ソフトウェアトリガ）

図 111. シーケンスのシングル変換、ソフトウェアトリガ



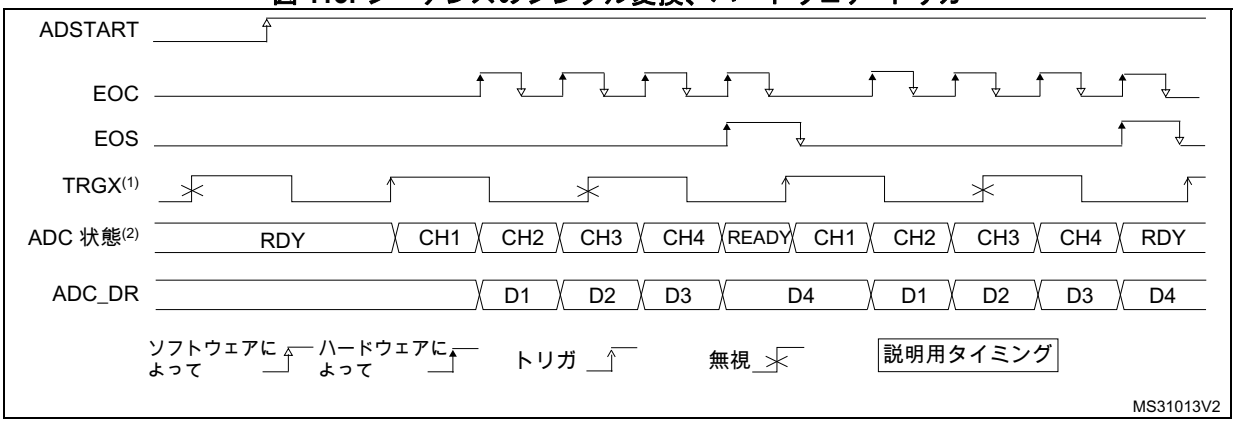
1. EXTEN[1:0] = 00、CONT = 0
2. 選択されるチャネル = 1、9、10、17；AUTDLY = 0

図 112. シーケンスの連続変換、ソフトウェア･トリガ



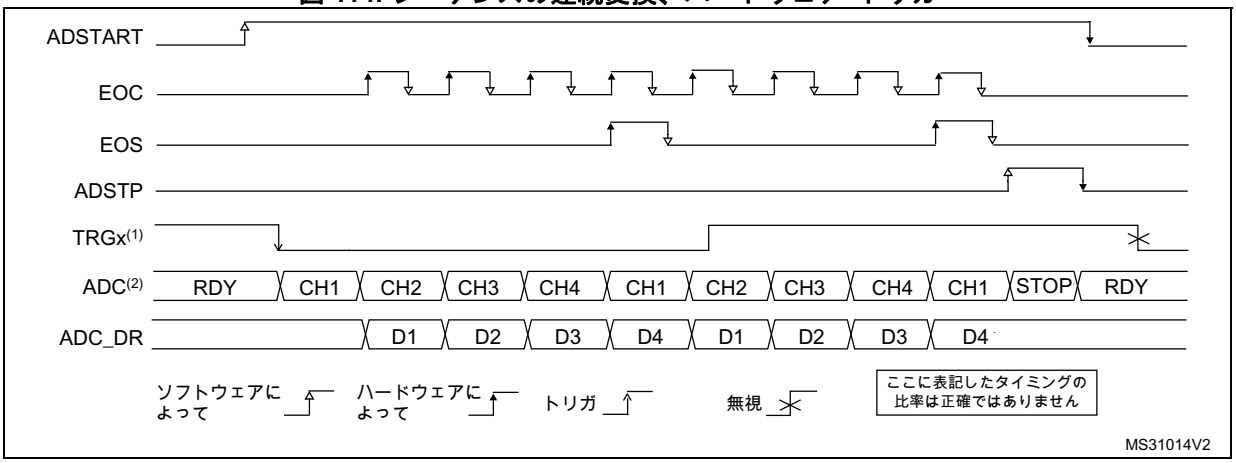
1. EXTEN[1:0] = 00、CONT = 1
2. 選択されるチャネル = 1、9、10、17；AUTDLY = 0

図 113. シーケンスのシングル変換、ハードウェア･トリガ



1. TRGx（オーバー周波数）がトリガ･ソースとして選択されます（EXTEN[1:0] = 01、CONT = 0）。
2. 選択されるチャネル = 1、2、3、4；AUTDLY = 0

図 114. シーケンスの連続変換、ハードウェア･トリガ



1. TRGx がトリガ･ソースとして選択されます（EXTEN[1:0] = 10、CONT = 1）。
2. 選択されるチャネル = 1、2、3、4；AUTDLY = 0

21.4.26 データ管理

データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）

データおよび配置

各レギュラ変換チャネルの終了時（EOC イベントの発生時）、変換データの結果は 16 ビット幅の ADC_DR データレジスタに格納されます。

各インジェクト変換チャネルの終了時（JEOC イベントの発生時）、変換データの結果は対応する 16 ビット幅の ADC_JDRy データレジスタに格納されます。

ADC_CFGR レジスタの ALIGN ビットは、変換後に格納されるデータの配置を選択します。データは、図 115、図 116、図 117、および図 118 に示すように、右詰めまたは左詰めに配置できます。

特殊なケース：左詰めの場合、分解能が 6 ビットに設定されている場合を除き、データはハーフワード単位に配置されます。この場合、図 117 および図 118 に示すように、データはバイト単位に配置されます。

注： **左詰めは、オーバーサンプリングモードでサポートされません。ROVSE ビットおよび／または JOVSE ビットがセットされている場合、ALIGN ビット値は無視され、ADC は右詰めデータのみ提供します。**

オフセット

オフセット y（y = 1、2、3、4）は、ADC_OFRy レジスタのビット OFFSETy_EN = 1 に設定することでチャネルに適用されます。オフセットを適用するチャネルは、ADC_OFRy レジスタの OFFSETy_CH[4:0] ビットにプログラムします。この場合、変換された値は OFFSETy[11:0] ビットに書き込まれたユーザ定義のオフセットだけ減算されます。結果は負の値になることがあり、そのため読出しデータは符号付きで、SEXT ビットは拡張符号値を表します。

注： **オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSE および／または JOVSE ビットがセットされている場合、ADC_OFRy レジスタの OFFSETy_EN ビットの内容は無視されます（リセットとみなされます）。**

表 170 に、アナログウォッチドッグ 1 に可能なすべての分解能での比較方法を示します。

表 168. オフセット計算対データ分解能

分解能（ビット RES[1:0]）	元の変換データとオフセット間での減算：		結果	コメント
	元の変換データ、左詰め	オフセット		
00：12 ビット	DATA[11:0]	OFFSET[11:0]	符号付き 12 ビットデータ	-
01：10 ビット	DATA[11:2],00	OFFSET[11:0]	符号付き 10 ビットデータ	ユーザは OFFSET[1:0] を 00 に設定する必要があります。
10：8 ビット	DATA[11:4],0000	OFFSET[11:0]	符号付き 8 ビットデータ	ユーザは OFFSET[3:0] を 0000 に設定する必要があります。
11：6 ビット	DATA[11:6],000000	OFFSET[11:0]	符号付き 6 ビットデータ	ユーザは OFFSET[5:0] を 000000 に設定する必要があります。

チャネル“i”に対応する ADC_DR（レギュラ･チャネル）または ADC_JDRy（インジェクト･チャネル、y = 1、2、3、4）からデータを読み出す場合：

- 対応するチャネルのオフセットの 1 つが有効（ビット OFFSETy_EN = 1）である場合、読出しデータは符号付きです。
- このチャネルの 4 つのオフセットに有効なものがない場合、読出しデータは符号なしです。

図 115、図 116、図 117、および図 118 に、符号付きおよび符号なしのデータの配置を示します。

図 115. 右詰め（オフセット無効、符号なしの値）

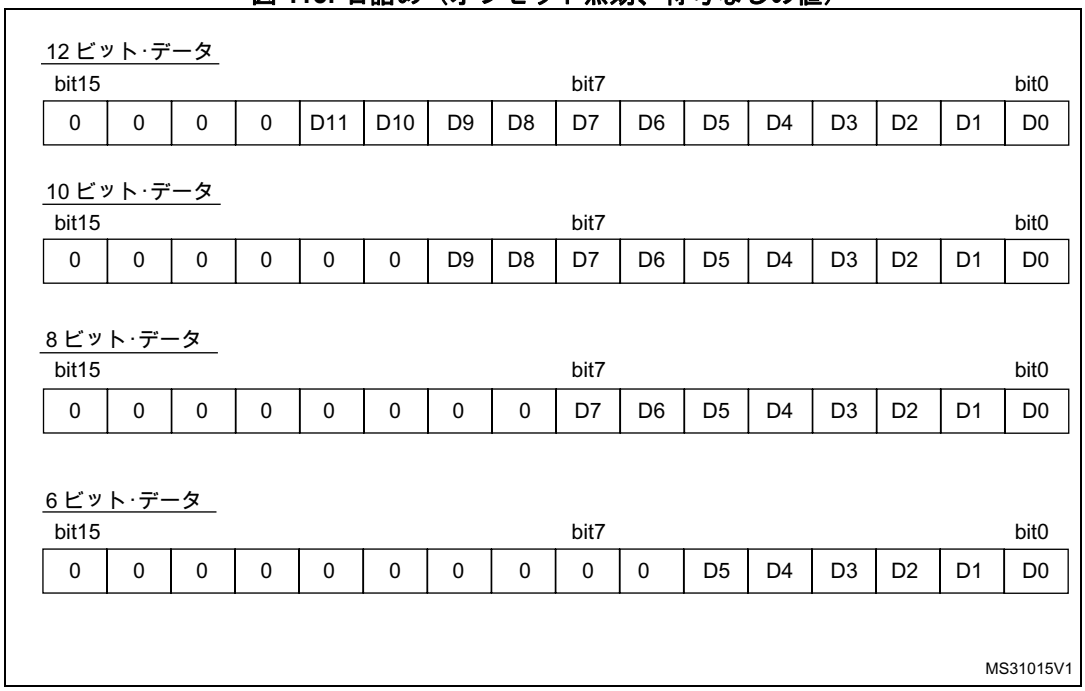


図 116. 右詰め（オフセット有効、符号付きの値）

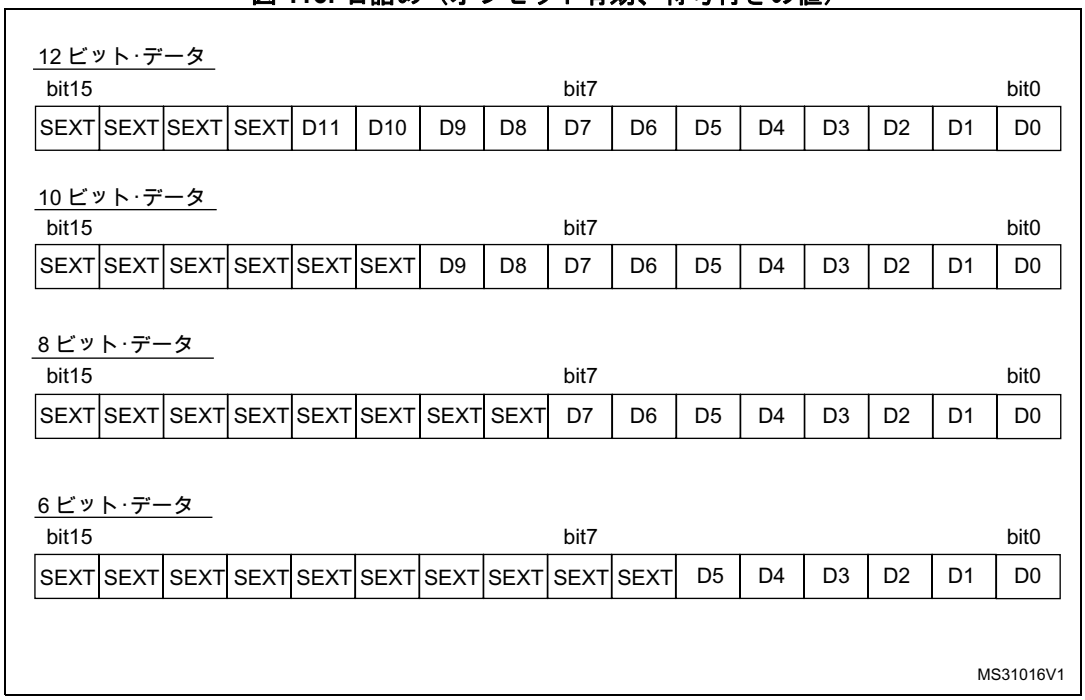


図 117. 左詰め（オフセット無効、符号なしの値）

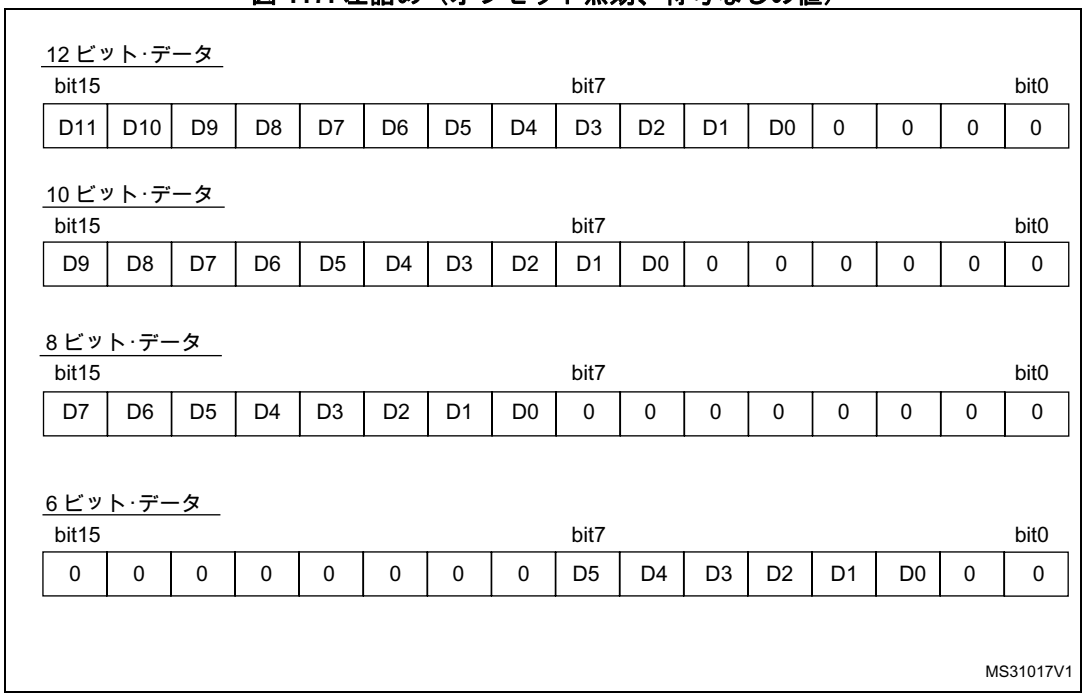
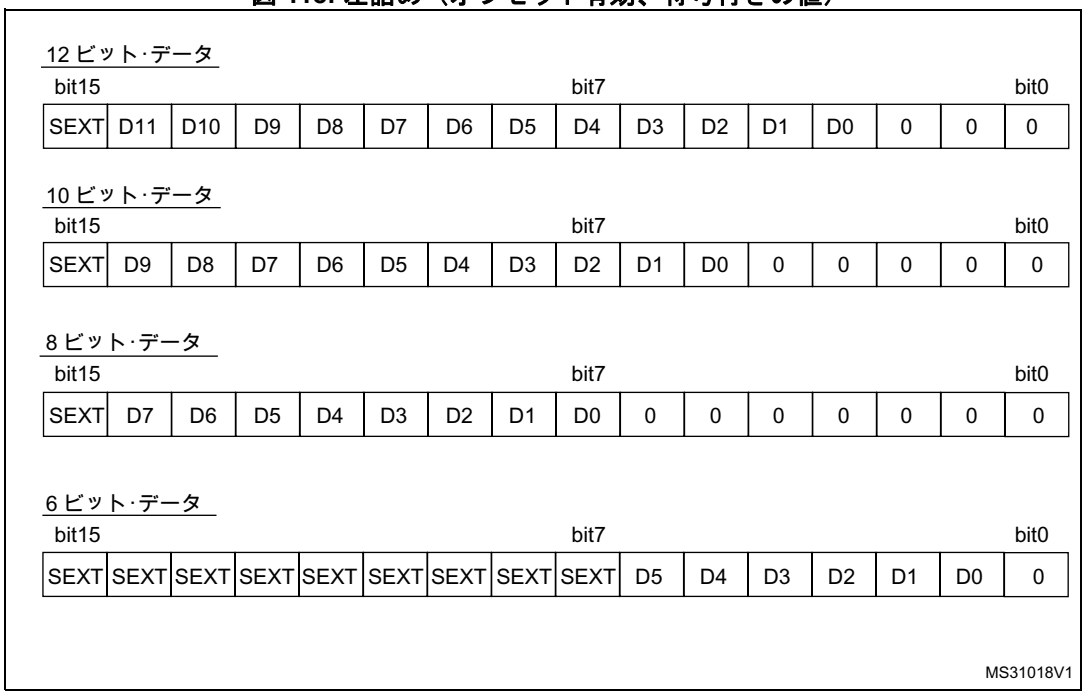


図 118. 左詰め（オフセット有効、符号付きの値）



ゲイン補正

ADC_CFGR2 レジスタで GCOMP ビットがセットされているとき、ゲイン補正はすべての変換データで有効です。各変換後、データは次の式で計算されます。

$$DATA = DATA(\text{adc result}) \times (GCOMPCOEFF) / 4096$$

GCOMPCOEFF は 0 から 16383 までプログラムできるため、実際のゲイン補正係数は 0 から 3.999756 までです。

結果のデータを RDATA または JDATAx レジスタに格納する前に、LSB−1 値が評価されて、データを丸め、エラーを最小化します。

ゲイン補正は、オーバーサンプリング にも効果的です。ゲイン補正がオーバーサンプリング モードで使用されるとき、電力消費を最小化するために、ゲイン計算は累積および右シフト操作後に行われます（ゲイン計算は、変換ごとではなく、一度だけ行われます）。

オフセット補正

オフセット操作時に ADC_OFRｙ レジスタの SATEN ビットがセットされているとき、データは符号なしです。すべてのオフセット･データは 0x000 で飽和します（12 ビット･モードの場合）。OFFSETPOS ビットをセットするとオフセットの方向が正になり、データは 0xFFF で飽和します（12 ビット･モードの場合）。8 ビット･モードのときには、上記のそれぞれで 0x00 と 0xFF で飽和します。

アナログ･ウォッチドッグ比較は、オフセットおよびゲイン補正後に、符号なしの値に対して行われます。正しいウォッチドッグ操作のためには、オフセット補正後のデータは符号なし形式でなければなりません（ADC_OFRy レジスタの SATEN ビットをセット）。

A/D コンバータのオーバーラン（OVR、OVRMOD）

オーバーラン･フラグ（OSR）は、新しい変換データが使用可能になる前に、（CPU または DMA によって）レギュラ変換データが読み出されなかったときに、バッファ･オーバーラン･イベントを通知します。

OVR フラグは、新しい変換が完了した時点で EOC フラグが 1 のままであった場合にセットされます。ビット OVRIE = 1 の場合、割込みを生成できます。

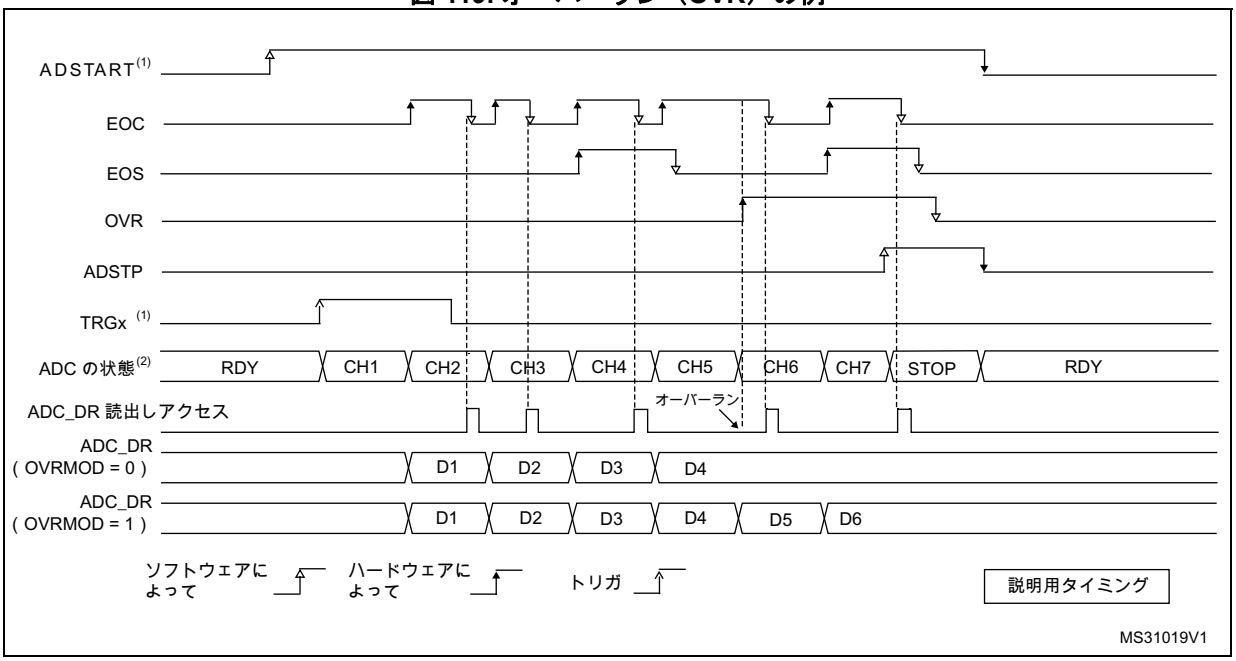
オーバーラン条件が発生しても、A/D コンバータは動作を継続します。ソフトウェアがビット ADSTP = 1 に設定することによってシーケンスの停止およびリセットを決めるまで、変換を続行できます。

OVR フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

制御ビット OVRMOD をプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD = 0：オーバーラン･イベントが発生しても、データ･レジスタは上書きされません。古いデータが保持され、新しい変換は破棄され、失われます。OVR が 1 のままの場合、その後の変換は行われますが、結果データは破棄されます。
- OVRMOD = 1：データ･レジスタは最後の変換結果で上書きされ、以前の未読データは失われます。OVR が 1 のままの場合、その後の変換は通常どおり動作し、ADC_DR レジスタは常に最新の変換データを含みます。

図 119. オーバーラン（OVR）の例



注： **インジェクト･チャネルでは、4 つのインジェクト･チャネルにそれぞれ専用のデータ･レジスタがあるため、オーバーランを検出する機能はありません。**

DMA を使用しない変換シーケンスの管理

変換が十分遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアは EOC フラグと関連の割込みを使用して、各データを処理する必要があります。変換が終了するごとに、EOC はセットされ、ADC_DR レジスタを読み出すことができます。オーバーランイベントをエラーとして管理するには、OVRMOD を 0 に設定する必要があります。

オーバーランなしでの DMA を使用しない変換の管理

毎回データの読出しをせずに ADC に 1 つまたは複数のチャネルを変換させると便利な場合があります（たとえば、アナログウォッチドッグがある場合）。この場合、OVRMOD ビットを 1 に設定する必要があり、ソフトウェアは OVR フラグを無視する必要があります。オーバーラン･イベントが発生しても A/D コンバータは変換を続行し、ADC_DR レジスタは常に最新の変換を含みます。

DMA を使用した変換の管理

変換されたチャネルの値は特定のデータレジスタに格納されるので、複数のチャネルの変換には DMA の使用が便利です。これによって、ADC_DR レジスタにすでに格納されているデータの損失を防ぐことができます。

DMA モードが有効なとき（シングル A/D コンバータ･モードまたはデュアル A/D コンバータ･モードで MDMA が 00 以外に設定されている場合に ADC_CFGR レジスタの DMAEN ビットがセットされている場合）、各チャネルの変換後、DMA リクエストが生成されます。これにより、変換されたデータを ADC_DR レジスタからソフトウェアで選択した場所へ転送できます。

これにもかかわらず、DMA が DMA 転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合（OVR = 1）、A/D コンバータは DMA リクエストの生成を停止し、新しい変換に対応するデータは DMA によって転送されません。これは、RAM に転送されるすべてのデータを有効とみなすことができることを意味します。

OVRMOD ビットの設定に応じて、データは保存または上書きされます（セクション ：A/D コンバータのオーバーラン（OVR、OVRMOD）を参照してください）。

DMA 転送リクエストは、ソフトウェアが OVR ビットをクリアするまでブロックされます。

アプリケーションの用途に応じて 2 つの DMA モードがあり、ADC_CFGR レジスタの DMACFG ビット（シングル ADC モード）で設定され、ADC_CCR レジスタの DMACFG ビット（デュアル ADC モード）で設定されます。

- DMA ワンショット･モード（DMACFG = 0）。
  このモードは、DMA を固定数のデータを転送するようにプログラムする場合に適しています。
- DMA サーキュラ･モード（DMACFG = 1）。
  このモードは、DMA をサーキュラ･モードでプログラムする場合に適しています。

DMA ワンショット･モード（DMACFG = 0）

このモードでは、A/D コンバータは新しい変換データが使用可能になるたびに DMA 転送リクエストを生成し、変換が再び開始された場合でも、DMA が最後の DMA 転送に達すると（DMA_EOT 割込みが発生すると）（DMA の段落を参照）、DMA リクエストの生成を停止します。

DMA 転送完了後（DMA コントローラで設定されたすべての転送後）：

- ADC データレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMA コントローラに対する新しいDMA リクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMA は停止します。

DMA サーキュラ･モード（DMACFG = 1）

このモードでは、DMA が最後の DMA 転送に達していても、A/D コンバータは新しい変換データがデータ･レジスタに置かれるたびに DMA 転送リクエストを生成します。これにより、DMA を連続的なアナログ入力データストリームを処理するようにサーキュラモードに設定できます。

21.4.27 動的低電力機能

自動遅延変換モード（AUTDLY）

ADC は、AUTDLY 設定ビットによって制御されている自動遅延変換モードを実装しています。自動遅延変換は、ソフトウェアを単純化する場合だけでなく、ADC オーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化する場合に便利です。

AUTDLY = 1 の場合、新しい変換は同じグループの前のデータがすべて処理された場合にのみ開始できます。

- レギュラ変換の場合：ADC_DR レジスタの読出し後、または EOC ビットのクリア後（図 120 を参照）
- インジェクト変換の場合：JEOS ビットのクリア後（図 121 を参照）

これは、データを読み出すシステムの速度に A/D コンバータの速度を自動的に適応させる方法です。

遅延は、各レギュラ変換後（DISCEN = 0 か 1 かにかかわらず）、およびインジェクト変換の各シーケンス後（JDISCEN = 0 か 1 かにかかわらず）に挿入されます。

注： **遅延は、インジェクト･シーケンスの各変換の間には挿入されません（最後の変換を除く）。**

変換時、この遅延中に発生した（同じ変換グループに対する）ハードウェアトリガイベントは無視されます。

注： **ソフトウェアトリガは、変換を再開するために、この遅延中に ADSTART ビットまたは JADSTART ビットをセットできるようにしている点で、ハードウェアトリガとは異なります。新しい変換を起動する前にデータを読み出すかどうかは、ソフトウェアに依存します。**

遅延は、異なるグループの変換（レギュラ変換に続いてインジェクト変換、またはその逆）の間には挿入されません。

- レギュラ変換の自動遅延中にインジェクトトリガが発生すると、すぐにインジェクト変換が開始されます（図 121 を参照）。
- インジェクトシーケンスが完了すると、新しいレギュラ変換を開始する前に、ADC は前のレギュラ変換の遅延を待ちます（終了していない場合）（図 123 を参照）。

自動インジェクト･モード（JAUTO = 1）では挙動が少し異なり、前のインジェクト変換シーケンスの自動遅延が終了した場合（JEOS がクリアされた場合）にのみ新しいレギュラ変換を開始できます。これは、新しいシーケンスを開始する前に、ソフトウェアが特定のシーケンスのデータをすべて読み出すことができるようにするためです（図 124 を参照）。

自動遅延モード（JAUTO = 1、CONT = 1、および AUTDLY = 1）と組み合わされた連続自動インジェクション･モードの変換を停止するには、次の手順に従います。

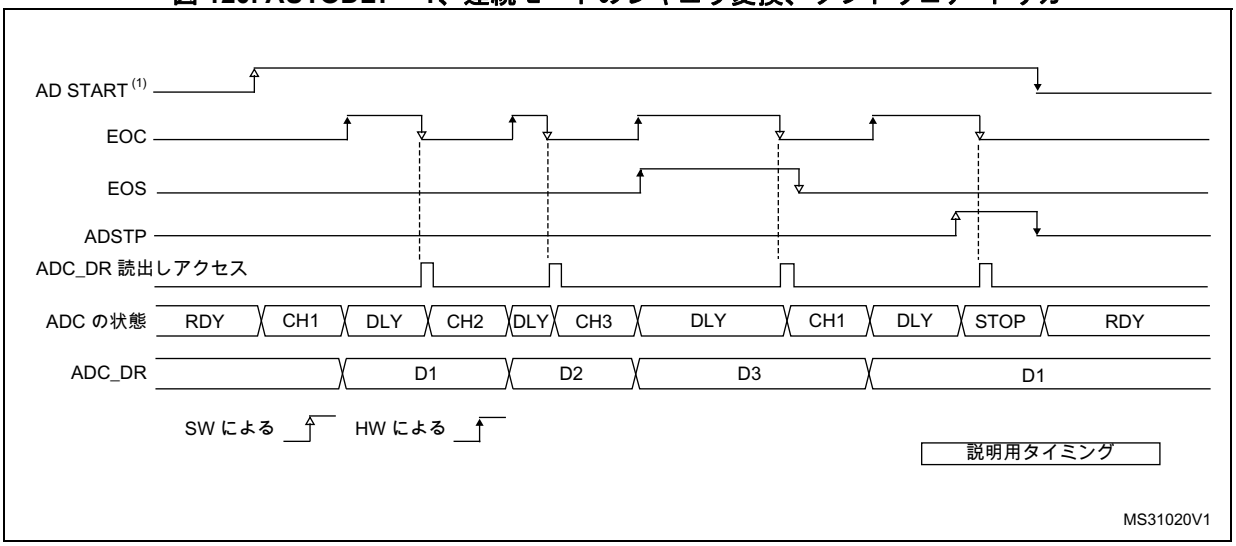
1. JEOS = 1 になるまで（リスタートされる変換がなくなるまで）待ちます。
2. JEOS をクリアします。
3. ADSTP をセットします。
4. レギュラ･データを読み出します。

この手順に従わない場合、ADSTP がセットされた後で JEOS がクリアされると、新しいレギュラシーケンスが再開されます。

AUTDLY モードでは、すでに実行中のレギュラシーケンス中、またはシーケンスの最後のレギュラ変換後の遅延中に発生した場合、ハードウェアレギュラトリガイベントは無視されます。ただし、この遅延後に発生した場合は、後続の遅延のインジェクトシーケンス中に発生した場合でも、保留とみなされます。その後、インジェクトシーケンスの遅延終了時に変換が開始されます。

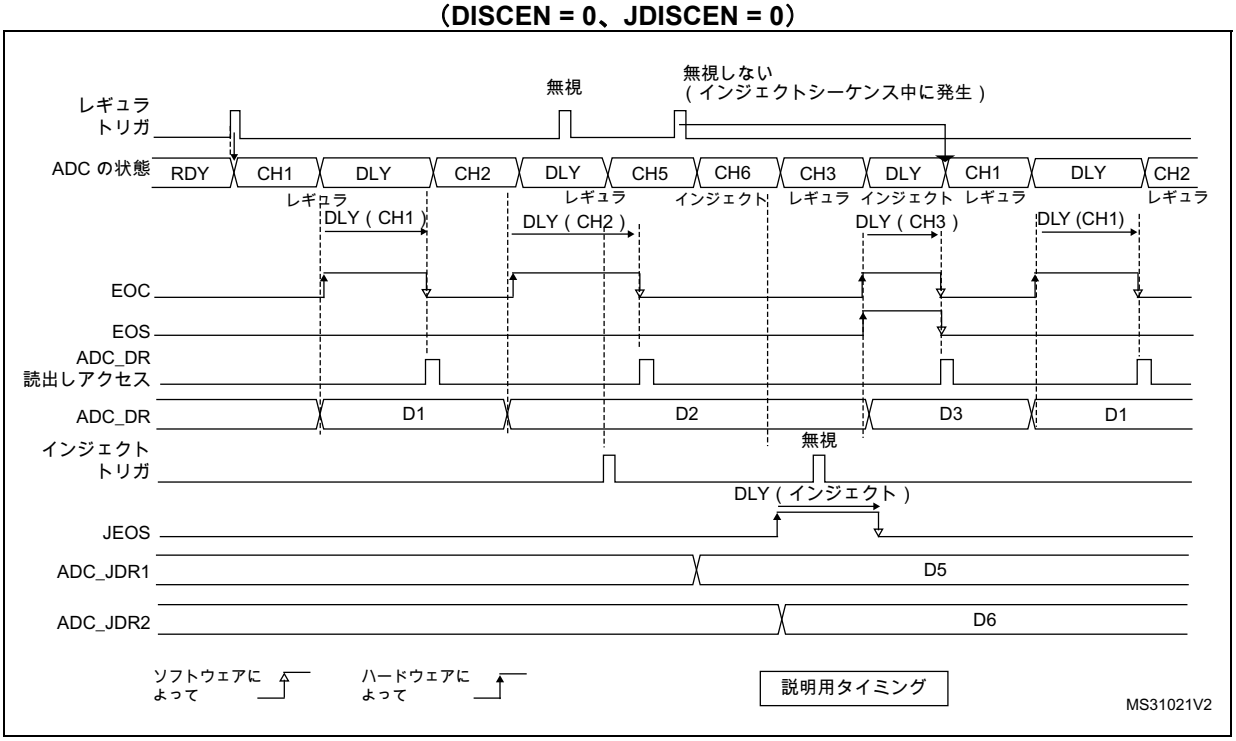
AUTDLY モードでは、すでに実行中のインジェクトシーケンス中、またはシーケンスの最後のインジェクト変換後の遅延中に発生した場合、ハードウェアインジェクトトリガイベントは無視されます。

図 120. AUTODLY = 1、連続モードのレギュラ変換、ソフトウェア･トリガ



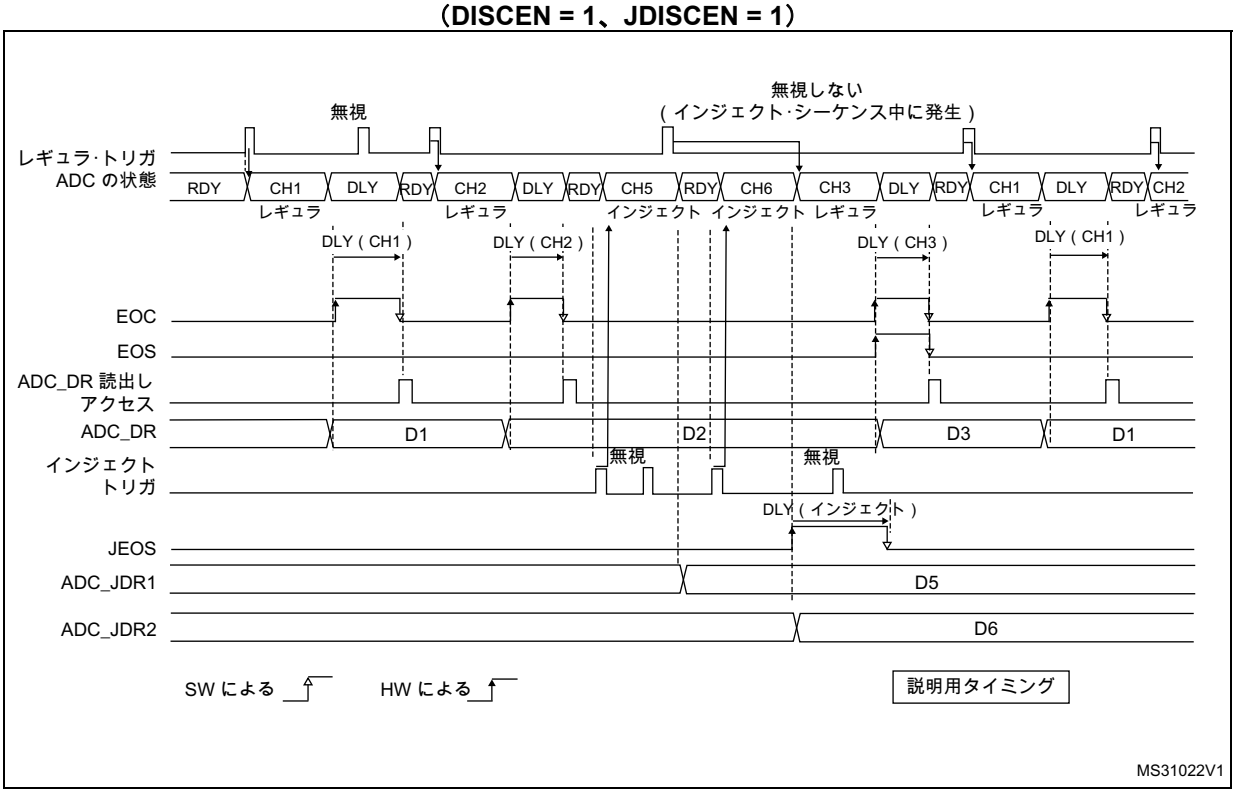
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 00（ソフトウェア･トリガ）、CONT = 1、チャネル = 1、2、3
3. インジェクト設定：無効

図 121. AUTODLY = 1、インジェクト変換によって中断されたレギュラ･ハードウェア変換（DISCEN = 0、JDISCEN = 0）



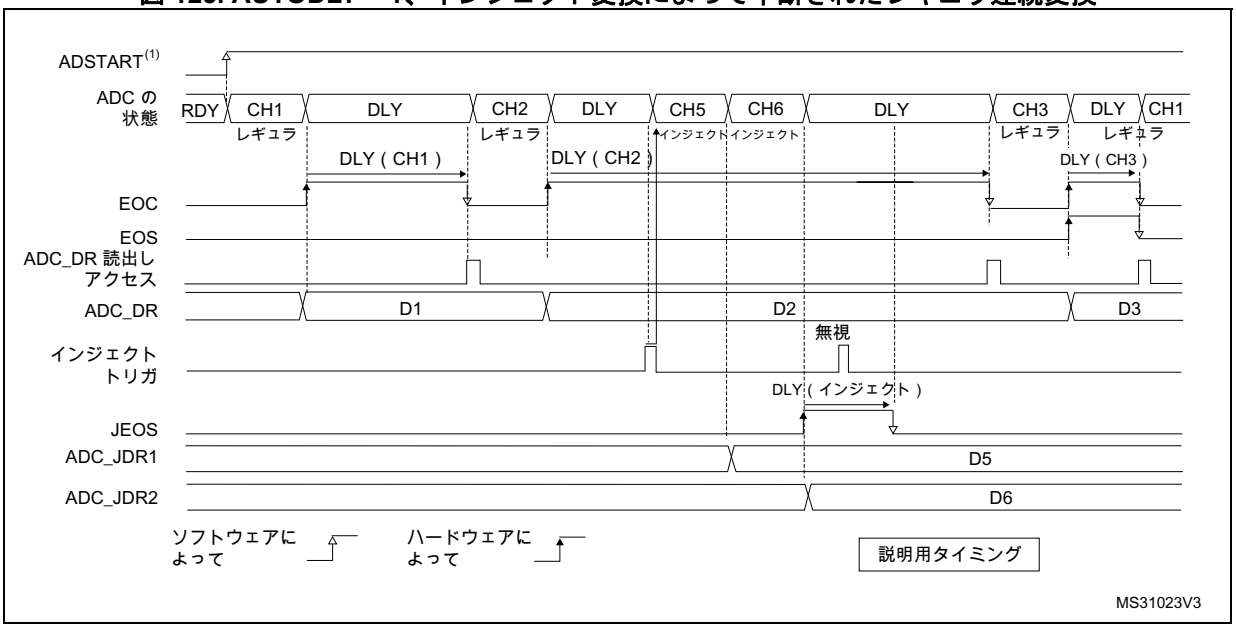
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 01（ハードウェア･トリガ）、CONT = 0、DISCEN = 0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN[1:0] = 01（ハードウェア･トリガ）、JDISCEN = 0、CHANNELS = 5、6

図 122. AUTODLY = 1、インジェクト変換によって中断されたレギュラ･ハードウェア変換（DISCEN = 1、JDISCEN = 1）



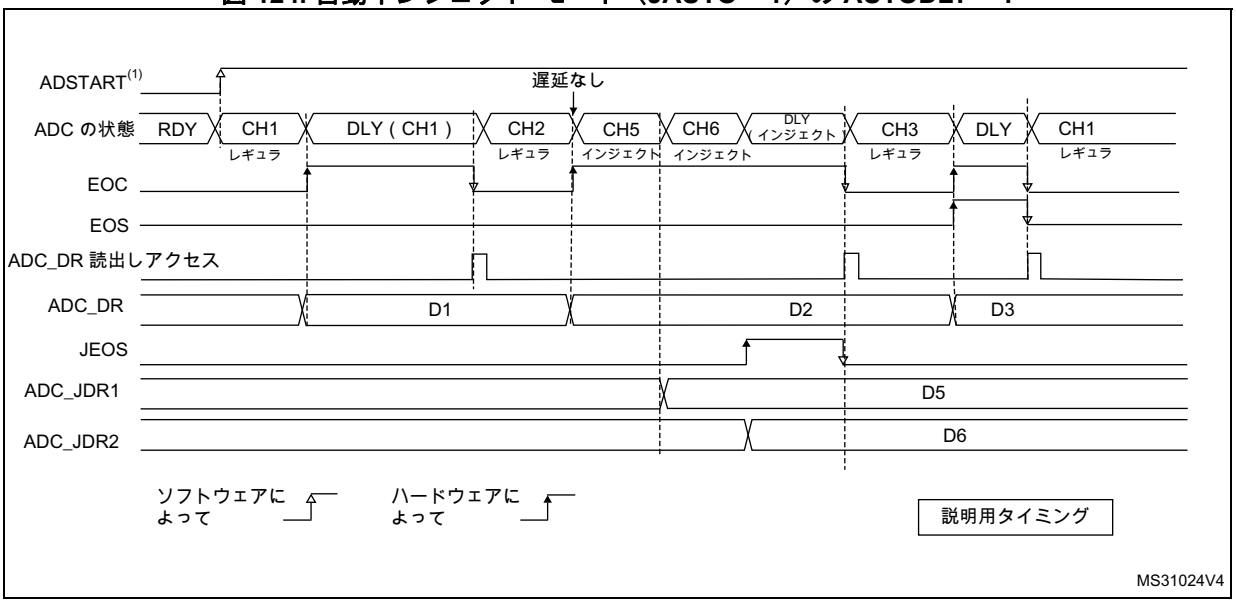
1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 01（ハードウェア･トリガ）、CONT = 0、DISCEN = 1、DISCNUM = 1、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN[1:0] = 01（ハードウェア･トリガ）、JDISCEN = 1、CHANNELS = 5、6

図 123. AUTODLY = 1、インジェクト変換によって中断されたレギュラ連続変換



1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 00（ソフトウェア･トリガ）、CONT = 1、DISCEN = 0、CHANNELS = 1、2、3
3. インジェクト設定：JEXTEN[1:0] = 01（ハードウェア･トリガ）、JDISCEN = 0、CHANNELS = 5、6

図 124. 自動インジェクト･モード（JAUTO = 1）の AUTODLY = 1

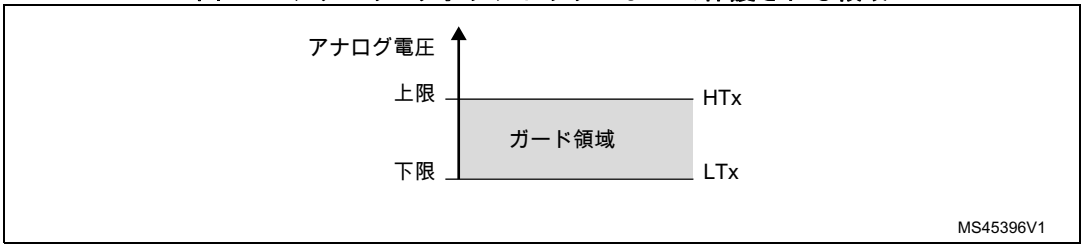


1. AUTDLY = 1
2. レギュラ設定：EXTEN[1:0] = 00（ソフトウェア･トリガ）、CONT = 1、DISCEN = 0、CHANNELS = 1、2
3. インジェクト設定：JAUTO = 1、CHANNELS = 5、6

21.4.28 アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）

3 つの AWD アナログウォッチドッグは、設定された電圧範囲（ウィンドウ）内に一部のチャネルが留まっているかどうかを監視します。

図 125. アナログ･ウォッチドッグによって保護される領域



AWDx フラグと割込み

ADC_IER レジスタの AWDyIE（x = 1、2、3）をセットすることによって、3 つのアナログ･ウォッチドッグの割込みをそれぞれ有効にできます。

AWDx（x = 1、2、3）フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

A/D コンバータの変換結果は、配置される前に低閾値および高閾値と比較されます。

アナログウォッチドッグ 1 の説明

AWD アナログウォッチドッグ 1 は、ADC_CFGR レジスタの AWD1EN ビットをセットすることによって有効になります。ウォッチドッグは、選択された 1 つのチャネルまたはすべての有効チャネル (1) が設定された電圧範囲（ウィンドウ）内にとどまっているかどうかを監視します。

表 169 に､1 つまたは複数のチャネル上でアナログウォッチドッグを有効にするためにどのようにして ADC_CFGR レジスタを設定するかを示します。

表 169. アナログウォッチドッグチャネル選択

アナログウォッチドッグによって保護されるチャネル	AWD1SGL ビット	AWD1EN ビット	JAWD1EN ビット
なし	x	0	0
すべてのインジェクトチャネル	0	0	1
すべてのレギュラチャネル	0	1	0
すべてのレギュラおよびインジェクトチャネル	0	1	1
1 つの(1) インジェクトチャネル	1	0	1
1 つの (1) レギュラチャネル	1	1	0
1 つの (1) レギュラまたはインジェクトチャネル	1	1	1

1. AWD1CH[4:0] ビットによって選択されます。チャネルは、適切なレギュラまたはインジェクトシーケンスで変換されるようにプログラムする必要があります。

アナログウォッチドッグ（AWD1）ステータスビットは、ADC によって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。

これらの閾値は、アナログウォッチドッグ 1 の ADC_TR1 レジスタの HT1[11:0] ビットおよび LT1[11:0] ビットでプログラムされます。12 ビット未満の分解能でデータを変換するときには（RES[1:0] ビットに従って）、内部比較は常に 12 ビット全体の元の変換データ（左詰め）に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

表 170 に、アナログウォッチドッグ 1 に可能なすべての分解能での比較方法を示します。

表 170. アナログウォッチドッグ 1 の比較

分解能（ビット RES[1:0]）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：12 ビット	DATA[11:0]	LT1[11:0] および HT1[11:0]	-
01：10 ビット	DATA[11:2],00	LT1[11:0] および HT1[11:0]	ユーザは LT1[1:0] と HT1[1:0] を 00 に設定する必要があります。
10：8 ビット	DATA[11:4],0000	LT1[11:0] および HT1[11:0]	ユーザは LT1[3:0] と HT1[3:0] を 0000 に設定する必要があります。
11：6 ビット	DATA[11:6],000000	LT1[11:0] および HT1[11:0]	ユーザは LT1[5:0] と HT1[5:0] を 000000 に設定する必要があります。

1. アナログウォッチドッグ比較の詳細についてはは、セクション ：ゲイン補正を参照してください。

ウォッチドッグ 1 のアナログウォッチドッグフィルタ

ADC が 1 つの入力チャネルだけで設定されているとき（スキャンモードで複数のチャネルを選択することはできない）、有効な ADC 変換データ間隔を ADC_TR1 レジスタで設定できます。

- 変換データが ADC_TR1 で定義された間に入ると、DMA リクエストが生成されます。
- そうでない場合、DMA リクエストは発行されません。RDATA レジスタは、変換ごとに更新されます。ADC_TR1 の AWDFILT ビットで指定された値より多い回数だけデータが範囲外であった場合、AWDx フラグがセットされ、対応する割込みが発行されます。

アナログ･ウォッチドッグ 2 および 3 の説明

2 番目と 3 番目のアナログ･ウォッチドッグはより柔軟で、AWDxCH[18:0]（x = 2、3）の対応するビットをプログラムすることにより、いくつかの選択されたチャネルをガードできます。

対応するウォッチドッグは、AWDxCH[18:0]（x = 2、3）の任意のビットがセットされると有効になります。

これらは 8 ビットの分解能に制限され、閾値の最上位の 8 ビットのみを HTx[7:0] と LTx[7:0] にプログラムすることができます。表 171 に、可能なすべての分解能での比較方法を示します。

表 171. アナログウォッチドッグ 2 および 3 の比較

分解能（ビット RES[1:0]）	アナログウォッチドッグ比較：		コメント
	元の変換データ、左詰め(1)	閾値	
00：12 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	DATA[3:0] はこの比較に適していません
01：10 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	DATA[3:2] はこの比較に適していません
10：8 ビット	DATA[11:4]	LTx[7:0] および HTx[7:0]	-
11：6 ビット	DATA[11:6],00	LTx[7:0] および HTx[7:0]	ユーザは LTx[1:0] と HTx[1:0] を 00 に設定する必要があります

1. アナログウォッチドッグ比較の詳細についてはは、セクション ：ゲイン補正を参照してください。

ADCy_AWDx_OUT 信号出力生成

各アナログウォッチドッグは、一部のオンチップタイマに ETR 入力（外部トリガ）が直接接続された内部ハードウェア信号 ADCy_AWDx_OUT （y=ADC の数、x=ウォッチドッグの数）と関連付けられています。ETR として ADCy_AWDx_OUT 信号を選択する方法を理解するには、オンチップタイマのセクションを参照してください。

ADCy_AWDx_OUT は、関連付けられたアナログウォッチドッグが有効になった時にアクティブ化されます。

- ADCy_AWDx_OUT は、ガードされた変換がプログラムされた閾値の外にある場合にセットされます。
- ADCy_AWDx_OUT は、次のガードされた変換がプログラムされた閾値内にある場合、終了後にリセットされます（次のガードされた変換がプログラムされた閾値の外にある場合は 1 のまま維持されます）。
- ADCy_AWDx_OUT は A/D コンバータを無効にした場合（ADDIS = 1 に設定した場合）もリセットされます。レギュラまたはインジェクト変換の停止（ADSTP = 1 または JADSTP = 1 に設定）は、ADCy_AWDx_OUT の生成にまったく影響しない点に注意してください。

注： **AWDx フラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。AWDx フラグは、ADCy_AWDx_OUT の生成にまったく影響しません（ソフトウェアによって AWDx フラグがクリアされず、このフラグが 1 のまま維持される間、ADCy_AWDx_OUT をトグルできます）。**

図 126. ADCy_AWDx_OUT 信号生成（すべてのレギュラチャネル）

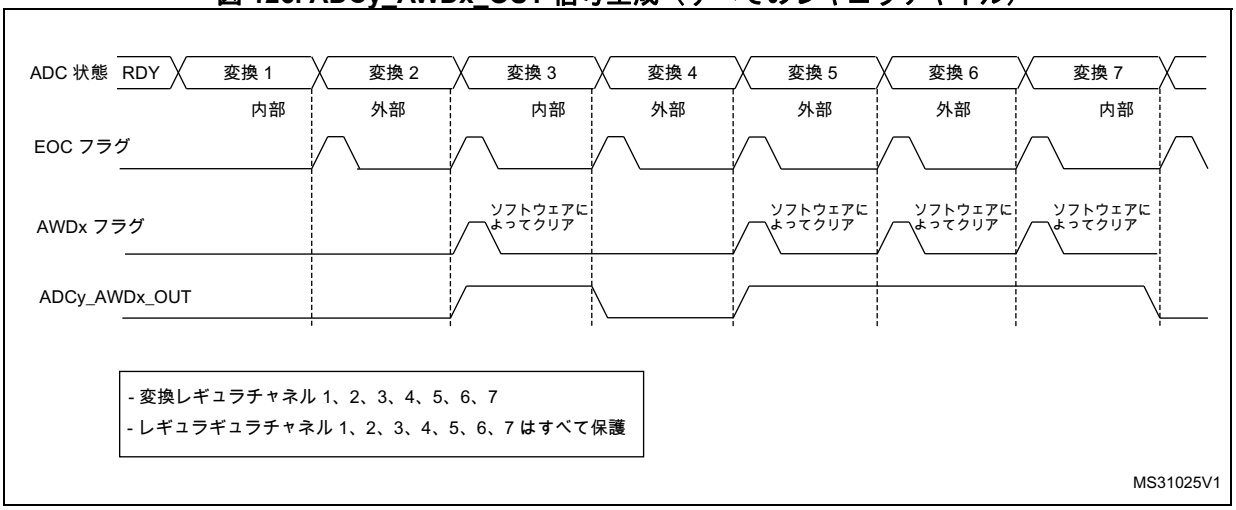


図 127. ADCy_AWDx_OUT 信号生成（ソフトウェアによって AWDx フラグがクリアされない場合）

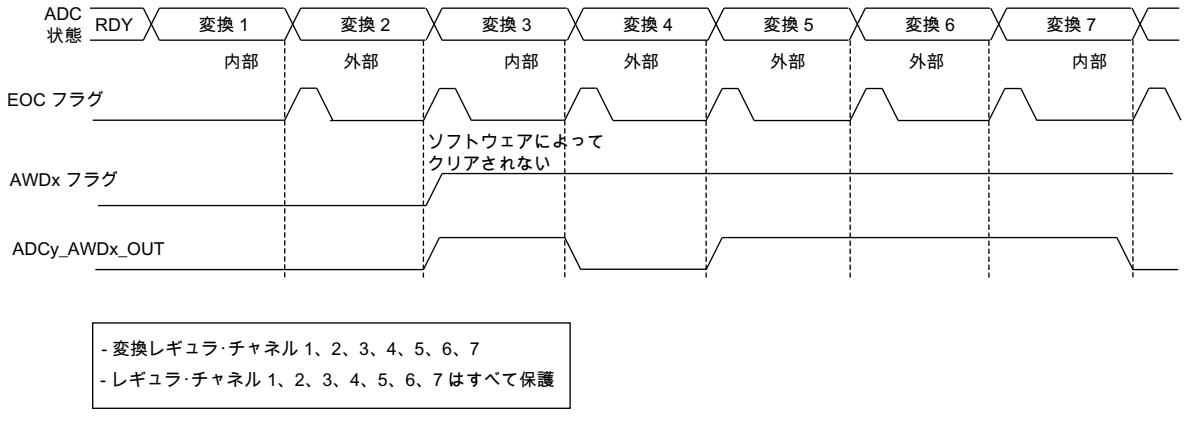


図 128. ADCy_AWDx_OUT 信号生成（1 つのレギュラチャネル）

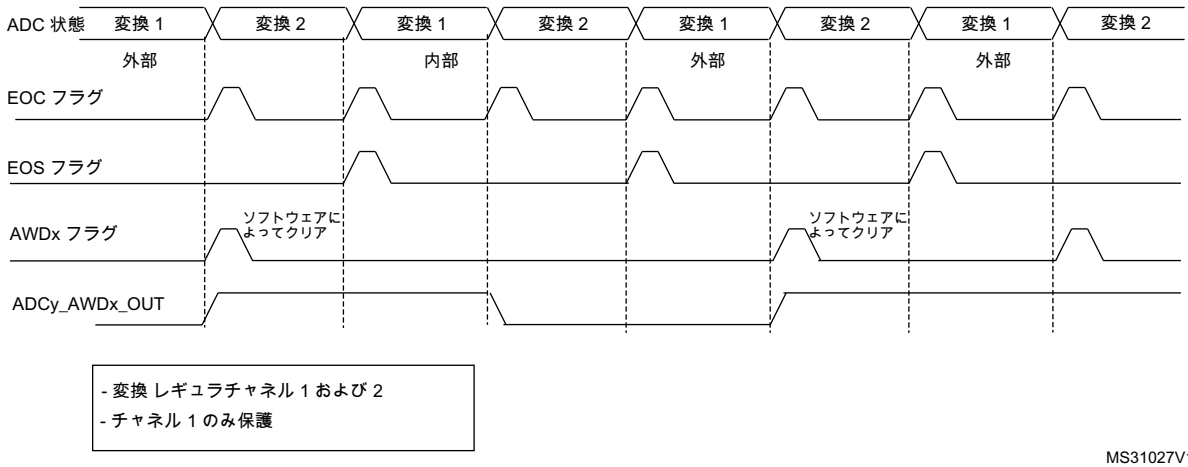
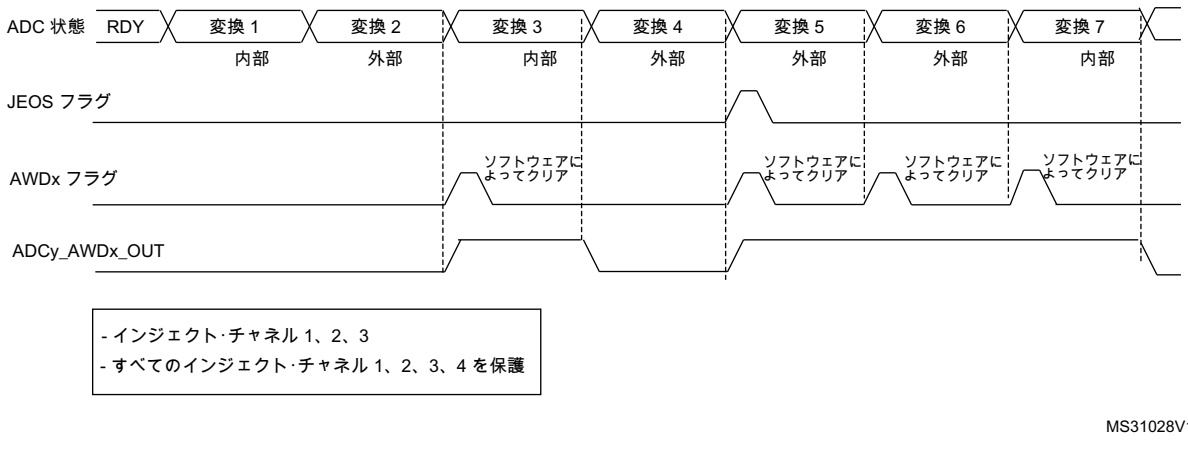


図 129. ADCy_AWDx_OUT 信号生成（すべてのインジェクト·チャネル）



アナログ･ウォッチドッグ閾値制御

LTx[11:0] および HTx[11:0] は、アナログデジタル変換中（ADC 内部状態の変換開始から変換終了後までの間）に変更できます。LTx[11:0] および HTx[11:0] が、ADC ガードチャネルの ADC 変換中に更新された場合、この変換については、ウォッチドッグ機能はマスクされます。このマスクは、次の変換の開始時に除去され、その結果、A/D コンバータによる次の変換からアナログ･ウォッチドッグ閾値が適用されます。アナログ･ウォッチドッグ比較は、変換が終了するたびに行われます。現在の ADC データが新しい間隔から外れていた場合、割込みまたは ADC_AWDx_OUT 信号は発行されません。割込みおよび AWD 生成は、閾値更新後に開始された変換の終了時にのみ行われます。AWD_xOUT がすでにアサートされていた場合、新しい閾値をプログラムしても、AWDx_OUT 信号はデアサートされません。

アナログウォッチドッグとゲインおよびオフセット補正

ゲインおよびオフセット補正が有効なとき、アナログウォッチドッグは閾値と補正後のデータを比較します。

注： **オフセット補正が有効なとき（ADC_OFRy レジスタの OFFSETy_EN がセットされている）、データ･オーバーフローまたはアンダーフローによって正しくないウォッチドッグ結果になることがあります。飽和が有効なとき（ADC_OFRy の SATEN がセットされている）、ウォッチドッグは正しい結果を提供します。ただし、この場合、符号付きデータ形式は使用できません。**

21.4.29 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPU の負荷を軽減します。複数の変換を処理して、最大 16 ビット幅の単一データに平均化できます。

以下の形式で結果を提供します。N および M は調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

平均化、データレートの削減、SNR の向上、基本的フィルタリングをハードウェアによって実行できます。

オーバーサンプリング比 N は、ADC_CFGR2 レジスタの OVFS[2:0] ビットによって定義され、その範囲は 2～256倍 です。分周係数 M は、最大 8 ビットの右ビットシフトから成っており、ADC_CFGR2 レジスタの OVSS[3:0] ビットを使用して定義します。

合計ユニットは最大 20 ビット（256x 12 ビットの結果）の結果をもたらし、最初に右へシフトされます。最下位 16 ビットまで切り詰められ、シフトによって残った最下位ビットを使用して最も近い値に丸められてから ADC_DR データレジスタに転送されます。

注： **シフト後の中間結果が 16 ビットを超える場合、結果は飽和なしで単純に切り詰められます。**

図 130. 20 ビットから 16 ビットへの結果の切り詰め

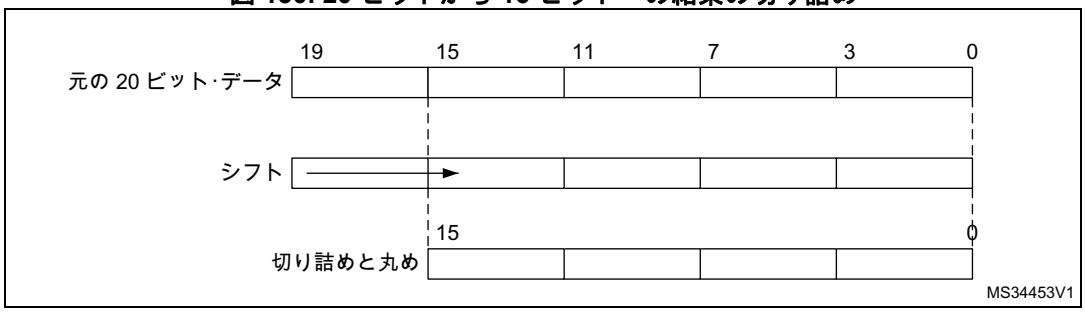


図 131 に、元の 20 ビットの累積データから最終的な 16 ビットの結果への処理の数値例を示します。

図 131. 5 ビット·シフトと丸めの数値例



表 172 に、元の変換データが 0xFFF の場合のさまざまな N と M の組み合わせでのデータフォーマットを示します。

表 172. 最大出力結果対 N と M（灰色のセルは切り詰め）

オーバーサンプリング比	最大値元のデータ	シフトなし OVSS = 0000	1 ビットシフト OVSS = 0001	2 ビットシフト OVSS = 0010	3 ビットシフト OVSS = 0011	4 ビットシフト OVSS = 0100	5 ビットシフト OVSS = 0101	6 ビットシフト OVSS = 0110	7 ビットシフト OVSS = 0111	8 ビットシフト OVSS = 1000
2x	0x1FFE	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040	0x020
4x	0x3FFC	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080	0x0040
8x	0x7FF8	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100	0x0080
16x	0xFFF0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200	0x0100
32x	0x1FFE0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400	0x0200
64x	0x3FFC0	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800	0x0400
128x	0x7FF80	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x0800
256x	0xFFF00	0xFF00	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF

オーバーサンプリング·モードでの変換タイミングに変更はありません。サンプル時間はオーバーサンプリング·シーケンス全体を通じて一定に保たれます。新しいデータは N 回の変換ごとに提供され、同等遅延は N x T_{CONV} = N x（t_{SMPL} + t_{SAR}）に等しくなります。フラグは、次のようにセットされます。

- サンプリング·フェーズの終了（EOSMP）は、各サンプリング·フェーズ後にセットされます。
- 変換の終了（EOC）は、N 回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了（EOS）は、オーバーサンプリングされたデータのシーケンスが完了すると発生します（すなわち、N x シーケンス長の変換合計後）。

オーバーサンプリング時の ADC 動作モードのサポート（シングル ADC モード）

オーバーサンプリングモードでは、ほとんどの ADC 動作モードが維持されます。

- シングルまたは連続モードの変換
- ソフトウェアまたはトリガによる ADC 変換の開始
- 変換中の ADC の停止（中止）
- オーバーラン検出時の CPU または DMA 経由でのデータの読出し
- 低電力モード（AUTDLY）
- プログラム可能な分解能：この場合、削減された変換値（ADC_CFGR1 レジスタの RES[1:0] ビットに従って）の累積、切り詰め、丸め、およびシフトは、12 ビット変換と同様に行われます。

注： **オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADC_CFGR1 の ALIGN ビットは無視され、データは常に右詰めで提供されます。**

オフセット補正は、オーバーサンプリングモードでサポートされません。ROVSE および／または JOVSE ビットがセットされている場合、ADC_OFRy レジスタの OFFSETy_EN ビットの内容は無視されます（リセットとみなされます）。

アナログウォッチドッグ

アナログウォッチドッグ機能は保持されますが（AWDSGL および AWDEN ビット）、次のような違いがあります。

- RES[1:0] ビットは無視され、比較は常に HT[11:0] および LT[11:0] の 12 ビットすべての値に基づいて実行されます。
- 比較は、16 ビットのオーバーサンプリングされた結果の上位 12 ビット ADC_DR[15:4] に対して行われます。

注： **高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなるからです。たとえば、オーバーサンプリングされた結果が 4 ビット･シフトされた場合、12 ビットの右詰めデータになり、有効なアナログ･ウォッチドッグ比較が行われるのは、8 ビットに対してだけです。比較は ADC_DR[11:4] と HT[0:7] / LT[[0:7] の間で行われ、HT[11:8] / LT[11:8] はリセットする必要があります。**

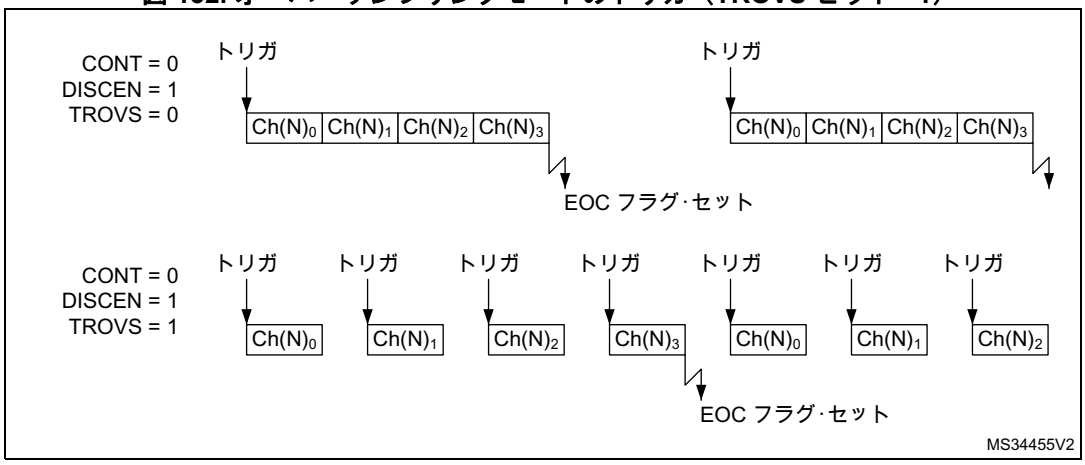
トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり強力なフィルタではありませんが（ロールオフが低速であり、阻止帯域減衰が限られる）、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます（一般に主電源またはスイッチモードの電源が原因）。この目的のために、ADC_CFGR2 の TROVS ビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

下記の図 132 に、不連続モード中にトリガに反応して変換を開始する方法を示します。

TROVS ビットがセットされている場合、DISCEN ビットの内容は無視され、1 とみなされます。

図 132. オーバーサンプリングモードのトリガ（TROVS ビット =1）



オーバーサンプリング時のインジェクトおよびレギュラシーケンサ管理

オーバーサンプリングモードでは、インジェクトおよびレギュラシーケンサに異なる動作を行わさせることができます。シーケンサを同時に使用する必要がある場合、それら両方を制限付きでオーバーサンプリングを有効にできます（これは、独自の累算ユニットに関連します）。

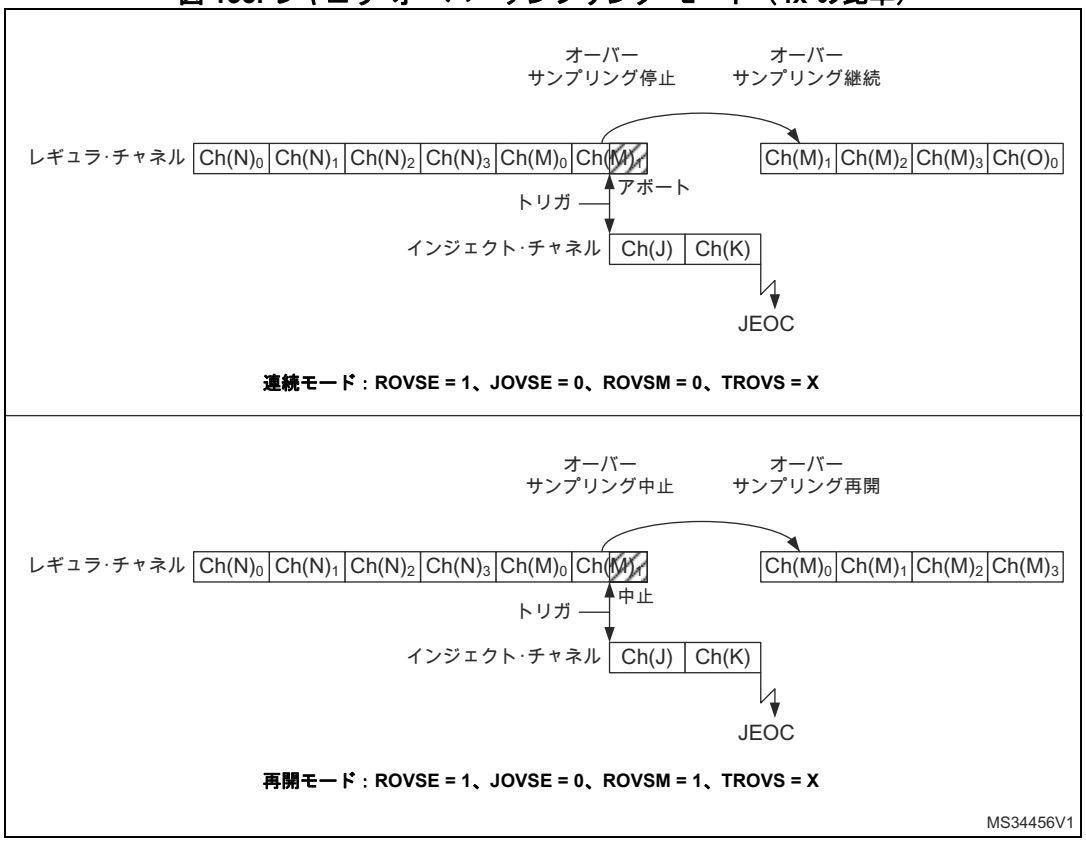
オーバーサンプリングレギュラチャネルのみ

レギュラオーバーサンプリングモードビット ROVSM は、インジェクト変換によって割込みが発生した場合のレギュラサンプリングシーケンスの再開方法を定義します。

- 連続モードでは、最後の有効なデータ（インジェクトトリガによる変換中止リクエストの前）から累算を再開します。これにより、インジェクションの頻度に関係なく、オーバーサンプリングを完了できます（トリガ間で最低 1 回のレギュラ変換を完了できることを前提とします）。
- 再開モードでは、累算が 0 から再開します（前の変換結果は無視されます）。このモードでは、オーバーサンプリングに使用されるすべてのデータが 1 つの時間スロット内で連続して変換されたことを保証できます。オーバーサンプリング時間を超えるインジェクショントリガ時間がかかることに注意が必要です。この条件が満たされない場合、オーバーサンプリングを完了できず、レギュラ·シーケンサはブロックされます。

図 133 では、4 倍のオーバーサンプリング比の例を示しています。

図 133. レギュラ･オーバーサンプリング･モード（4x の比率）



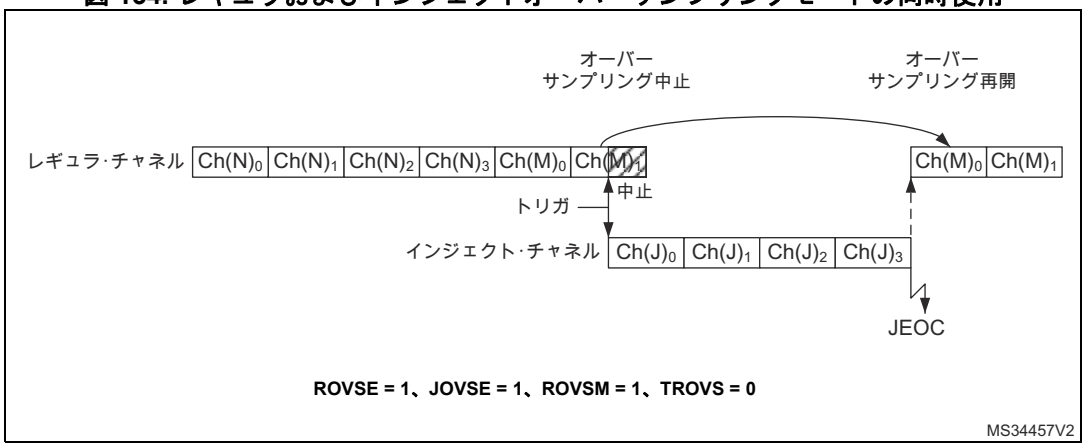
オーバーサンプリングインジェクトチャネルのみ

インジェクトオーバーサンプリングモードビット JOVSE は、インジェクトシーケンサでの変換専用のオーバーサンプリングを有効にします。

オーバーサンプリングレギュラおよびインジェクトチャネル

ROVSE ビットおよび JOVSE ビットの両方をセットできます。この場合、レギュラオーバーサンプリングモードは以下の図 134 に示すように、強制的に再開モードになります（ROVSM ビット無視）。

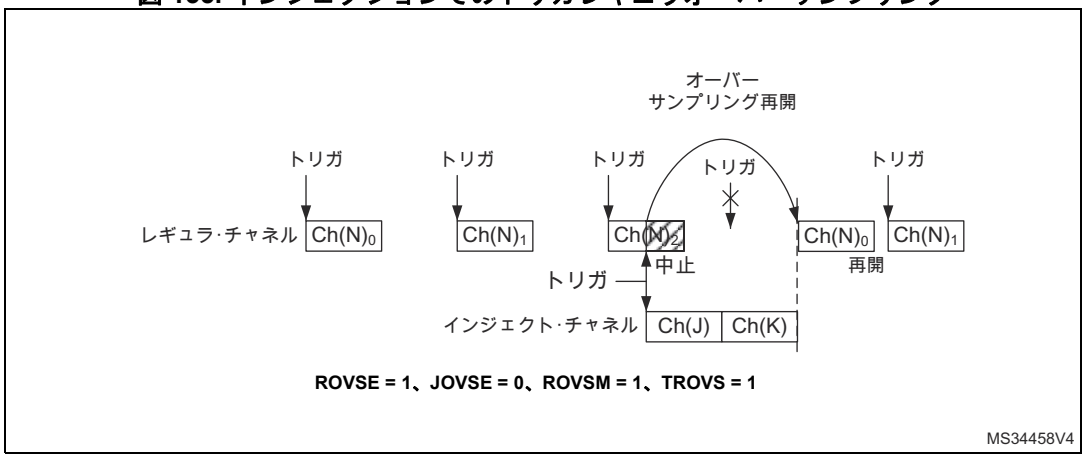
図 134. レギュラおよびインジェクトオーバーサンプリングモードの同時使用



インジェクト変換でのトリガレギュラオーバーサンプリングモード

インジェクト変換でのトリガレギュラオーバーサンプリングモードが可能です。この場合、インジェクトモードオーバーサンプリングモードは無効になり、ROVSM ビットは無視されます（再開モードが強制されます）。JOVSE ビットをリセットする必要があります。動作については、以下の図 135 に示します。

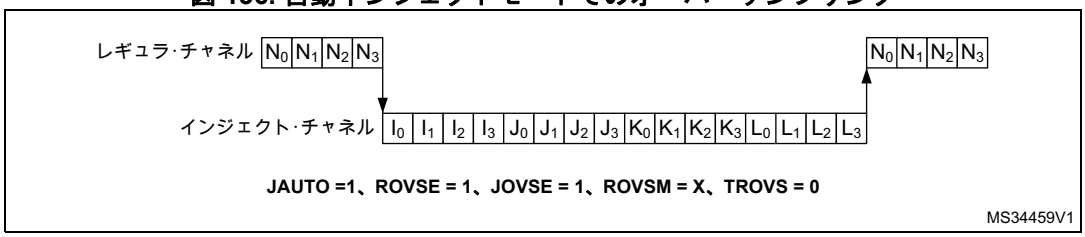
図 135. インジェクションでのトリガレギュラオーバーサンプリング



自動インジェクトモード

自動インジェクトシーケンスをオーバーサンプリングして、レジスタにすべての変換結果を格納し、DMA リソースを保存できます。このモードは、レギュラおよびインジェクトオーバーサンプリング両方がアクティブの状態でのみ使用可能です。JAUTO = 1、ROVSE = 1 および JOVSE = 1 の場合です。その他の組み合わせはサポートされません。ROVSM ビットは自動インジェクトモードでは無視されます。以下の図 136 に、変換シーケンスの方法を示します。

図 136. 自動インジェクトモードでのオーバーサンプリング



TROVS ビットを使用してトリガモードを有効にすることもできます。この場合、ADC は次のように設定する必要があります。JAUTO = 1、DISCEN = 0、JDISCEN = 0、ROVSE = 1、JOVSE = 1 および TROVSE = 1。

オーバーサンプリング時のデュアル ADC モードのサポート

デュアル ADC 設定で動作時、インジェクト同時モードおよびレギュラ同時モードに対してオーバーサンプリングを有効にできます。この場合、2 つの ADC をまったく同じ設定（オーバーサンプリング含む）でプログラムする必要があります。

他のすべてのデュアル ADC モードは、レギュラまたはインジェクトオーバーサンプリングが有効になっている場合（ROVSE = 1 または JOVSE = 1）サポートされません。

組み合わせモードの概要

以下の表 173 に、サポートされていないモードを含むすべての組み合わせの概要を示します。

表 173. オーバーサンプリング動作モードの概要

レギュラオーバーサンプリング ROVSE	インジェクトオーバーサンプリング JOVSE	オーバーサンプリングモード ROVSM 0 = 連続 1 = 再開	トリガレギュラモード TROVS	コメント
1	0	0	0	レギュラ連続モード
1	0	0	1	サポートされません
1	0	1	0	レギュラ再開モード
1	0	1	1	トリガレギュラ再開モード
1	1	0	X	サポートされません
1	1	1	0	インジェクトおよびレギュラ再開モード
1	1	1	1	サポートされません
0	1	X	X	インジェクトオーバーサンプリング

21.4.30 デュアル ADC モード

2 つ以上の ADC を持つデバイス（図 137 を参照）ではデュアル ADC モードを使用できます。

デュアル ADC モードでは、変換の開始は ADCx_CCR レジスタのDUAL[4:0] ビットによって選択されたモードに応じて、ADCx マスタにより ADC スレーブに交互または同時にトリガされます。

次の 4 つのモードを備えています。

- インジェクト同時モード
- レギュラ同時モード
- インタリーブモード
- オルタネートトリガモード

これらのモードを次のように組み合わせて使用することも可能です。

- インジェクト同時モード + レギュラ同時モード
- レギュラ同時モード + オルタネートトリガモード
- インジェクト同時モード + インタリーブモード

デュアル ADC モードで（ADCx_CCR レジスタの DUAL[4:0] ビットがゼロ以外の場合）、ADC_CFGR レジスタの CONT、AUTDLY、DISCEN、DISCNUM[2:0]、JDISCEN、JQM、JAUTO ビットは、マスタおよびスレーブ ADC の間で共有されます。スレーブ ADC のビットは、常にマスタ ADC の対応するビットと等しくなります。

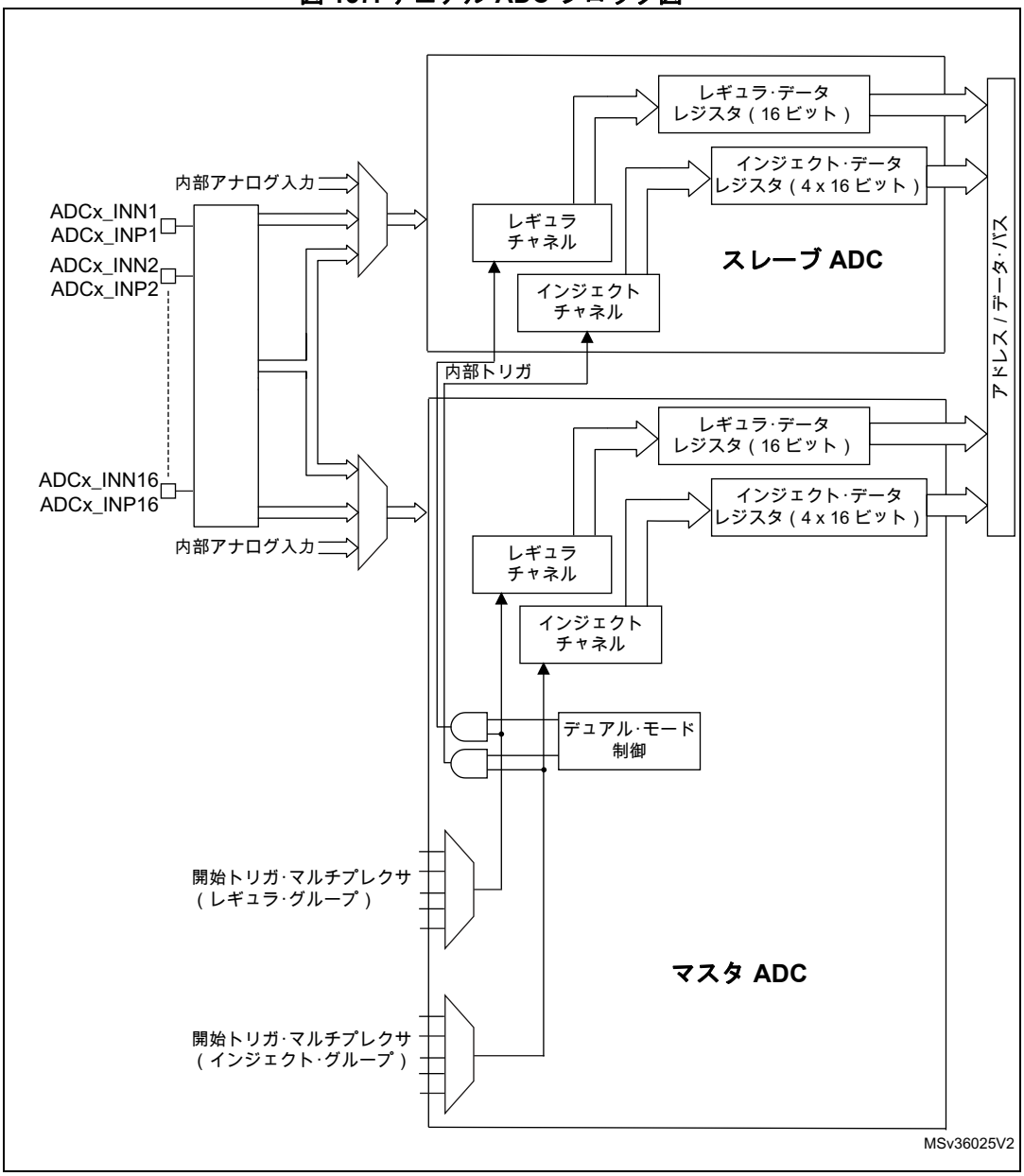
デュアル·モードで変換を開始するには、ソフトウェアまたはハードウェアのトリガやレギュラまたはインジェクトのトリガを設定するために、マスタ A/D コンバータの EXTEN[1:0] 、EXTSEL、JEXTEN[1:0]、JEXTSEL ビットのみをプログラムする必要があります（スレーブ A/D コンバータの EXTEN[1:0] および JEXTEN[1:0] ビットは無視されます）。

レギュラ同時モードまたはインタリーブモードでは、マスタ ADC のADSTART または ADSTPビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のADSTART または ADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

インジェクト同時モードまたはオルタネートトリガモードでは、マスタ ADC のJADSTART または JADSTP ビットを一度セットすると、スレーブ ADC の対応するビットも自動的にセットされます。ただし、スレーブ ADC のJADSTART または JADSTPビット は、必ずしもマスタ ADC ビットと同時にクリアされません。

デュアル ADC モードでは、ADC 共通データレジスタ（ADCx_CDR）を読み出すことで、マスタおよびスレーブ ADC の変換データを並列で読み出すことができます。ステータスビットも、デュアルモードステータスレジスタ（ADCx_CSR）を読み出すことで、並列で読み出すことができます。

図 137. デュアル ADC ブロック図 (1)



1. スレーブ ADC にも外部トリガは存在しますが、この図には表示していません。
2. ADC 共通データレジスタ（ADCx_CDR）にはマスタとスレーブの両方の ADC レギュラ変換データが格納されます。

インジェクト同時モード

このモードは、ビット DUAL[4:0] = 00101 をプログラムすることによって選択されます。

このモードは、インジェクト･チャネル･グループを変換します。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサ（ADC_JSQR レジスタの JEXTSEL ビットによって選択）から供給されます。

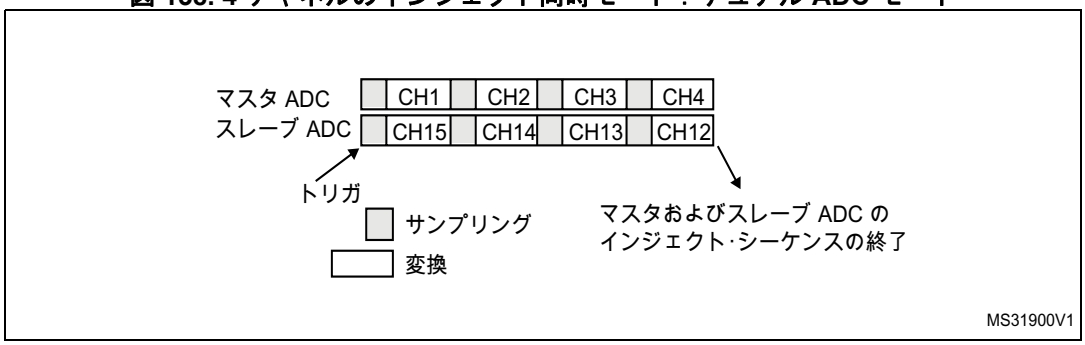
注： **2 つの ADC の同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの ADC のサンプリング時間が重なってはいけません）。**

同時モードでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち長い方のシーケンスより長くなるようにしなければなりません。そうしないと、最長のシーケンスを実行している A/D コンバータが前の変換を完了する前に、最短のシーケンスを実行している A/D コンバータがリスタートすることがあります。

レギュラ変換であれば、1 つまたはすべての ADC で行うことができます。その場合、それらは互いに独立していてインジェクトイベントが発生すると中断されます。それら（レギュラ変換）はインジェクト変換グループ終了時に再開されます。

- マスタ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはマスタ ADC_JDRy レジスタに格納され、JEOS 割込みが生成されます（有効な場合）。
- スレーブ ADC での変換イベントのインジェクトシーケンスの終了時（JEOS）、変換データはスレーブ ADC_JDRy レジスタに格納され、JEOS 割込みが生成されます（有効な場合）。
- マスタインジェクトシーケンスの長さがスレーブインジェクトシーケンスの長さと等しい場合（例：図 138）、ソフトウェアでは 2 つの JEOS 割込みのうちの 1 つのみを有効にすることができ（例：マスタ JEOS）、（マスタ ADC_JDRy とスレーブ ADC_JDRy レジスタの）両方の変換データを読み出すことができます。

図 138. 4 チャネルのインジェクト同時モード：デュアル ADC モード



JDISCEN = 1 の場合、インジェクト･シーケンスの各同時変換にはインジェクト･トリガ･イベントが発生する必要があります。

このモードは、AUTDLY モードと組み合わせることができます。

- 変換の同時インジェクトシーケンスの終了後、マスタとスレーブADCの両方の JEOS ビットがクリアされた場合（遅延フェーズ）にのみ、新しいインジェクトトリガイベントは受け入れられます。実行中のインジェクトシーケンスで発生したすべての新しいインジェクトトリガイベントと、これに関連する遅延フェーズは無視されます。
- マスタ ADC のレギュラ変換シーケンスが終了すると、マスタデータレジスタ（ADC_DR）が読み出された場合にのみ、マスタ ADC の新しいレギュラトリガイベントが受け入れられます。実行中のレギュラシーケンスでマスタ ADC に対して発生したすべての新しいレギュラトリガイベントと、これに関連する遅延フェーズは無視されます。
  スレーブ ADC で発生するレギュラシーケンスでも同じ挙動を示します。

独立インジェクト変換に対応したレギュラ同時モード

このモードは、ビット DUAL[4:0]=00110 をプログラムすることによって選択されます。

このモードは、レギュラチャネルグループに対して行われます。外部トリガのソースは、マスタ ADC のレギュラグループマルチプレクサ（ADC_CFGR レジスタの EXTSEL ビットによって選択）から供給されます。同時トリガは、スレーブ ADC に供給されます。

このモードでは、独立インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクション·リクエストにより、現在の同時変換がアボートされます。これらの変換は、インジェクト変換の完了後に再開されます。

注： **2 つの A/D コンバータの同じチャネルを変換しないでください（同じチャネルを変換する場合、2 つの A/D コンバータのサンプリング時間が重なってはいけません）。**

レギュラ同時モードでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、長いシーケンスの ADC が前の変換を完了する前に、短いシーケンスの ADC がリスタートすることがあります。

データが読み出せるタイミングは、割込みによってソフトウェアに通知されます。

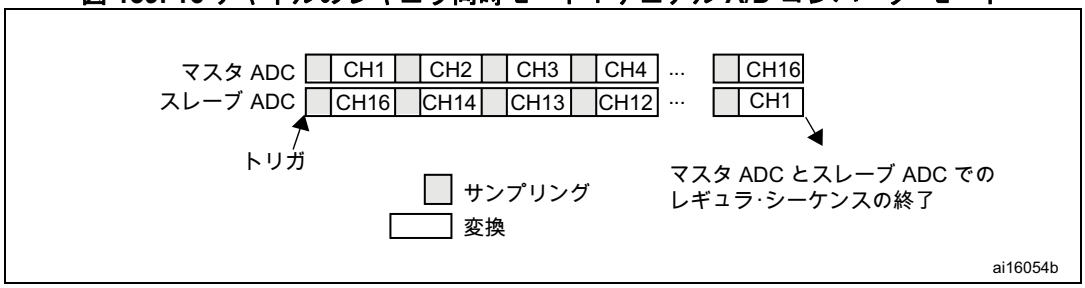
- マスタ ADC での各変換イベント（EOC）の終了時、マスタ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではマスタ ADC の ADC_DR を読み出すことができます。
- スレーブ ADC での各変換イベント（EOC）の終了時、スレーブ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ ADC の ADC_DR を読み出すことができます。
- マスタレギュラシーケンスの長さがスレーブレギュラシーケンスの長さと等しい場合（例：図 139）、ソフトウェアでは 2 つの EOC 割込みのうちの 1 つのみを有効にすることができ（例：マスタ EOC）、共通データレジスタ（ADCx_CDR）から両方の変換データを読み出すことができます。

DMA を使用してレギュラデータを読み出すことができます。次の 2 つの方法が可能です。

- 2 つの DMA チャネルを使用します（1つはマスタ用、もう 1つはスレーブ用）。この場合、ビット MDMA[1:0] をクリア状態に保つ必要があります。
  - マスタから ADC_DR を読み出すために、DMA のマスタ ADC チャネルを設定します。DMA リクエストは、マスタ ADC の各 EOC イベントで生成されます。
  - スレーブから ADC_DR を読み出すために、DMA のスレーブ ADC チャネルを設定します。DMA リクエストは、スレーブ ADC の各 EOC イベントで生成されます。
- MDMAモードを使用して、他の用途のために1つのDMA チャネルを残します。
  - MDMA[1:0] = 10 または 11 を設定します（分解能による）。
  - DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADCx_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
  - マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ A/D コンバータの変換データは ADCx_CDR 32 ビット·レジスタの上位ハーフワードから、マスタ A/D コンバータの変換データは ADCx_CCR レジスタの下位ハーフワードから取得できます。
  - EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

注： **MDMA モード（MDMA[1:0] = 10 または 11）で、ユーザはマスタのシーケンスとスレーブのシーケンスに同じ変換数をプログラムする必要があります。そうしないと、残りの変換では DMA リクエストが生成されません。**

図 139. 16 チャネルのレギュラ同時モード：デュアル A/D コンバータ･モード



DISCEN = 1 の場合、レギュラ･シーケンスの各「n」回の同時変換でレギュラ･トリガ･イベントが発生する必要があります（「n」は DISCNUM で定義される）。

このモードは、AUTDLY モードと組み合わせることができます。

- シーケンスの同時変換の終了後、シーケンスの次の変換は、共通データレジスタ ADCx_CDR（またはマスタ ADC のレギュラデータレジスタ）が読み出された場合（遅延フェーズ）にのみ開始されます。
- 変換の同時レギュラシーケンスの終了後､新しいレギュラトリガイベントは共通データレジスタ（ADCx_CDR）が読み出された場合（遅延フェーズ）にのみ受け入れられます。実行中のレギュラシーケンスで発生したすべての新しいレギュラトリガイベントと､これに関連する遅延フェーズは無視されます。

AUTDLY モードと組み合わせたレギュラ同時モードでデータを処理するために、DMA を使用することができます（このとき、マルチ DMA モードが使われるものとします。つまり MDMA ビットが 10 または 11 に設定されていなければなりません）。

レギュラ同時モードを AUTDLY モードと組み合わせる場合、ユーザは次の内容を確認する必要があります。

- マスタのシーケンスの変換数が、スレーブの変換数と同じであること。
- シーケンスが同時変換されるたび、スレーブ ADC の変換の長さがマスタ ADC の変換の長さよりも短いこと。シーケンスの長さは、変換するチャネルの数および各チャネルのサンプリングタイムと分解能に依存する点に注意してください。

注： **このレギュラ同時モードと AUTDLY モードの組み合わせは、レギュラチャネルのみがプログラムされている場合に制限されます。この組み合わせモードでインジェクトチャネルをプログラムすることは禁止されています。**

独立インジェクト変換に対応したインタリーブモード

このモードは、ビット DUAL[4:0]=00111 をプログラムすることによって選択されます。

このモードは、レギュラグループ（通常は 1 つのチャネル）でのみ開始できます。外部トリガソースは、マスタ ADC のレギュラチャネルマルチプレクスから供給されます。

外部トリガが発生した後、

- マスタ ADC はすぐに開始します。
- スレーブ ADC は、マスタ ADC のサンプリングフェーズが完了した後で、数 ADC クロックサイクルの遅延後に開始します。

インタリーブモードにおける 2 つの変換間の最小遅延は ADCx_CCR レジスタの DELAY ビットで設定されます。この遅延は、マスタ変換のサンプリングフェーズが終了してから半サイクル後にカウントを開始します。これにより、ADC は相補 ADC が依然として入力をサンプリングしている場合、変換を開始できません（与えられた時間に 1 つの ADC のみ入力信号をサンプリングすることができます）。

- DELAY に設定可能な最小値は 1 です。これにより、マスタ ADC サンプリングフェーズのアナログスイッチを開いてスレーブ ADC サンプリングフェーズのアナログスイッチを閉じるまでに、最低 1 つのサイクル時間を確保できます。
- 最大 DELAY 値は選択した分解能に対するサイクル数と同じです。ただし、ほかの ADC が依然として入力をサンプリングしているときに ADC が変換を開始しないよう、ユーザはこの遅延を適切に計算する必要があります。

マスタとスレーブの両方の ADC で CONT ビットがセットされている場合、両方の ADC の選択されたレギュラチャネルが連続的に変換されます。

スレーブ ADC での各変換の終了イベント（EOC）時に、データを読み出せるタイミングが、割込みによってソフトウェアに通知されます。スレーブおよびマスタ EOC 割込みが生成され（EOCIE が有効な場合）、ソフトウェアではスレーブ／マスタ ADC の ADC_DR を読み出すことができます。

注： **スレーブの EOC 割込みのみを有効にして共通データレジスタ（ADCx_CDR）を読み出すことができます。ただしこの場合、ユーザは変換時間が適合しており、それによって新しいマスタ変換をリスタートする前に、シーケンス内では常にマスタ変換に続いてスレーブ変換が実行されることを確認する必要があります。MDMA モードを使用することを推奨します。**

DMA で転送されたレギュラデータを持つこともできます。この場合、各 ADC の個別の DMA リクエストを使用できず、次のように MDMA モードの使用が必須になります。

- MDMA[1:0] = 10 または 11 を設定します（分解能による）。
- DMA チャネルを 1 つ使用します（マスタのうちの 1 つ）。共通 ADC レジスタ（ADCx_CDR）を読み出すために、DMA のマスタ ADC チャネルを設定します。
- マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、スレーブ ADC の変換データを ADCx_CDR 32 ビットレジスタの上位ハーフワードで使用でき、またマスタ ADC の変換データを ADCx_CCR レジスタの下位ハーフワードで使用できます。
- EOC フラグは、DMA が ADCx_CCR レジスタを読み出すと両方ともクリアされます。

図 140. 連続変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード

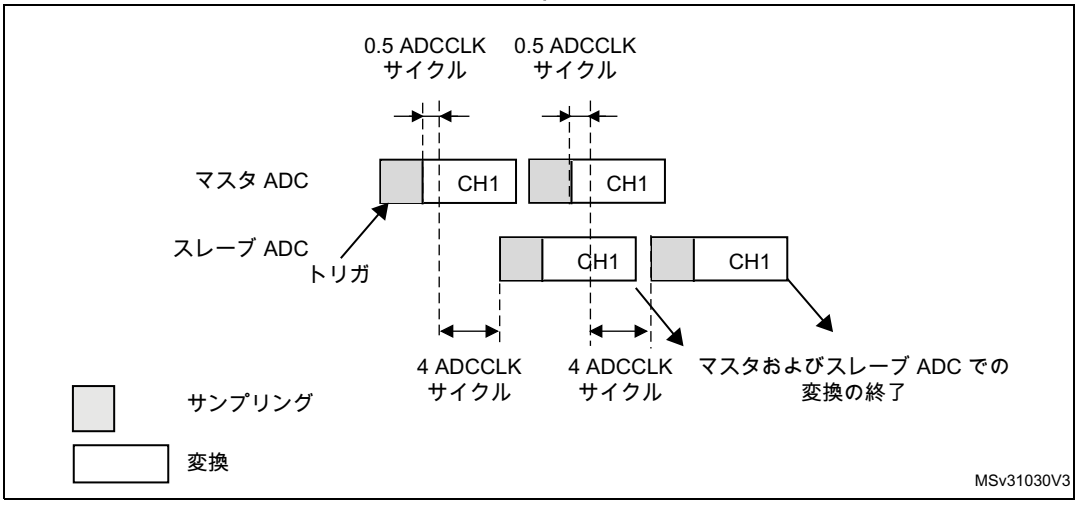
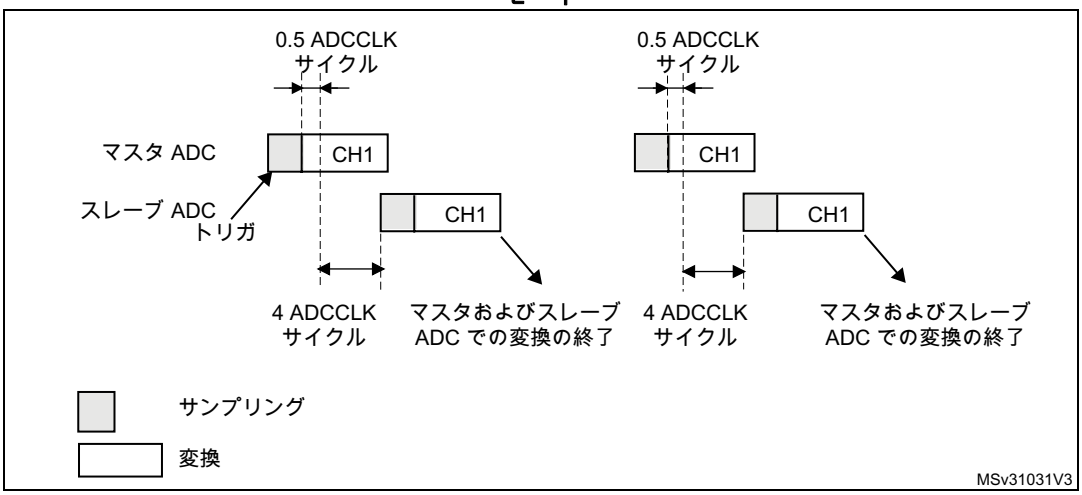


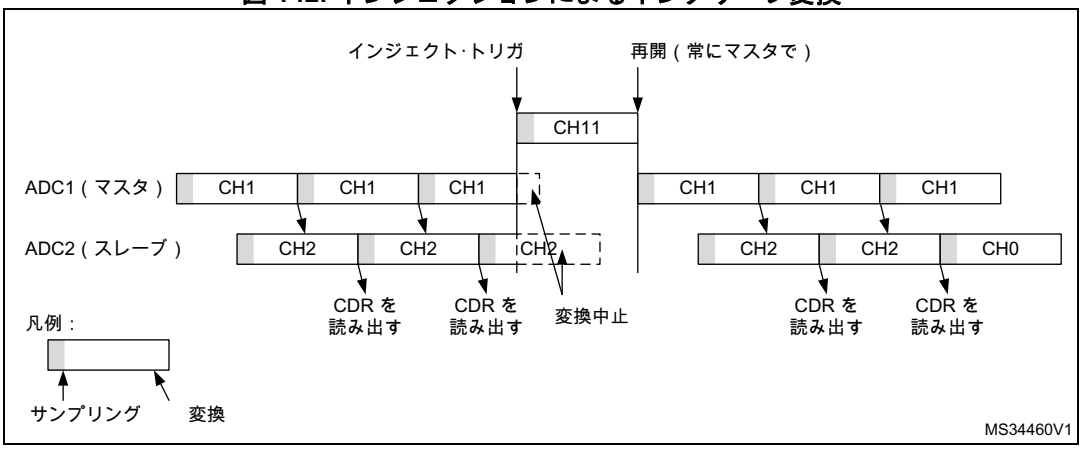
図 141. シングル変換モードにおける 1 チャネルのインタリーブモード：デュアル ADC モード



DISCEN = 1 の場合、レギュラ･シーケンスの各「n」回の同時変換（「n」は DISCNUM で定義される）でレギュラ･トリガ･イベントが発生する必要があります。

このモードでは、インジェクト変換がサポートされます。（マスタまたはスレーブでの）インジェクションが完了すると、マスタとスレーブの両方のレギュラ変換がアボートされ、マスタからシーケンスが再開されます（以下の図 142 を参照）。

図 142. インジェクションによるインタリーブ変換



オルタネートトリガモード

このモードは、ビット DUAL[4:0]=01001 をプログラムすることによって選択されます。

このモードは、インジェクトグループでのみ開始できます。外部トリガのソースは、マスタ ADC のインジェクトグループマルチプレクサです。

このモードは、ハードウェアトリガの選択時にのみ使用できます。JEXTEN[1:0] に 00 を設定しないでください。

インジェクト不連続モードは無効です（両方の A/D コンバータで JDISCEN = 0）。

1. 最初のトリガが発生すると、グループ内のインジェクトされたマスタ A/D コンバータの各チャネルが変換されます。
2. 2 番目のトリガが発生すると、グループ内のすべてのインジェクトされたスレーブ ADC チャネルが変換されます。
3. 以下同様です。

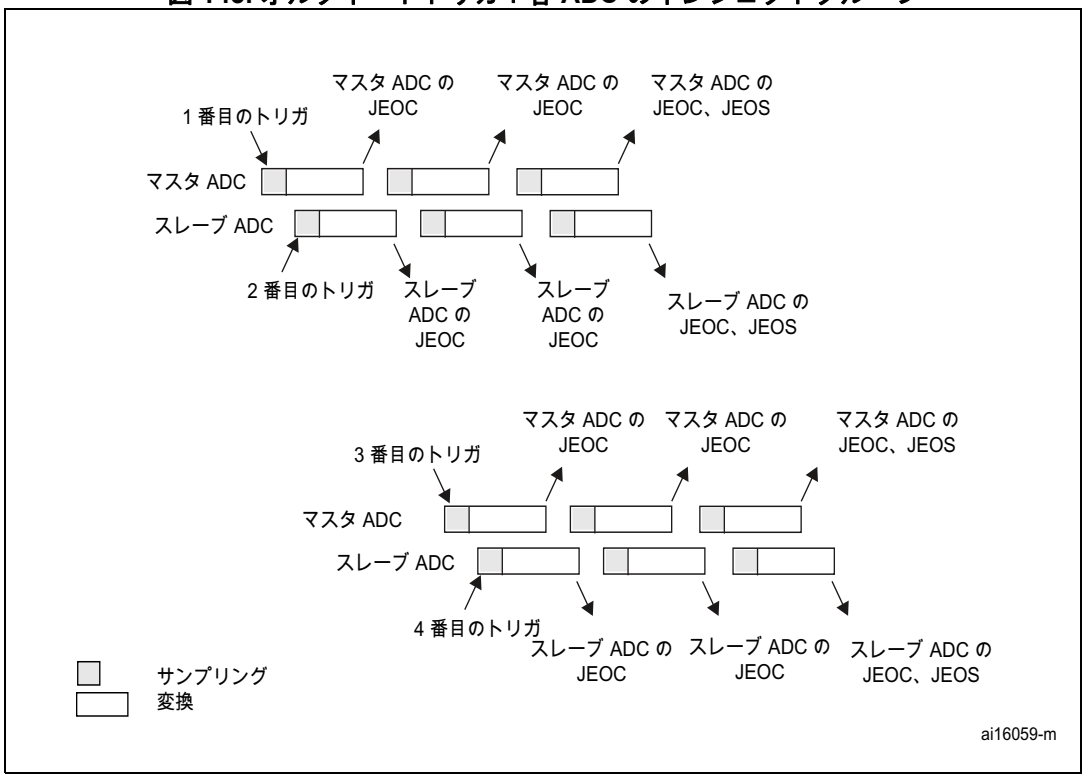
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

JEOC 割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内すべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、グループ内のマスタ ADC のインジェクトチャネルを変換することによって、オルタネートトリガプロセスがリスタートします。

図 143. オルタネートトリガ：各 ADC のインジェクトグループ



注： **レギュラ変換であれば 1 つまたはすべての ADC で有効にすることができます。この場合、レギュラ変換は互いに独立しています。ADC がインジェクト変換を行う必要があるとき、レギュラ変換は中断されます。インジェクト変換が終了すると(レギュラ変換が)再開されます。**

2 つのトリガイベント間の間隔は 1 ADC クロック期間と同等か長い必要があります。同じ ADC で変換を開始する 2 つのトリガイベント間の最小間隔はシングル ADC モードと同じです。

インジェクト不連続モードは有効です（両方の A/D コンバータで JDISCEN = 1）。

マスタとスレーブの両方の A/D コンバータに対してインジェクト不連続モードが有効な場合、

- 最初のトリガが発生すると、マスタ ADC の最初のインジェクトチャネルが変換されます。
- 2 番目のトリガが発生すると、スレーブ ADC の最初のインジェクトチャネルが変換されます。
- 以下同様です。

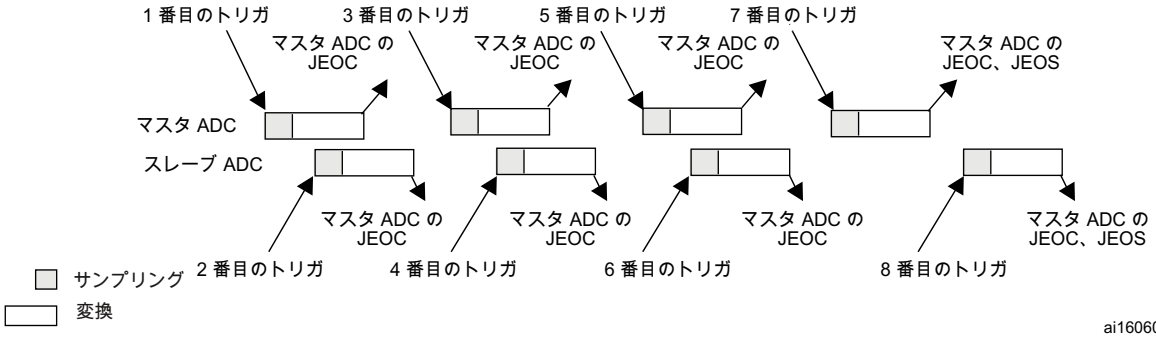
グループ内のマスタ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

グループ内のスレーブ ADC のすべてのインジェクトチャネルが変換されると、JEOS 割込み（有効な場合）が生成されます。

JEOC 割込み（有効な場合）は、各インジェクト変換後に生成することもできます。

グループ内のすべてのインジェクトチャネルが変換された後で別の外部トリガが発生した場合は、オルタネートトリガプロセスがリスタートします。

図 144. オルタネートトリガ：不連続モードにおける 4 つのインジェクトチャネル（各 ADC）



レギュラ／インジェクト同時モードの組み合わせ

このモードは、ビット DUAL[4:0] = 00001 をプログラムすることによって選択されます。

レギュラグループの同時変換を中断して、インジェクトグループの同時変換を開始することができます。

注： **レギュラ／インジェクト同時モードの組み合わせでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、最長のシーケンスを実行している A/D コンバータが前の変換を完了する前に、最短のシーケンスを実行している A/D コンバータがリスタートすることがあります。**

レギュラ同時 + オルタネートトリガモードの組み合わせ

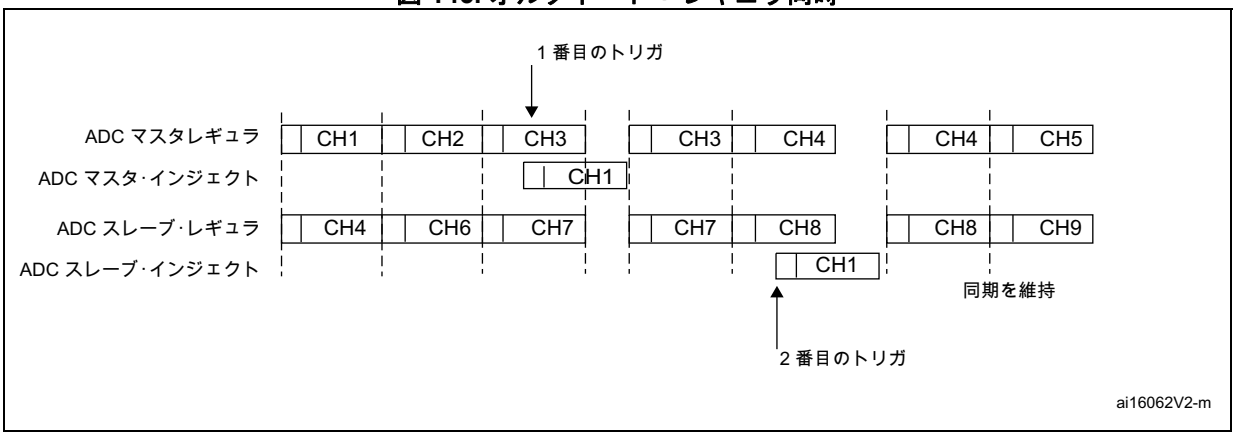
このモードは、ビット DUAL[4:0] = 00010 をプログラムすることによって選択されます。

レギュラ･グループの同時変換を中断して、インジェクト･グループのオルタネート･トリガ変換を開始することができます。図 145 は、同時レギュラ変換に割り込むオルタネートトリガの動作を示します。

インジェクトオルタネート変換は、インジェクトイベント後、ただちに開始されます。すでにレギュラ変換が実行中の場合、インジェクト変換後の同期を確保するために、すべての（マスタ／スレーブ）ADC のレギュラ変換は停止し、インジェクト変換の終了と同期して再開されます。

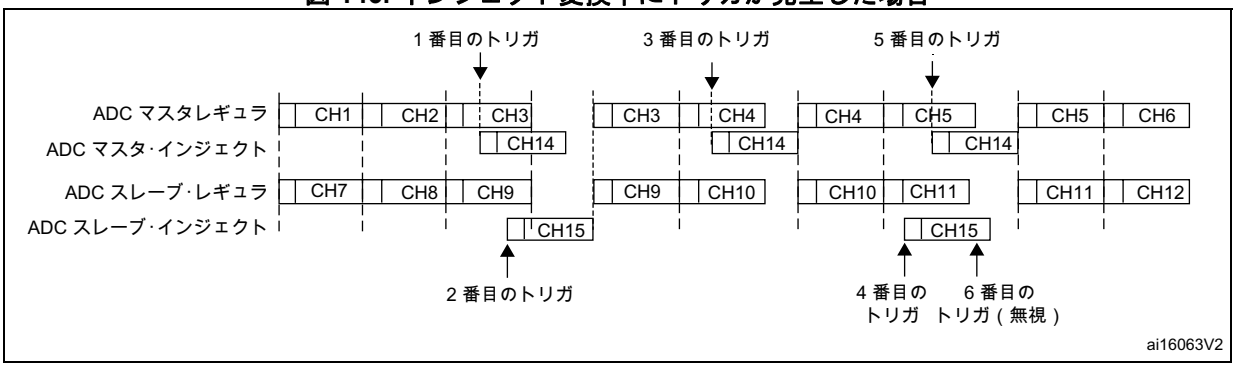
注： **レギュラ同時 + オルタネートトリガモードの組み合わせでは、同じ長さのシーケンスを変換するか、トリガ間隔が 2 つのシーケンスのうち変換時間が長い方のシーケンスより長くなるようにしなければなりません。そうしないと、最長のシーケンスを実行している A/D コンバータが前の変換を完了する前に、最短のシーケンスを実行している A/D コンバータがリスタートすることがあります。**

図 145. オルタネート + レギュラ同時



レギュラ変換に割り込んだインジェクト変換の最中にトリガが発生した場合はオルタネートトリガが処理されます。図 146 に、この場合の挙動を示します（関連付けられたオルタネート変換は完了していないため、6 番目のトリガは無視されます）。

図 146. インジェクト変換中にトリガが発生した場合



インジェクト同時モードとインタリーブの組み合わせ

このモードは、ビット DUAL[4:0] = 00011 をプログラムすることによって選択されます。

インタリーブ変換を同時インジェクト·イベントで中断できます。

この場合、インタリーブ変換は即座に割り込まれ、同時インジェクト変換が開始します。インジェクトシーケンス終了時に、インタリーブ変換が再開されます。インタリーブレギュラ変換が再開すると、実行される最初のレギュラ変換は常にマスタのものです。図 147、図 148 および図 149 には、例を使用して動作を示しています。

注意： **このモードでは、1 回の読出しアクセスでレギュラデータを読み出すために共通データレジスタを使用する必要があります。逆に、マスタ - スレーブのデータコヒーレンシは保証されません。**

図 147. インジェクトシーケンス CH11、CH12 とインタリーブシングルチャネル CH 0

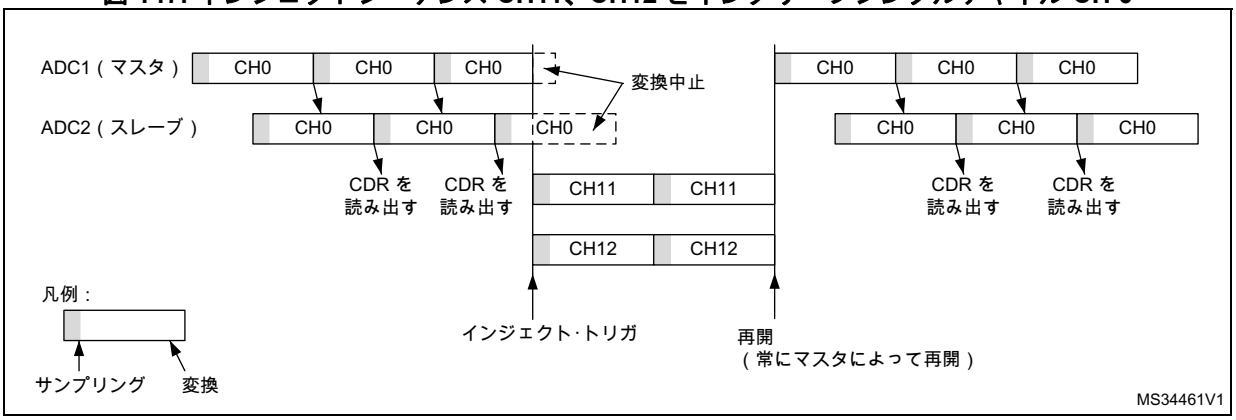


図 148. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 1：マスタ割込み優先

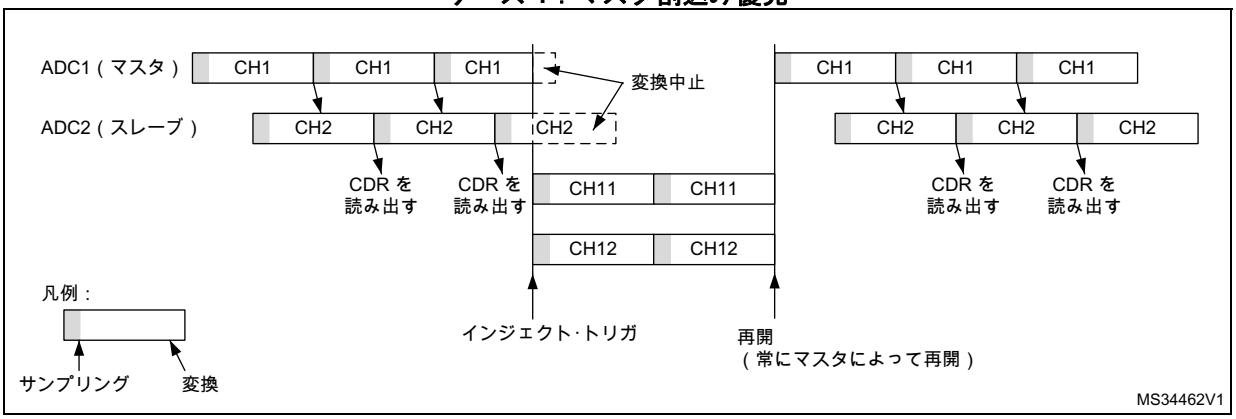
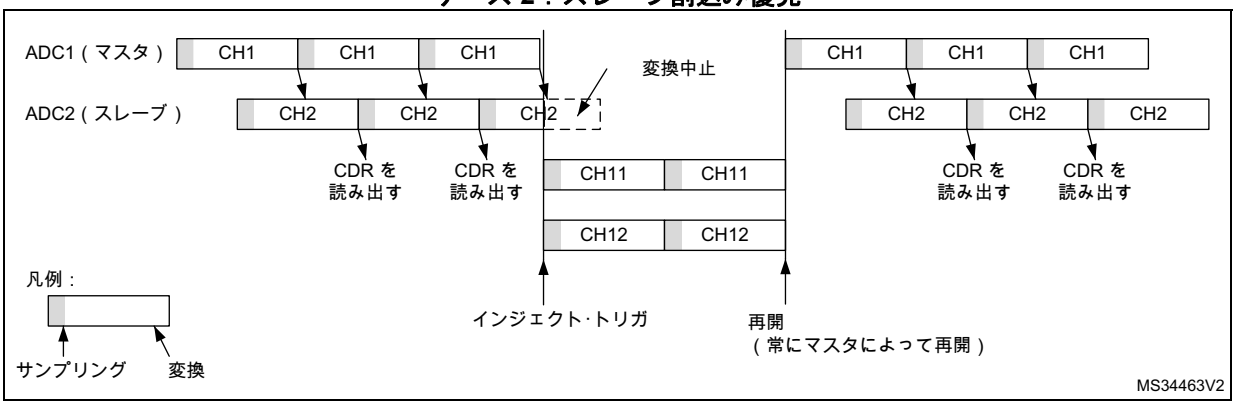


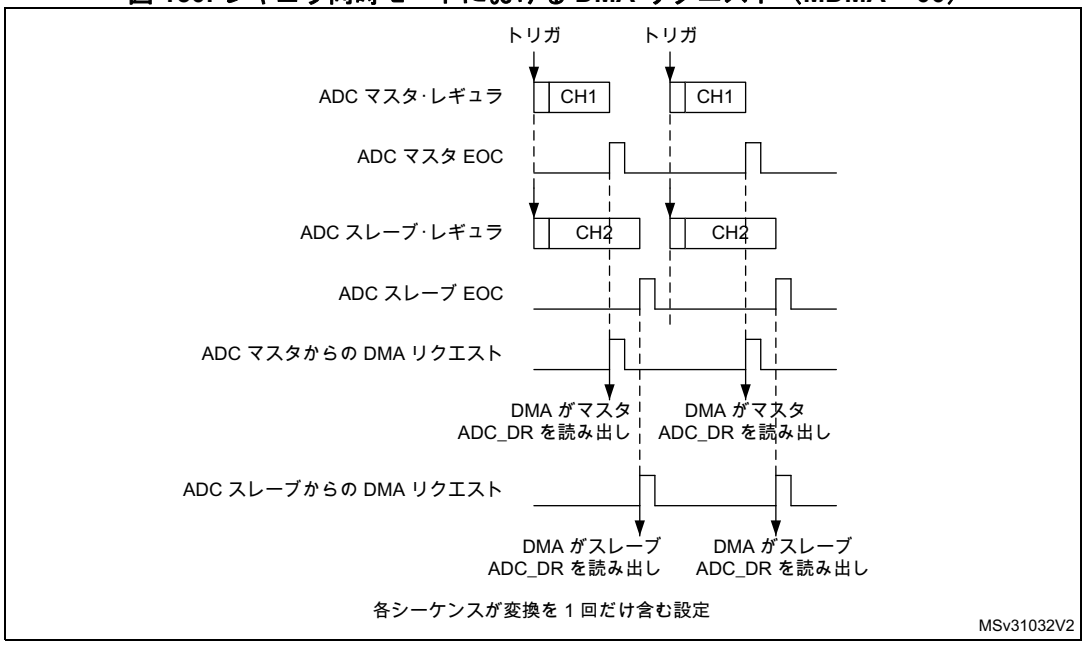
図 149. インジェクトシーケンス CH11、CH12 と 2 つのインタリーブチャネル（CH1、CH2）- ケース 2：スレーブ割込み優先



デュアル ADC モードにおける DMA リクエスト

すべてのデュアル ADC モードでは、シングルモードの場合と同様、データを転送するために 2 つの DMA チャネル（マスタ用 1 つとスレーブ用 1 つ）を使用することができます（図 150：レギュラ同時モードにおける DMA リクエスト（MDMA = 00）を参照）。

図 150. レギュラ同時モードにおける DMA リクエスト（MDMA = 00）



レギュラ同時モードおよびインタリーブ·モードでは 1 つの DMA チャネルを使用して両方のデータを転送することで、DMA チャネルを 1 つ節約できます。そのためには、ADCx_CCR レジスタで MDMA ビットを設定する必要があります。

- **MDMA = 10**：マスタおよびスレーブの両方の EOC イベントが発生するたびに DMA リクエストが 1 つ生成されます。このとき、2 つのデータアイテムが使用可能です。また、32 ビットレジスタ ADCx_CDR には、2 つの ADC で変換されたデータアイテムを表す 2 つのハーフワードが含まれます。スレーブ ADC データは上位ハーフワードを使用し、マスタ ADC データは下位ハーフワードを使用します。
  このモードは、分解能が 10 ビットまたは 12 ビットの場合に、インタリーブモードおよびレギュラ同時モードで使用されます。

  例：

  インタリーブデュアルモード：2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADCx_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

  2 番目の DMA リクエスト：ADCx_CDR[31:0] = SLV_ADC_DR[15:0] | MST_ADC_DR[15:0]

図 151. レギュラ同時モードにおける DMA リクエスト（MDMA = 10）

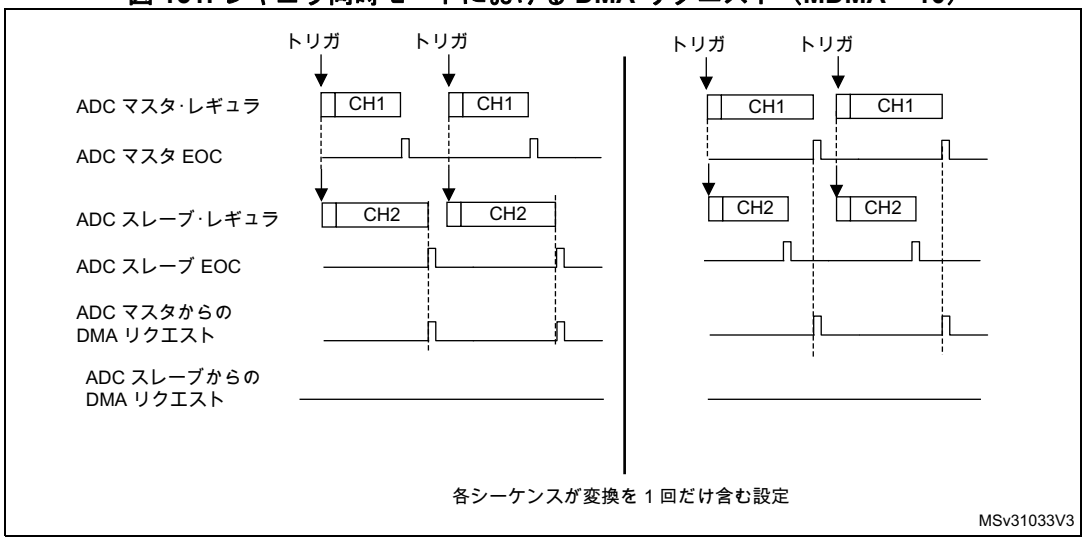
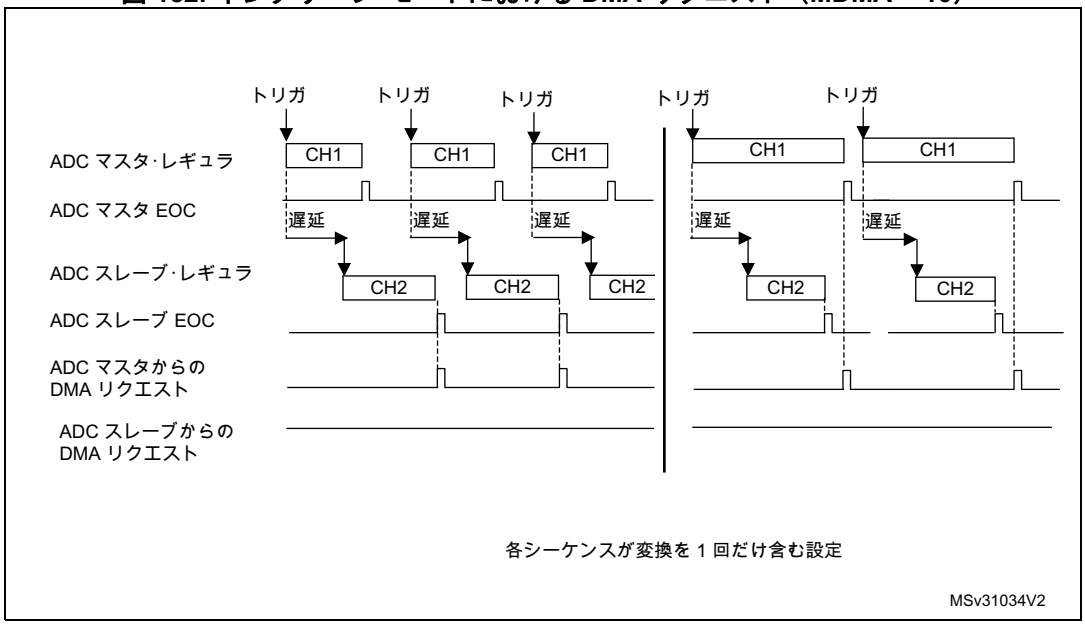


図 152. インタリーブ·モードにおける DMA リクエスト（MDMA = 10）



注： **MDMA モードを使用する場合は、マスタおよびスレーブ変換時間を適切に設定するよう注意する必要があります。これにより、新しい変換が使用可能になる前に DMA リクエストを生成して両方のデータ（マスタ + スレーブ）の読出しを処理できます。**

- **MDMA = 11**：このモードは MDMA = 10 に類似しています。唯一の違いは、DMA リクエストごとに（データ項目は 2 つ使用可能）、A/D コンバータによる 2 つの変換データ項目を表す 2 バイトがハーフワードとして転送される点です。

  このモードは、分解能が 6 ビットまたは 8 ビットでデータが符号なしである場合に、インタリーブモードおよびレギュラ同時モードで使用されます（関連するすべてのチャネルについて、オフセットは無効である必要があります）。

  例：

  インタリーブデュアルモード：2 つのデータ項目が使用可能になるたびに DMA リクエストが 1 つ生成されます。

  最初の DMA リクエスト：ADCx_CDR[15:0] = SLV_ADC_DR[7:0] | MST_ADC_DR[7:0]

  2 番目の DMA リクエスト：ADCx_CDR[15:0] = SLV_ADC_DR[7:0] | MST_ADC_DR[7:0]

オーバーラン検出

デュアル A/D コンバータ・モード（DUAL[4:0] が 00000 以外の場合）で、A/D コンバータのうちの 1 つにオーバーランが検出された場合、RAM に転送されるすべてのデータが有効であることを保証するために、それ以上 DMA リクエストが発行されません（この動作は MDMA の設定にかかわらず発生します）。いずれかの A/D コンバータのデータ・レジスタに有効なデータが格納されていることにより、その A/D コンバータに対応する EOC ビットがセットされたままになることがあります。

MDMA モード選択時の DMA ワンショットモード／DMA サーキュラモード

MDMA モードが選択されている場合（10 または 11）、セクション ：DMA を使用した変換の管理で説明しているように、DMA ワンショット・モード/サーキュラ・モードのどちらかを選択するために ADCx_CCR レジスタの DMACFG ビット も設定する必要があります。（マスタおよびスレーブ ADC_CFGR の DMACFG ビットは無関係です）。

デュアル A/D コンバータ・モードにおける変換の停止

デュアル ADC モードにおける ADC の変換を両方とも停止するには、マスタ ADC の制御ビットに ADSTP/JADSTP をセットする必要があります。デュアル ADC モードには、スレーブ ADC のその他の ADSTP 制御ビットの効果はありません。

両方の ADC が実質的に無効になると、マスタおよびスレーブ ADC の両方のADSTART/JADSTART ビット がハードウェアによってクリアされます。

21.4.31 温度センサ

温度センサを使用して、デバイスの接合温度（Tj）を測定できます。温度センサは、センサの出力電圧をデジタル値に変換するために ADC 入力チャネルに内部接続されます。使用しないときには、センサをパワーダウンモードにすることができます。サポートする温度範囲は -40～125 °C です。

図 153 に、温度センサおよび ADC 間の接続のブロック図を示します。

温度センサの出力電圧は、温度に比例して変化します。このラインのオフセットは、プロセスのばらつきにより、チップごとに異なります（チップ間で最大 45 °C）。

較正されていない内部温度センサは、温度の絶対値の代わりに温度変化を検出するアプリケーションに適しています。温度センサの測定精度を高めるために、生産時に ST によって各デバイスの較正値がシステムメモリに格納されています。

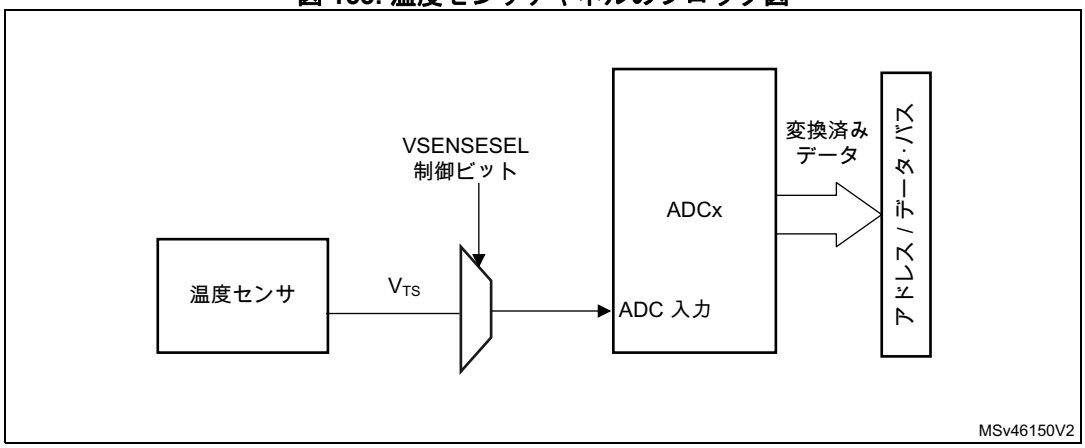
製造プロセス中に、温度センサと内部電圧基準の較正データがシステムメモリ領域に格納されます。ユーザアプリケーションはこれらを読み出して、温度センサまたは内部基準の精度の向上に使用できます（詳しい情報はデータシートを参照してください。）。

温度センサは、センサの出力電圧をデジタル値に変換するために ADC 入力チャネルに内部接続されます。内部温度センサの変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性のセクションを参照してください。

使用しないときには、センサをパワーダウンモードにすることができます。

図 153 に、温度センサのブロック図を示します。

図 153. 温度センサチャネルのブロック図



温度の読出し

温度センサを使用するには：

1. V_{TS} に接続する A/D コンバータ入力チャネルを選択します。
2. 適切なサンプリング時間をプログラムします（デバイス･データシートの電気的特性のセクションを参照）。
3. ADCx_CCR レジスタの VSENSESEL ビットをセットして、温度センサをパワーダウンモードからウェイクアップします。
4. ADC 変換を開始します。
5. A/D コンバータデータレジスタの V_{TS} データを読み出します。
6. 次の式を使用して、実際の温度を計算します。

$$\text{温度}(°C) = \frac{TS_CAL2_TEMP - TS_CAL1_TEMP}{TS_CAL2 - TS_CAL1} \times (TS_DATA - TS_CAL1) + 30\ °C$$

ここで、

- TS_CAL2 は、TS_CAL2_TEMP で得られた温度センサの較正値です。
- TS_CAL1 は、TS_CAL1_TEMP で得られた温度センサの較正値です。
- TS_DATA は、ADC によって変換された実際の温度センサの出力値です。

  TS_CAL1 および TS_CAL2 較正ポイントの詳細については、デバイスデータシートを参照してください。

注： **センサがパワーダウンモードからウェイクアップして、正しいレベルで V_{TS} を出力できるようになるまでには時間がかかります（スタートアップ時間）。ADC にも起動後のスタートアップ時間があるので、遅延を最小にするには、ADEN ビットと VSENSESEL ビットを同時にセットしてください。**

前述の式は、TS_CAL1/TS_CAL2 の測定で使用した値と同じ V_{REF+} 電圧を使用して TS_DATA を測定した場合に適用できます。V_{REF+} が異なる場合は、式を調整する必要があります。たとえば、V_{REF+} = 3.3 V で TS_DATA を測定し、TS_CAL のデータを V_{REF+}= 3.0 V で取得している場合、TS_DATA の値を TS_DATA x (3.3/3.0) に置き換える必要があります。

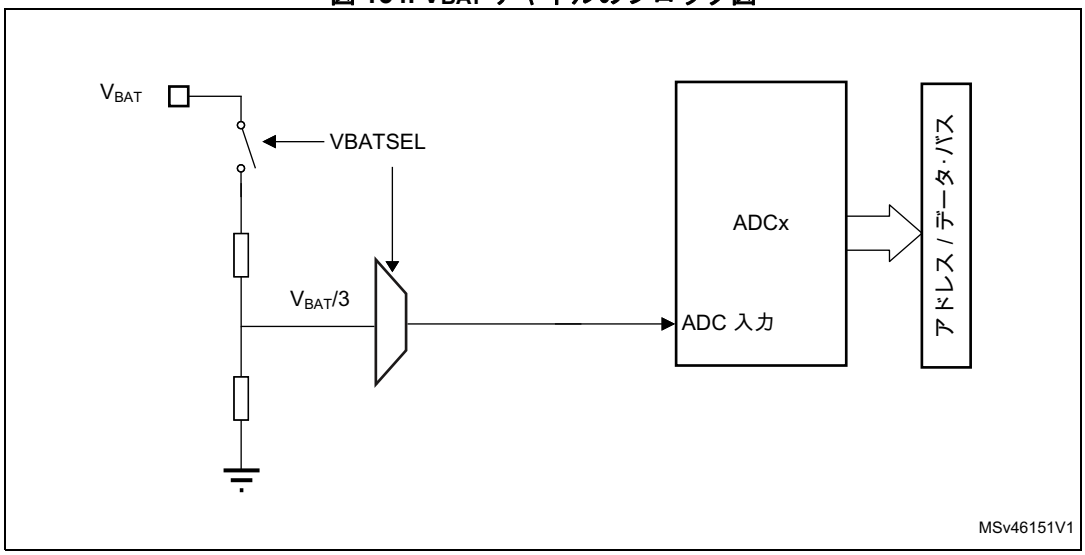
21.4.32 V_{BAT} 電源監視

ADCx_CCR レジスタの VBATSEL ビットは、バッテリ電圧に切り替えるために使用されます。V_{BAT} 電圧は V_{DDA} より高くなることがあるので、ADC の正しい動作を確保するために、V_{BAT} ピンはブリッジ 3 分圧回路に内部接続されています。このブリッジは、VBATSEL ビットがセットされると自動的に有効になり、V_{BAT}/3 を ADC 入力チャネルに接続します。結果として、変換されたデジタル値は V_{BAT} 電圧の 1/3 です。不要なバッテリ消費を避けるには、ADC 変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

V_{BAT} /3 電圧の変換時に適用するサンプリング時間の値については、デバイスデータシートの電気的特性を参照してください。

下の図に、V_{BAT} センシング機能のブロック図を示します。

図 154. V_{BAT} チャネルのブロック図



1. V_{BAT}/3 の内部チャネルの変換を有効にするには、VBATSEL ビットをセットする必要があります。

21.4.33 内部電圧基準の監視

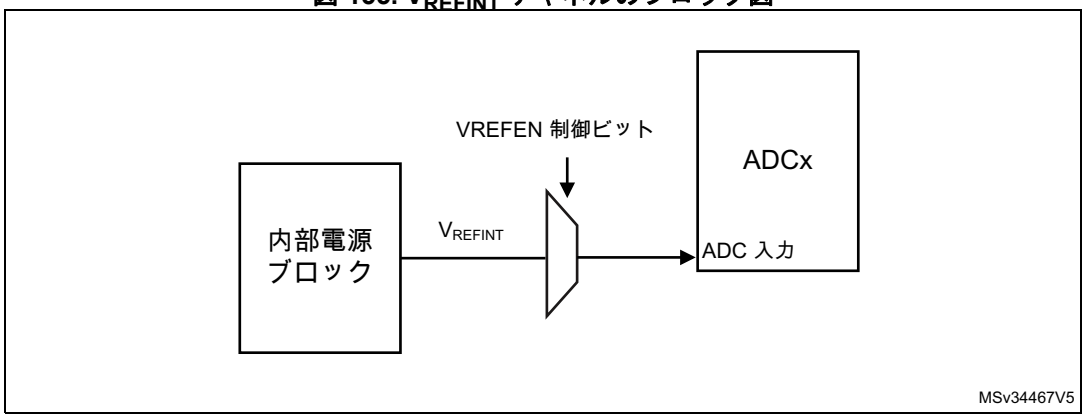
A/D コンバータの V_{REF+} 電圧レベルを評価するための基準点を得るために、内部電圧基準（V_{REFINT}）を監視できます。

内部基準電圧（V_{REFINT}）は ADC1_INP18、ADC3_INP18、ADC4_INP18、および ADC5_INP18 に内部接続されています。

内部電圧基準の変換時に適用するサンプリング時間の値については、製品のデータシートの電気的特性のセクションを参照してください。

図 155 に、V_{REFINT} センシング機能のブロック図を示します。

図 155. V_{REFINT} チャネルのブロック図



1. 内部チャネル（V_{REFINT}）の変換を有効にするには、ADCx_CCR レジスタに VREFEN ビットをセットする必要があります。

内部基準電圧を使用した実際の V_{REF+} 電圧の計算

デバイスに印加される電源電圧は、ばらつきがあり、または正確にはわからないことがあります。V_{DDA} が V_{REF+} に接続されているときには、実際の V_{DDA} 電圧を内蔵内部基準電圧 (V_{REFINT}) を使用して計算することが可能です。V_{REFINT} と、製造プロセスにおいて V_{DDA_Charac} で A/D コンバータによって得られた較正データを使用して、実際の V_{DDA} 電圧レベルを評価できます。

デバイスに印加される実際の V_{REF+} 電圧は、次の式で求められます。

$$V_{REF+} = V_{REF+_Charac} \times VREFINT_CAL / VREFINT_DATA$$

ここで、

- V_{REF+_Charac} は、製造プロセスで V_{REFINT} を使用して特性を評価したときの V_{REF+} の電圧値です。この値は、デバイスのデータシートに指定されています。
- VREFINT_CAL は、V_{REFINT} の較正値です。
- VREFINT_DATA は、A/D コンバータで変換された実際の V_{REFINT} の出力値です。

電源を基準とした A/D コンバータ測定値から絶対電圧値への変換

A/D コンバータは、V_{REF+} と、変換されるチャネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。

ほとんどのアプリケーションでは、V_{DDA} 値は未知であり、A/D コンバータで変換された値は右詰めされます。この場合、この比を V_{DDA} に依存しない電圧に変換する必要があります。

$$V_{CHANNELx} = \frac{V_{REF+}}{FULL_SCALE} \times ADC_DATA$$

V_{REF+} を上記の式で置き換えることによって、絶対電圧値を次の計算式で求めることができます。

$$V_{CHANNELx} = \frac{V_{REF+_Charac} \times VREFINT_CAL \times ADC_DATA}{VREFINT_DATA \times FULL_SCALE}$$

V_{REF+} が既知であり、A/D コンバータで変換された値が右詰めされるアプリケーションでは、次の式を使用して絶対電圧値を求めることができます。

$$V_{CHANNELx} = \frac{V_{REF+}}{FULL_SCALE} \times ADC_DATA$$

ここで、

- V_{REF+_Charac} は、製造プロセスで V_{REFINT} を使用して特性を評価したときの V_{REF+} の電圧値です。
- VREFINT_CAL は、V_{REFINT} の較正値です。
- ADC_DATA は、チャネル x で A/D コンバータによって測定された値です（右詰め）。
- VREFINT_DATA は、A/D コンバータで変換された実際の V_{REFINT} の出力値です。
- FULL_SCALE は、A/D コンバータ出力の最大デジタル値です。たとえば、分解能が 12 ビットの場合は 2^{12} - 1 = 4095、8 ビットの場合は 2^8 - 1 = 255 です。

注： **A/D コンバータ測定が 16 ビット右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。**

21.5 低電力モードでのA/D コンバータ

表 174. 低電力モードが A/D コンバータに与える影響

モード	説明
SLEEP	影響はありません。 DMA リクエストは機能します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。 DMA リクエストは機能します。
STOP 0/STOP 1	A/D コンバータは動作しません。状態は保持されます。 A/D コンバータは静的電流を消費します。消費電力を低減するには、あらかじめペリフェラルを無効化することを推奨します。
STANDBY	A/D コンバータはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

21.6 A/D コンバータの割込み

各 A/D コンバータで、次のような場合に割込みを生成できます。

- ADC パワーアップ後、ADC の準備ができたとき（ADRDY フラグ）
- レギュラグループの任意の変換終了時（EOC フラグ）
- レギュラグループの任意の変換シーケンス終了時（EOS フラグ）
- インジェクトグループの任意の変換終了時（JEOC フラグ）
- インジェクトグループの任意の変換シーケンス終了時（JEOS フラグ）
- アナログウォッチドッグ検出の発生時（AWD1、AWD2、および AWD3 フラグ）
- サンプリングフェーズ終了の発生時（EOSMP フラグ）
- データオーバーランの発生時（OVR フラグ）
- インジェクトシーケンスのコンテキストキューのオーバーフロー時（JQOVF フラグ）

高い柔軟性を実現するため、個別の割込みイネーブルビットを使用できます。

表 175. ADC ごとの ADC 割込み

割込みイベント	イベントフラグ	有効制御ビット
ADC レディ	ADRDY	ADRDYIE
レギュラグループの変換終了	EOC	EOCIE
レギュラ･グループの変換シーケンスの終了	EOS	EOSIE
インジェクト･グループの変換終了	JEOC	JEOCIE
インジェクト･グループの変換シーケンスの終了	JEOS	JEOSIE
アナログ･ウォッチドッグ 1 のステータス･ビットのセット	AWD1	AWD1IE
アナログ･ウォッチドッグ 2 のステータス･ビットのセット	AWD2	AWD2IE
アナログ･ウォッチドッグ 3 のステータス･ビットのセット	AWD3	AWD3IE
サンプリング･フェーズの終了	EOSMP	EOSMPIE
オーバーラン	OVR	OVRIE
インジェクト･コンテキスト･キューのオーバーフロー	JQOVF	JQOVFIE

21.7 ADC レジスタ（ADC ごと）

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

21.7.1 A/D コンバータの割込みおよびステータス・レジスタ（ADC_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
					rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF**：インジェクト・コンテキスト・キューのオーバーフロー

このビットは、コンテキストのインジェクトキューのオーバーフローが発生したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。詳細については、セクション 21.4.21：インジェクト変換のコンテキストのキューを参照してください。

0：インジェクトコンテキストキューのオーバーフローは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトコンテキストキューのオーバーフローが発生しました。

ビット 9 **AWD3**：アナログウォッチドッグ 3 フラグ

このビットは、変換された電圧が、ADC_TR3 レジスタのフィールド LT3[7:0] および HT3[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 3 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 3 イベントが発生しました。

ビット 8 **AWD2**：アナログウォッチドッグ 2 フラグ

このビットは、変換された電圧が、ADC_TR2 レジスタのフィールド LT2[7:0] および HT2[7:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 2 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 2 イベントが発生しました。

ビット 7 **AWD1**：アナログウォッチドッグ 1 フラグ

このビットは、変換された電圧が、ADC_TR1 レジスタのフィールド LT1[11:0] および HT1[11:0] でプログラムされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：アナログウォッチドッグ 1 イベントは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：アナログウォッチドッグ 1 イベントが発生しました。

ビット 6 **JEOS**：インジェクト･チャネルのシーケンス終了フラグ

このビットは、グループ内のすべてのインジェクトチャネル変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：インジェクト変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクト変換シーケンスが完了しました。

ビット 5 **JEOC**：インジェクト･チャネルの変換終了フラグ

このビットは、チャネルの各インジェクト変換の終了時に、新しいデータが対応する ADC_JDRy レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または対応する ADC_JDRy レジスタを読み出すことによってクリアされます。

0：インジェクトチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：インジェクトチャネル変換が完了しました。

ビット 4 **OVR**：A/D コンバータのオーバーラン

このビットは、レギュラ･チャネルでオーバーランが発生したときにハードウェアによってセットされ、EOC フラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで 1 を書き込むことによってクリアされます。

0：オーバーランは発生していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：オーバーランが発生しました。

ビット 3 **EOS**：レギュラ･シーケンス終了フラグ

このビットは、レギュラチャネルシーケンスの変換が終了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0：レギュラ変換シーケンスは完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラ変換シーケンスが完了しました。

ビット 2 **EOC**：変換終了フラグ

このビットは、チャネルの各レギュラ変換の終了時に、新しいデータが ADC_DR レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または ADC_DR レジスタを読み出すことによってクリアされます。

0：レギュラチャネル変換は完了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：レギュラチャネル変換が完了しました。

ビット 1 **EOSMP**：サンプリング終了フラグ

このビットは、任意のチャネル（レギュラチャネルのみ）の変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

0：サンプリングフェーズは終了していません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：サンプリングフェーズは終了しました。

ビット 0 **ADRDY**：A/D コンバータ･レディ

このビットは、A/D コンバータが有効にされた後（ビット ADEN = 1）、A/D コンバータが変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで 1 を書き込むことによってクリアされます。

0：ADC はまだ変換を開始する準備ができていません（またはフラグイベントはソフトウェアによってすでに確認され、クリアされています）。

1：ADC は変換を開始する準備ができました。

21.7.2 ADC 割込み有効レジスタ（ADC_IER）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVFIE**：インジェクトコンテキストキューのオーバーフロー割込みイネーブル

このビットは、インジェクトコンテキストキューのオーバーフローの割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトコンテキストキューのオーバーフロー割込みは無効です。

1：インジェクトコンテキストキューのオーバーフロー割込みは有効です。JQOVF ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **AWD3IE**：アナログウォッチドッグ 3 割込みイネーブル

このビットは、アナログウォッチドッグ 2 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 3 割込みは無効です。

1：アナログウォッチドッグ 3 割込みは有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8 **AWD2IE**：アナログウォッチドッグ 2 割込みイネーブル

このビットは、アナログウォッチドッグ 2 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 2 割込みは無効です。

1：アナログウォッチドッグ 2 割込みは有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 7 **AWD1IE**：アナログウォッチドッグ 1 割込みイネーブル

このビットは、アナログウォッチドッグ 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：アナログウォッチドッグ 1 割込みは無効です。

1：アナログウォッチドッグ 1 割込みは有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 6 **JEOSIE**：インジェクト変換シーケンス終了割込みイネーブル

このビットは、インジェクト変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOS 割込み無効

1：JEOS 割込みは有効です。JEOS ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 5 **JEOCIE**：インジェクト変換終了割込みイネーブル

このビットは、インジェクト変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：JEOC 割込みは無効です。

1：JEOC 割込みは有効です。JEOC ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 4 **OVRIE**：オーバーラン割込みイネーブル

このビットは、レギュラ変換のオーバーラン割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：オーバーラン割込みは無効です。

1：オーバーラン割込みは有効です。OVR ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 3 **EOSIE**：レギュラ変換シーケンス終了割込みイネーブル

このビットは、レギュラ変換シーケンス終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOS 割込み無効

1：EOS 割込みは有効です。EOS ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 2 **EOCIE**：レギュラ変換終了割込みイネーブル

このビットは、レギュラ変換終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOC 割込みは無効です。

1：EOC 割込みは有効です。EOC ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 1 **EOSMPIE**：レギュラ変換のサンプリング終了フラグ割込みイネーブル

このビットは、レギュラ変換のサンプリングフェーズ終了時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：EOSMP 割込みは無効です。

1：EOSMP 割込みは有効です。EOSMP ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 0 **ADRDYIE**：ADC レディ割込みイネーブル

このビットは、ADC レディ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：ADRDY 割込み無効

1：ADRDY 割込みは有効です。ADRDY ビットがセットされると、割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

21.7.3 A/D コンバータ制御レジスタ（ADC_CR）

アドレスオフセット：0x08

リセット値：0x2000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rs	rw	rw	rw												
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
										rs	rs	rs	rs	rs	rs

ビット 31 **ADCAL**：A/D コンバータの較正

このビットは、ADC の較正を開始するためにソフトウェアによってセットされます。最初にADCALDIF ビットをプログラムして、この較正がシングルエンドまたは差動入力モードに適用されるかどうかを確認します。

較正の完了後、ハードウェアによってクリアされます。

0：較正が完了しました。

1：ADC を較正するには、1 を書き込みます。1 として読み出されたときには、較正が実行中であることを意味します。

注： ソフトウェアは、ADEN = 0 のときだけ ADCAL をセットして、較正を起動することができます。ソフトウェアは、ADEN = 1 かつ ADSTART = 0 かつ JADSTART = 0 である（A/D コンバータが有効であり、変換中でない）ときだけ、ADC_CALFACT を書き込むことによって較正係数を更新できます。

ビット 30 **ADCALDIF**：較正の差動モード

このビットは、較正のシングルエンド入力または差動入力モードを設定するために、ソフトウェアによってセット／クリアされます。

0：ADCAL を書き込むと、シングルエンド入力モードで較正が起動されます。

1：ADCAL を書き込むと、差動入力モードで較正が起動されます。

注： ソフトウェアは、A/D コンバータが無効で較正中でないときだけ（ADCAL = 0、JADSTART = 0、JADSTP = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、このビットを書き込むことができます。

ビット 29 **DEEPPWD**：ディープパワーダウンイネーブル

このビットは、ADC をディープパワーダウンモードにするために、ソフトウェアによってセット／クリアされます。

0：A/D コンバータのディープ･パワーダウン･モードは無効です。

1：A/D コンバータのディープ･パワーダウン･モードは有効です（デフォルトのリセット状態）。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、JADSTP = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、このビットを書き込むことができます。

ビット 28 **ADVREGEN**：ADC 電圧レギュレータイネーブル

これらのビットは、ADC 電圧レギュレータを有効にするために、ソフトウェアによってセットされます。較正の起動や ADC の有効化などの操作を実行する前に、まずは ADC 電圧レギュレータを有効にし、ソフトウェアはレギュレータの起動時間まで待つ必要があります。

0：ADC 電圧レギュレータは無効です。

1：ADC 電圧レギュレータは有効です。

ADC 電圧レギュレータの有効化および無効化シーケンスの詳細については、セクション 21.4.6：ADC ディープパワーダウンモード（DEEPPWD）および ADC 電圧レギュレータ（ADVREGEN）を参照してください。

ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、ADEN = 0）、このビットフィールドをプログラムできます。

ビット 27:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **JADSTP**：A/D コンバータのインジェクト変換停止コマンド

このビットは、実行中のインジェクト変換を停止および破棄するためにソフトウェアによってセットされます（JADSTP コマンド）。

変換が効果的に破棄され、ADC インジェクトシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいインジェクト変換の開始（JADSTART コマンド）を受け入れる準備ができます。

0：ADC インジェクト変換停止コマンドは実行中ではありません。

1：実行中の ADC インジェクト変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： **ソフトウェアは、JADSTART = 1 かつ ADDIS = 0である（A/D コンバータが有効であり、最終的にインジェクト変換が実行され、A/D コンバータを無効にする保留中のリクエストがない）ときだけ、JADSTP をセットできます。**

自動インジェクション･モード（JAUTO = 1）で ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

ビット 4 **ADSTP**：A/D コンバータのレギュラ変換停止コマンド

このビットは、実行中のレギュラ変換を停止および破棄するためにソフトウェアによってセットされます（ADSTP コマンド）。

変換が効果的に破棄され、ADC レギュラシーケンスとトリガを再設定できる場合に、ハードウェアによってクリアされます。これで、ADC は新しいレギュラ変換の開始（ADSTART コマンド）を受け入れる準備ができます。

0：ADC レギュラ変換停止コマンドは実行中ではありません。

1：実行中の ADC レギュラ変換を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注： **ソフトウェアは、ADSTART = 1 かつ ADDIS = 0 である（A/D コンバータが有効であり、最終的にレギュラ変換が実行され、A/D コンバータを無効にする保留中のリクエストがない）ときだけ、ADSTP をセットできます。**

自動インジェクション･モード（JAUTO = 1）で ADSTP ビットをセットすると、レギュラ変換とインジェクト変換の両方がアボートされます（JADSTP は使用しないでください）。

デュアル A/D コンバータのレギュラ同時モードとインタリーブ･モードでは、レギュラ変換を停止するためにマスタ A/D コンバータの ADSTP ビットを使用する必要があります。その他の ADSTP ビットはインアクティブです。

ビット 3 **JADSTART**：ADC インジェクト変換開始

このビットは、インジェクトチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット JEXTEN[1:0] に応じて、変換はただちに開始されるか（ソフトウェア･トリガ設定）またはインジェクト･ハードウェア･トリガ･イベントが発生（ハードウェア･トリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

– シングル変換モードでソフトウェア･トリガが選択された場合（JEXTSEL = *0*x0）：インジェクト変換シーケンス終了（JEOS）フラグのアサート時。

– すべての場合：JADSTP コマンドの実行後、JADSTP ビットのハードウェアによるクリアと同時。

0：ADC インジェクト変換は実行中ではありません。

1：インジェクト変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にインジェクトチャネルが変換されることを意味します。

注： ソフトウェアは、ADEN = 1 かつ ADDIS =0 である（A/D コンバータが有効であり、A/D コンバータを無効にする保留中のリクエストがない）ときだけ、JADSTART をセットできます。

自動インジェクション･モード（JAUTO = 1）では、ADSTART ビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 2 **ADSTART**：ADC レギュラ変換開始

このビットは、レギュラチャネルの ADC 変換を開始するためにソフトウェアによってセットされます。設定ビット EXTEN[1:0] に応じて、変換はただちに開始されるか（ソフトウェア･トリガ設定）またはレギュラ･ハードウェア･トリガ･イベントが発生（ハードウェア･トリガ設定）したときに開始されます。

次のときに、ハードウェアによってクリアされます。

– シングル変換モードでソフトウェア･トリガが選択された場合（EXTSEL = *0*x0）：レギュラ変換シーケンス終了（EOS）フラグのアサート時。

– すべての場合：ADSTP コマンドの実行後、ADSTP ビットのハードウェアによるクリアと同時。

0：ADC レギュラ変換は実行中ではありません。

1：レギュラ変換を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、最終的にレギュラチャネルが変換されることを意味します。

注： ソフトウェアは、ADEN = 1 かつ ADDIS = 0 である（A/D コンバータが有効であり、A/D コンバータを無効にする保留中のリクエストがない）ときだけ、ADSTART をセットできます。

自動インジェクション･モード（JAUTO = 1）では、ADSTART ビットのセットによってレギュラおよび自動インジェクト変換を開始します（JADSTART はクリアされたままでなければなりません）。

ビット 1 **ADDIS**：A/D コンバータ無効化コマンド

このビットは、ADC を無効にして（ADDIS コマンド）、パワーダウン状態（OFF 状態）にするためにソフトウェアによってセットされます。

ADC が効果的に無効化されると、ハードウェアによってクリアされます（ADEN もこの時点でハードウェアによってクリアされます）。

0：ADDIS コマンドは実行中ではありません。

1：ADC を無効にするには、1 を書き込みます。1 として読み出されたときには、ADDIS コマンドが実行中であることを意味します。

注： ソフトウェアは、ADEN = 1 で ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、ADDIS をセットできます。

ビット 0 **ADEN**：A/D コンバータ･イネーブル制御

このビットは、ADC を有効にするために、ソフトウェアによってセットされます。フラグ ADRDY がセットされていれば、A/D コンバータは実質的に動作準備が完了しています。

ADDIS コマンドの実行後、A/D コンバータが無効になるとハードウェアによってクリアされます。

0：ADC は無効です（OFF 状態）。

1：ADC を有効にするには、1 を書き込みます。

注： ソフトウェアは、1 に設定する必要がある ADVREGEN ビットを除き、ADC_CR レジスタのすべてのビットが 0 のとき（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、ADEN = 0）だけ、ADEN をセットできます（また、ソフトウェアは、電圧レギュレータの起動時間まで待っていなければなりません）。

21.7.4 A/D コンバータ設定レジスタ（ADC_CFGR）

アドレスオフセット：0x0C

リセット値：0x8000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JQDIS	AWD1CH[4:0]					JAUTO	JAWD1 EN	AWD1 EN	AWD1S GL	JQM	JDISCE N	DISCNUM[2:0]			DISC EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALIGN	AUT DLY	CONT	OVR MOD	EXTEN[1:0]		EXTSEL 4	EXTSEL 3	EXTSEL 2	EXTSEL 1	EXTSEL 0	RES[1:0]		Res.	DMA CFG	DMA EN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット 31 **JQDIS**：インジェクト･キュー･ディスエーブル

これらのビットは、インジェクトキューメカニズムを無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトキューは有効です。

1：インジェクトキューは無効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。

JQDIS ビットをセットまたはリセットすることで、インジェクト･キューは一掃され、JSQR レジスタがクリアされます。

ビット 30:26 **AWD1CH[4:0]**：アナログウォッチドッグ 1 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャネルを選択します。

00000：A/D コンバータのアナログ入力チャネル 0 を AWD1 で監視（ADC1 のみ使用可能）

00001：A/D コンバータのアナログ入力チャネル 1 を AWD1 で監視

.....

10010：A/D コンバータのアナログ入力チャネル 18 を AWD1 で監視

その他。予約済み。使用できません。

注： 一部のチャネルは物理的に未接続です。それらに対応する AWD1CH[4:0] の設定はリセット値のままにしておいてください。

AWD1CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 25 **JAUTO**：自動インジェクト･グループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：自動インジェクトグループ変換は無効です。

1：自動インジェクトグループ変換は有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（レギュラ変換とインジェクト変換がどちらも実行中でない）ときのみ、このビットに書き込むことができます。

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ A/D コンバータの JAUTO ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの JAUTO ビット と等しくなります。

ビット 24 **JAWD1EN**：アナログウォッチドッグ 1 有効（インジェクトチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルのアナログウォッチドッグ 1 は無効です。

1：インジェクトチャネルのアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 23 **AWD1EN**：アナログウォッチドッグ 1 有効（レギュラチャネル）

このビットは、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルのアナログウォッチドッグ 1 は無効です。

1：レギュラチャネルのアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 22 **AWD1SGL**：ウォッチドッグ 1 を単一チャネルまたはすべてのチャネルで有効にします。

このビットは、AWD1CH[4:0] ビットによって指定されたチャネルまたはすべてのチャネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。

0：すべてのチャネルでアナログウォッチドッグ 1 は有効です。

1：単一チャネルでアナログウォッチドッグ 1 は有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 21 **JQM**：JSQR キュー･モード

このビットは、ソフトウェアによってセット／クリアされます。

空のキューの管理方法について定義します。

0：JSQR モード 0：キューは空にならず、JSQR に最後に書き込んだ設定を保持します。

1：JSQR モード 1：キューは空である場合もありますが、このとき、インジェクトシーケンスのソフトウェア／ハードウェアトリガは、最後の有効なインジェクトシーケンスの完了後、すぐに両方とも内部的に無効化されます。

詳細については、セクション 21.4.21：インジェクト変換のコンテキストのキューを参照してください。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ A/D コンバータの JQM ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの JQM ビットと等しくなります。

ビット 20 **JDISCEN**：インジェクトチャネルの不連続モード

このビットは、グループのインジェクトチャネルに対する不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトチャネルの不連続モードは無効です。

1：インジェクトチャネルの不連続モードは有効です。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、このビットに書き込むことができます。

自動インジェクト･モードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

デュアルモードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ ADC の JDISCEN ビットへの書込みができなくなり､その内容はマスタ ADC の JDISCEN ビットと等しくなります。

ビット 19:17 **DISCNUM[2:0]**：不連続モードチャネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるレギュラチャネルの数を定義するために、ソフトウェアによって書き込まれます。

000：1 チャネル
001：2 チャネル
.....
111：8 チャネル

注： **ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。**

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビット が 0 以外）である場合、スレーブ A/D コンバータの DISCNUM[2:0] ビット への書込みができなくなり、その内容はマスタ A/D コンバータの DISCNUM[2:0] ビットと等しくなります。

ビット 16 **DISCEN**：レギュラ･チャネルの不連続モード

このビットは、レギュラチャネルの不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラチャネルの不連続モードは無効です。
1：レギュラチャネルの不連続モードは有効です。

注： **不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1 と CONT = 1 の両方を設定することは禁じられています。**

自動インジェクト･モードと不連続モードの両方を同時に使用することはできません。JAUTO のセット時には、ビット DISCEN と JDISCEN はクリアされたままでなければなりません。

ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ A/D コンバータの DISCEN ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの DISCEN ビット と等しくなります。

ビット 15 **ALIGN**：データの配置

このビットは、右詰めまたは左詰めを選択するために、ソフトウェアによってセット／クリアされます。セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）を参照してください。

0：右詰め
1：左詰め

注： **ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。**

ビット 14 **AUTDLY**：遅延変換モード

このビットは、自動遅延変換モードを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0：自動遅延変換モードはオフです。
1：自動遅延変換モードはオンです。

注： **ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。**

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ A/D コンバータの AUTDLY ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの AUTDLY ビット と等しくなります。

ビット 13 **CONT**：レギュラ変換のシングル/連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的にレギュラ変換が行われます。

0：シングル変換モード

1：連続変換モード

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN = 1 と CONT = 1 の両方を設定することは禁じられています。

ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル･モードが有効（ADCx_CCR レジスタの DUAL ビットが 0 以外）である場合、スレーブ A/D コンバータの CONT ビットへの書込みができなくなり、その内容はマスタ A/D コンバータの CONT ビットと等しくなります。

ビット 12 **OVRMOD**：オーバーラン･モード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0：オーバーランが検出されたとき、ADC_DR レジスタの古いデータが保存されます。

1：オーバーランが検出されたとき、ADC_DR レジスタは最後の変換結果で上書きされます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 11:10 **EXTEN[1:0]**：レギュラチャネルの外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択しレギュラグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00：ハードウェアトリガ検出は無効です（変換はソフトウェアによって起動できます）。

01：立ち上がりエッジでハードウェアトリガを検出します。

10：立ち下がりエッジでハードウェアトリガを検出します。

11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 9:5 **EXTSEL[4:0]**：レギュラグループの外部トリガ選択

これらのビットは、レギュラグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0

00001：イベント 1

00010：イベント 2

00011：イベント 3

00100：イベント 4

00101：イベント 5

00110：イベント 6

00111：イベント 7

.....

11111：イベント 31

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 4:3 **RES[1:0]**：データ分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

00：12 ビット

01：10 ビット

10：8 ビット

11：6 ビット

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DMACFG**：ダイレクトメモリアクセス設定

このビットは、2 つの DMA 動作モードのいずれかを選択するためにソフトウェアによってセットまたはクリアされ、DMAEN = *1* のときのみ有効です。

0：DMA ワンショット･モードが選択されます。

1：DMA サーキュラ･モードが選択されます。

詳細については、セクション ：DMA を使用した変換の管理を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル A/D コンバータ･モードでは、このビットは無関係で、ADCx_CCR レジスタの制御ビット DMACFG に置き換えられます。

ビット 0 **DMAEN**：ダイレクト･メモリ･アクセス有効化

このビットは、DMA リクエストの生成を有効にするために、ソフトウェアによってセット／クリアされます。これにより、DMA を使用して変換データを自動的に管理できます。詳細については、セクション ：DMA を使用した変換の管理を参照してください。

0：DMA は無効です。

1：DMA は有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

デュアル A/D コンバータモードでは、このビットは無関係で、ADCx_CCR レジスタの制御ビット MDMA[1:0] に置き換えられます。

21.7.5 ADC 設定レジスタ 2（ADC_CFGR2）

アドレス･オフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	SMPTRIG	BULB	SWTRIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GCOMP
				rw	rw	rw									rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	ROVSM	TROVS	OVSS[3:0]				OVSR[2:0]			JOVSE	ROVSE
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **SMPTRIG**：サンプリング時間制御トリガモード

このビットは、サンプリング時間制御トリガモードを有効にするために、ソフトウェアによってセット／クリアされます。

0：サンプリング時間制御トリガ･モードは無効です。

1：サンプリング時間制御トリガ･モードは有効です。

サンプリング時間はトリガの立ち上がりエッジで開始し、変換はトリガの立ち下がりエッジで開始します。

EXTEN[1:0] ビットは 01 に設定する必要があります。SMPTRIG ビットがセットされている場合、BULB ビットをセットすることはできません。

EXTEN[1:0] ビットが 00 に設定されているとき、サンプリングを開始するには SWTRIG ビットをセットし、変換を開始するには SWTRIG ビットをクリアします。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 26 **BULB**：バルブ･サンプリング･モード

このビットは、バルブサンプリングモードを有効にするために、ソフトウェアによってセット／クリアされます。

0：バルブ･サンプリング･モードは無効です。

1：バルブ･サンプリング･モードは有効です。サンプリング期間は前の変換終了直後に開始されます。

BULB ビットがセットされている場合、SAMPTRIG ビットをセットすることはできません。

最初の ADC 変換は SMPx ビットで指定されたサンプリング時間で実行されます。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 25 **SWTRIG**：サンプリング時間制御トリガ･モードのソフトウェア･トリガ･ビット

このビットは、バルブサンプリングモードを有効にするために、ソフトウェアによってセット／クリアされます。

0：ソフトウェア･トリガはサンプリング時間制御トリガ･モードの変換を開始します。

1：ソフトウェア･トリガはサンプリング時間制御トリガ･モードのサンプリングを開始します。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 24:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **GCOMP**：ゲイン補正モード

このビットは、ゲイン補正モードを有効にするために、ソフトウェアによってセット／クリアされます。

0：A/D コンバータのレギュラ動作モード

1：ゲイン補正は有効で、すべてのチャネルに適用されます。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **ROVSM**：レギュラ･オーバーサンプリング･モード

このビットは、レギュラオーバーサンプリングモードを選択するために、ソフトウェアによってセット／クリアされます。

0：連続モード：インジェクト変換がトリガされると、オーバーサンプリングが一時的に停止され、インジェクションシーケンス後に続行されます（オーバーサンプリングバッファはインジェクションシーケンス中、維持されます）。

1：再開モード：インジェクト変換がトリガされると、現在のオーバーサンプリングが中止され、インジェクションシーケンス後に最初から再開されます（オーバーサンプリングバッファはインジェクションシーケンス開始によって 0 になります）。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 9 **TROVS**：トリガ･レギュラ･オーバーサンプリング

このビットは、トリガオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：チャネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャネルのオーバーサンプリング変換ごとに新しいトリガが必要です。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8:5 **OVSS[3:0]**：オーバーサンプリング･シフト

このビットフィールドは、そのままのオーバーサンプリング結果に適用する右シフトを定義するために、ソフトウェアによってセット／クリアされます。

0000：シフトなし
0001：1 ビット･シフト
0010：2 ビット･シフト
0011：3 ビット･シフト
0100：4 ビット･シフト
0101：5 ビット･シフト
0110：6 ビット･シフト
0111：7 ビット･シフト
1000：8 ビット･シフト
その他のコードは予約済み

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 4:2 **OVSR[2:0]**：オーバーサンプリング比

このビットフィールドは、オーバーサンプリング比を定義するために、ソフトウェアによってセット／クリアされます。

000：2x
001：4x
010：8x
011：16x
100：32x
101：64x
110：128x
111：256x

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 1 **JOVSE**：インジェクト･オーバーサンプリング有効化

このビットは、インジェクトオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：インジェクトオーバーサンプリングは無効です。
1：インジェクトオーバーサンプリングは有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 0 **ROVSE**：レギュラ･オーバーサンプリング有効化

このビットは、レギュラオーバーサンプリングを有効にするために、ソフトウェアによってセット／クリアされます。

0：レギュラオーバーサンプリングは無効です。
1：レギュラオーバーサンプリングは有効です。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

21.7.6 A/D コンバータ･サンプル時間レジスタ 1（ADC_SMPR1）

アドレス･オフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SMPPLUS	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:1]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP5[0]	SMP4[2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **SMPPLUS**：サンプリング時間への 1 クロックサイクルの追加

1：2.5 ADC クロックサイクルのサンプリング時間が ADC_SMPR1 および ADC_SMPR2 レジスタに対して 3.5 ADC クロックサイクルになります。

0：サンプリング時間が、2.5 ADC クロックサイクルのままにセットされます。

進行中の変換がないことを確認するために、ソフトウェアは ADSTART= 0 および JADSTART= 0 の場合のみ、このビットに書き込めます。

ビット 30 予約済みであり、リセット値に保持する必要があります。

ビット 29:0 **SMPx[2:0]**：チャネル x サンプリング時間選択（x = 9 ~ 0）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000：2.5 ADC クロックサイクル
001：6.5 ADC クロックサイクル
010：12.5 ADC クロックサイクル
011：24.5 ADC クロックサイクル
100：47.5 ADC クロックサイクル
101：92.5 ADC クロックサイクル
110：247.5 ADC クロックサイクル
111：640.5 ADC クロックサイクル

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

一部のチャネルは物理的に未接続です。それらに対応する SMPx[2:0] の設定はリセット値のままにしておいてください。

21.7.7 A/D コンバータ･サンプル時間レジスタ 2（ADC_SMPR2）

アドレス･オフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:1]	
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	SMP14[2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:0 **SMPx[2:0]**：チャネル x サンプリング時間選択（x = 18 ~ 10）

これらのビットは、各チャネルのサンプリング時間を個別に選択するために、ソフトウェアによって書き込まれます。サンプリングサイクル中は、チャネル選択ビットを変更することはできません。

000：2.5 ADC クロックサイクル
001：6.5 ADC クロックサイクル
010：12.5 ADC クロックサイクル
011：24.5 ADC クロックサイクル
100：47.5 ADC クロックサイクル
101：92.5 ADC クロックサイクル
110：247.5 ADC クロックサイクル
111：640.5 ADC クロックサイクル

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

一部のチャネルは物理的に未接続です。それらに対応する SMPx[2:0] の設定はリセット値のままにしておいてください。

21.7.8 A/D コンバータ･ウォッチドッグ閾値レジスタ 1（ADC_TR1）

アドレス･オフセット：0x20

リセット値：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	HT1 [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AWDFILT[2:0]			LT1[11:0]											
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **HT1[11:0]**：アナログウォッチドッグ 1 高閾値

これらのビットは、アナログウォッチドッグ 1 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 15: 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **AWDFILT**：アナログウォッチドッグフィルタパラメータ

このビットは、ソフトウェアによってセット／クリアされます。

000：フィルタリングなし

001：2 個の連続した検出によって AWDx フラグまたは割込みが生成されます。

.....

111：8 個の連続した検出によって AWDx フラグまたは割込みが生成されます。

注： ソフトウェアは、ADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 11:0 **LT1[11:0]**：アナログウォッチドッグ 1 低閾値

これらのビットは、アナログウォッチドッグ 1 の低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

21.7.9 A/D コンバータ･ウォッチドッグ閾値レジスタ 2（ADC_TR2）

アドレス･オフセット：0x24

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **HT2[7:0]**：アナログウォッチドッグ 2 高閾値

これらのビットは、アナログウォッチドッグ 2 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **LT2[7:0]**：アナログウォッチドッグ 2 低閾値

これらのビットは、アナログウォッチドッグ 2 の低閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

21.7.10 A/D コンバータ･ウォッチドッグ閾値レジスタ 3（ADC_TR3）

アドレス･オフセット：0x28

リセット値：0x00FF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT3 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT3 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **HT3[7:0]**：アナログウォッチドッグ 3 高閾値

これらのビットは、アナログウォッチドッグ 3 の高閾値を定義するために、ソフトウェアによって書き込まれます。

セクション 21.4.28：アナログ･ウィンドウ･ウォッチドッグ（AWD1EN、JAWD1EN、AWD1SGL、AWD1CH、AWD2CH、AWD3CH、AWD_HTx、AWD_LTx、AWDx）を参照してください。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **LT3[7:0]**：アナログウォッチドッグ 3 低閾値

これらのビットは、アナログウォッチドッグ 3 の低閾値を定義するために、ソフトウェアによって書き込まれます。

このウォッチドッグでは、LT3 の 8 ビットを変換データの 最上位の8ビットと比較します。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

21.7.11 A/D コンバータ･レギュラ･シーケンス･レジスタ 1（ADC_SQR1）

アドレス･オフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ4 [4:0]					Res.	SQ3 [4:0]					Res.	SQ2[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ2 [3:0]				Res.	SQ1 [4:0]					Res.	Res.	L[3:0]			
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ4[4:0]**：レギュラシーケンスの 4 番目の変換

これらのビットには、レギュラ変換シーケンス内の 4 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ3[4:0]**：レギュラシーケンスの 3 番目の変換

これらのビットには、レギュラ変換シーケンス内の 3 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ2[4:0]**：レギュラシーケンスの 2 番目の変換

これらのビットには、レギュラ変換シーケンス内の 2 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ1[4:0]**：レギュラシーケンスの 1 番目の変換

これらのビットには、レギュラ変換シーケンス内の 1 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **L[3:0]**：レギュラチャネルシーケンス長

これらのビットは、レギュラチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

0000：1 変換
0001：2 変換
.....
1111：16 変換

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

注： 一部のチャネルは物理的に未接続です。これらを変換用に選択しないでください。

21.7.12 A/D コンバータ・レギュラ・シーケンス・レジスタ 2（ADC_SQR2）

アドレス・オフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ9 [4:0]					Res.	SQ8 [4:0]					Res.	SQ7[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ7 [3:0]				Res.	SQ6 [4:0]					Res.	SQ5 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ9[4:0]**：レギュラシーケンスの 9 番目の変換

これらのビットには、レギュラ変換シーケンス内の 9 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ8[4:0]**：レギュラシーケンスの 8 番目の変換

これらのビットには、レギュラ変換シーケンス内の 8 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ7[4:0]**：レギュラシーケンスの 7 番目の変換

これらのビットには、レギュラ変換シーケンス内の 7 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ6[4:0]**：レギュラシーケンスの 6 番目の変換

これらのビットには、レギュラ変換シーケンス内の 6 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ5[4:0]**：レギュラシーケンスの 5 番目の変換

これらのビットには、レギュラ変換シーケンス内の 5 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

注： 一部のチャネルは物理的に未接続です。これらを変換用に選択しないでください。

21.7.13 A/D コンバータ・レギュラ・シーケンス・レジスタ 3（ADC_SQR3）

アドレス・オフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	SQ14 [4:0]					Res.	SQ13 [4:0]					Res.	SQ12[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ12 [3:0]				Res.	SQ11 [4:0]					Res.	SQ10 [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **SQ14[4:0]**：レギュラシーケンスの 14 番目の変換

これらのビットには、レギュラ変換シーケンス内の 14 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **SQ13[4:0]**：レギュラシーケンスの 13 番目の変換

これらのビットには、レギュラ変換シーケンス内の 13 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **SQ12[4:0]**：レギュラシーケンスの 12 番目の変換

これらのビットには、レギュラ変換シーケンス内の 12 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ11[4:0]**：レギュラシーケンスの 11 番目の変換

これらのビットには、レギュラ変換シーケンス内の 11 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ10[4:0]**：レギュラシーケンスの 10 番目の変換

これらのビットには、レギュラ変換シーケンス内の 10 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

注： 一部のチャネルは物理的に未接続です。これらを変換用に選択しないでください。

21.7.14 A/D コンバータ･レギュラ･シーケンス･レジスタ 4（ADC_SQR4）

アドレス･オフセット：0x3C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
					rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **SQ16[4:0]**：レギュラシーケンスの 16 番目の変換

これらのビットには、レギュラ変換シーケンス内の 16 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **SQ15[4:0]**：レギュラシーケンスの 15 番目の変換

これらのビットには、レギュラ変換シーケンス内の 15 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

注： 一部のチャネルは物理的に未接続です。これらを変換用に選択しないでください。

21.7.15 A/D コンバータ・レギュラ・データ・レジスタ（ADC_DR）

アドレスオフセット：0x40

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **RDATA[15:0]**：レギュラ変換済みデータ

これらのビットは読出し専用です。これらは、最後の変換されたレギュラチャネルの変換結果を含んでいます。セクション 21.4.26：データ管理に示すように、データは左詰めまたは右詰めされています。

21.7.16 ADC インジェクトシーケンスレジスタ（ADC_JSQR）

アドレスオフセット：0x4C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
JSQ4[4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:1]			
rw	rw	rw	rw	rw			rw	rw	rw	rw		rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ20	Res.	JSQ1 [4:0]					JEXTEN[1:0]		JEXTSEL[4:0]					JL[1:0]	
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27 **JSQ4[4:0]**：インジェクトシーケンスの 4 番目の変換

これらのビットには、インジェクト変換シーケンス内の 4 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **JSQ3[4:0]**：インジェクトシーケンスの 3 番目の変換

これらのビットには、インジェクト変換シーケンス内の 3 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 20 予約済みであり、リセット値に保持する必要があります。

ビット 19:15 **JSQ2[4:0]**：インジェクトシーケンスの 2 番目の変換

これらのビットには、インジェクト変換シーケンス内の 2 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:9 **JSQ1[4:0]**：インジェクトシーケンスの 1 番目の変換

これらのビットには、インジェクト変換シーケンス内の 1 番目に割り当てるチャネル番号（0 ~ 18）がソフトウェアによって書き込まれます。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 8:7 **JEXTEN[1:0]**：インジェクト･チャネルの外部トリガの有効化および極性選択

これらのビットは、外部トリガ極性を選択しインジェクトグループのトリガを有効にするために、ソフトウェアによってセット／クリアされます。

00 : JQDIS = 0（キュー有効）の場合、ハードウェアおよびソフトウェア･トリガ検出は無効になります。
00 : また、JQDIS = 1（キュー無効）の場合、ハードウェア･トリガ検出は無効です（変換はソフトウェアによって起動できます）。
01：立ち上がりエッジでハードウェアトリガを検出します。
10：立ち下がりエッジでハードウェアトリガを検出します。
11：立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

JQM = 1 で、コンテキストのキューが空になると、インジェクト･シーケンスのソフトウェア･トリガとハードウェア･トリガは両方とも内部的に無効化されます（セクション 21.4.21 : インジェクト変換のコンテキストのキューを参照）。

ビット 6:2 **JEXTSEL[4:0]**：インジェクトグループの外部トリガ選択

これらのビットは、インジェクトグループの変換の開始をトリガするために使用される外部イベントを選択します。

00000：イベント 0
00001：イベント 1
00010：イベント 2
00011：イベント 3
00100：イベント 4
00101：イベント 5
00110：イベント 6
00111：イベント 7
.....
11111：イベント 31

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 1:0 **JL[1:0]**：インジェクトチャネルシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換数を定義するために、ソフトウェアによって書き込まれます。

00：1 変換
01：2 変換
10：3 変換
11：4 変換

注： ソフトウェアは、JADSTART = 0 である（インジェクト変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

注：　一部のチャネルは物理的に未接続です。これらを変換用に選択しないでください。

21.7.17 A/D コンバータ･オフセット y レジスタ（ADC_OFRy）

アドレスオフセット：0x60 + 0x04 * (y-1)（y = 1 から 4）

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OFFSET_EN	OFFSET_CH[4:0]					SATEN	OFFSETPOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw								

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

ビット 31 **OFFSET_EN**：オフセット y イネーブル

このビットは、OFFSETy[11:0] ビットにプログラムされたオフセットを有効または無効にするために、ソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 30:26 **OFFSET_CH[4:0]**：データ･オフセット y のチャネル選択

このビットは、OFFSETy[11:0] ビットにプログラムされたオフセットが適用されるチャネルを定義するために、ソフトウェアによって書き込まれます。

注：　ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

一部のチャネルは物理的に未接続です。これらをデータ･オフセット y 用に選択しないでください。

ビット 25 **SATEN**：飽和イネーブル

このビットは、オフセット機能のために 0x000 および 0xFFF で飽和を有効にするために、ソフトウェアによってセット／クリアされます。

0：飽和制御はなしで、オフセット結果は符号付きにできます。

1：飽和は有効であり、オフセット結果は符号なしで、0x000 および 0xFFF で飽和します。

注：　ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 24 **OFFSETPOS**：正のオフセット

このビットは、正のオフセットを有効にするために、ソフトウェアによってセット／クリアされます。

0：負のオフセット

1：正のオフセット

注：　ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 23:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **OFFSET[11:0]**：OFFSETy_CH[4:0] ビットにプログラムされたチャネルのデータ･オフセット y

これらのビットは、チャネル（レギュラまたはインジェクトのどちらか）を変換するときに元の変換データから引かれるオフセット y を定義するために、ソフトウェアによって書き込まれます。データオフセット y を適用するチャネルは、ビット OFFSETy_CH[4:0] ビットにプログラムする必要があります。変換結果は、ADC_DR レジスタ（レギュラ変換）または ADC_JDRyi レジスタ（インジェクト変換）から読み出すことができます。

注： ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

複数のオフセット（OFFSETy）が同じチャネルを指している場合、減算時には最小の x を持つオフセットのみが考慮されます。

例：OFFSET1_CH[4:0]=4 かつ OFFSET2_CH[4:0]=4 である場合、これはチャネル 4 を変換するときに減算される OFFSET1[11:0] です。

21.7.18 ADC インジェクトチャネル y データレジスタ（ADC_JDRy）

アドレスオフセット：0x80 + 0x04 * (y - 1)（y = 1 から 4）

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **JDATA[15:0]**：インジェクトデータ

これらのビットは読出し専用です。これらのビットには、インジェクトチャネル y の変換結果が格納されます。セクション 21.4.26：データ管理に示すように、データは左詰めまたは右詰めされています。

21.7.19 A/D コンバータ･アナログ･ウォッチドッグ 2 設定レジスタ（ADC_AWD2CR）

アドレス･オフセット：0xA0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH[18:16]		
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD2CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **AWD2CH[18:0]**：アナログウォッチドッグ 2 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 2 によって保護される入力チャネルを有効化して選択します。

AWD2CH[i] = 0：ADC アナログ入力チャネル i は AWD2 によって監視されません。

AWD2CH[i] = 1：ADC アナログ入力チャネル i は AWD2 によって監視されます。

AWD2CH[18:0] = 000..0 である場合、アナログ･ウォッチドッグ 2 は無効になります。

注： AWD2CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

一部のチャネルは物理的に未接続です。これらをアナログ･ウォッチドッグ用に選択しないでください。

21.7.20 A/D コンバータ･アナログ･ウォッチドッグ 3 設定レジスタ（ADC_AWD3CR）

アドレス･オフセット：0xA4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH[18:16]		
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWD3CH [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **AWD3CH[18:0]**：アナログウォッチドッグ 3 チャネル選択

これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグ 3 によって保護される入力チャネルを有効化して選択します。

AWD3CH[i] = 0：ADC アナログ入力チャネル i は AWD3 によって監視されません。

AWD3CH[i] = 1：ADC アナログ入力チャネル i は AWD3 によって監視されます。

AWD3CH[18:0] = 000..0 である場合、アナログ･ウォッチドッグ 3 は無効になります。

注： AWD3CH で選択したチャネルは、SQRi または JSQRi レジスタでも選択する必要があります。

ソフトウェアは、ADSTART = 0 かつ JADSTART = 0 である（変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

一部のチャネルは物理的に未接続です。これらをアナログ･ウォッチドッグ用に選択しないでください。

21.7.21 A/D コンバータ差動モード選択レジスタ（ADC_DIFSEL）

アドレスオフセット：0xB0

リセット値：0x0000 0000

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

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

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:0 **DIFSEL[18:0]**：チャネル（18 ~ 0）の差動モード

これらのビットは、ソフトウェアによってセットまたはクリアされます。これにより、チャネルをシングルエンド･モードまたは差動モードに設定できます。

DIFSEL[i] = 0：A/D コンバータのアナログ入力チャネルはシングルエンド･モードに設定されます。

DIFSEL[i] = 1：ADC アナログ入力チャネル i は差動モードで設定されます。

注： シングルエンドの I/O ポートまたは内部チャネルに接続されるチャネルに対応する DIFSEL ビットはリセット値（シングルエンド入力モード）に保持する必要があります。

ソフトウェアは､A/D コンバータが無効なときだけ（ADCAL = 0､JADSTART = 0､JADSTP = 0､ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、これらのビットに書き込むことができます。

21.7.22 A/D コンバータ較正係数（ADC_CALFACT）

アドレスオフセット：0xB4

リセット値：0x0000 0000

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **CALFACT_D[6:0]**：差動モードでの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

差動入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ A/D コンバータに格納されている現在の値と異なる場合は、新しい差動較正の起動時に、この新しい較正係数が適用されます。

注： ソフトウェアは、ADEN = 1、ADSTART = 0、および JADSTART = 0 のときのみ（A/D コンバータが有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **CALFACT_S[6:0]**：シングルエンドモードの較正係数

これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。

シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。

ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ A/D コンバータに格納されている現在の値と異なる場合は、新しいシングルエンド較正の起動時に、この新しい較正係数が適用されます。

注： ソフトウェアは、ADEN = 1、ADSTART = 0、および JADSTART = 0 のときのみ（A/D コンバータが有効であり、較正中でなく、変換中でない）、これらのビットに書き込むことができます。

21.7.23 A/D コンバータ･ゲイン補正レジスタ（ADC_GCOMP）

アドレスオフセット：0xC0

リセット値：0x0000 0000

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

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:0 **GCOMPCOEFF[13:0]**：ゲイン補正係数

これらのビットは、ゲイン補正係数をプログラムするために、ソフトウェアによってセット／クリアされます。

00 1000 0000 0000：ゲイン係数 0.5

.....

01 0000 0000 0000：ゲイン係数 1

10 0000 0000 0000：ゲイン係数 2

11 0000 0000 0000：ゲイン係数 3

.....

係数を 4096 で除算して、0 から 3.999756 までのゲイン係数を求めます。

注： このゲイン補正は、ADC_CFGR2 レジスタの GCOMP ビットが 1 にセットされているときのみ適用されます。

21.8 ADC 共通レジスタ

これらのレジスタは、マスタおよびスレーブ ADC に共通の制御およびステータスレジスタを定義します。

21.8.1 A/D コンバータ x 共通ステータス･レジスタ（ADCx_CSR）（x = 12 または 345）

アドレスオフセット：0x300

リセット値：0x0000 0000

このレジスタにより、別の ADC のステータスビットの状態を確認することができます。ただし、これは読出し専用で、別の ADC のステータスビットをクリアすることはできません。その代わり、対応する ADC_ISR レジスタに 0 を書き込むことによって各ステータスビットをクリアする必要があります。

1 つのインタフェースは ADC1 と ADC2 を制御し、もう 1 つのインタフェースは ADC3、ADC4、および ADC5 を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV
					r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
					r	r	r	r	r	r	r	r	r	r	r

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **JQOVF_SLV**：スレーブ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADC_ISR レジスタの JQOVF ビットのコピーです。

ビット 25 **AWD3_SLV**：スレーブ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADC_ISR レジスタの AWD3 ビットのコピーです。

ビット 24 **AWD2_SLV**：スレーブ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADC_ISR レジスタの AWD2 ビットのコピーです。

ビット 23 **AWD1_SLV**：スレーブ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADC_ISR レジスタの AWD1 ビットのコピーです。

ビット 22 **JEOS_SLV**：スレーブ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの JEOS ビットのコピーです。

ビット 21 **JEOC_SLV**：スレーブ ADC のインジェクト変換の終了フラグ
このビットは対応する ADC_ISR レジスタの JEOC ビットのコピーです。

ビット 20 **OVR_SLV**：スレーブ ADC のオーバーランフラグ
このビットは対応する ADC_ISR レジスタの OVR ビットのコピーです。

ビット 19 **EOS_SLV**:スレーブ ADC のレギュラシーケンスの終了フラグこのビットは対応する ADC_ISR レジスタの EOS ビットのコピーです。

ビット 18 **EOC_SLV**：スレーブ ADC のレギュラ変換の終了フラグ
このビットは対応する ADC_ISR レジスタの EOC ビットのコピーです。

ビット 17 **EOSMP_SLV**：スレーブ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADC_ISR レジスタの EOSMP2 ビットのコピーです。

ビット 16 **ADRDY_SLV**：スレーブ ADC レディ
このビットは対応する ADC_ISR レジスタの ADRDY ビットのコピーです。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **JQOVF_MST**：マスタ ADC のインジェクトコンテキストキューのオーバーフローフラグ
このビットは対応する ADC_ISR レジスタの JQOVF ビットのコピーです。

ビット 9 **AWD3_MST**：マスタ ADC のアナログウォッチドッグ 3 フラグ
このビットは対応する ADC_ISR レジスタの AWD3 ビットのコピーです。

ビット 8 **AWD2_MST**：マスタ ADC のアナログウォッチドッグ 2 フラグ
このビットは対応する ADC_ISR レジスタの AWD2 ビットのコピーです。

ビット 7 **AWD1_MST**：マスタ ADC のアナログウォッチドッグ 1 フラグ
このビットは対応する ADC_ISR レジスタの AWD1 ビットのコピーです。

ビット 6 **JEOS_MST**：マスタ ADC のインジェクトシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの JEOS ビットのコピーです。

ビット 5 **JEOC_MST**：マスタ ADC のインジェクト変換の終了フラグ
このビットは対応する ADC_ISR レジスタの JEOC ビットのコピーです。

ビット 4 **OVR_MST**：マスタ ADC のオーバーランフラグ
このビットは対応する ADC_ISR レジスタの OVR ビットのコピーです。

ビット 3 **EOS_MST**：マスタ ADC のレギュラシーケンスの終了フラグ
このビットは対応する ADC_ISR レジスタの EOS ビットのコピーです。

ビット 2 **EOC_MST**：マスタ ADC のレギュラ変換の終了フラグ
このビットは対応する ADC_ISR レジスタの EOC ビットのコピーです。

ビット 1 **EOSMP_MST**：マスタ ADC のサンプリングフェーズの終了フラグ
このビットは対応する ADC_ISR レジスタの EOSMP ビットのコピーです。

ビット 0 **ADRDY_MST**：マスタ ADC レディ
このビットは対応する ADC_ISR レジスタの ADRDY ビットのコピーです。

21.8.2 A/D コンバータ x 共通制御レジスタ（ADCx_CCR）（x = 12 または 345）

アドレスオフセット：0x308

リセット値：0x0000 0000

1 つのインタフェースは ADC1 と ADC2 を制御し、もう 1 つのインタフェースは ADC3、ADC4、および ADC5 を制御します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATSEL	VSENSESEL	VREFEN	PRESC[3:0]				CKMODE[1:0]	
							rw	rw	rw	rw	rw	rw	rw	rw	rw

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

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **VBATSEL**：VBAT 選択

このビットは、VBAT を制御するために、ソフトウェアによってセットまたはクリアされます。

0：V_{BAT} チャネルは無効です。

1：V_{BAT} チャネルは有効です。

ビット 23 **VSENSESEL**：V_{TS} 選択

このビットは、V_{TS} を制御するために、ソフトウェアによってセットまたはクリアされます。

0：温度センサ･チャネルは無効です。

1：温度センサチャネルは有効です。

ビット 22 **VREFEN**：V_{REFINT} 有効化

このビットは、V_{REFINT} チャネルを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：V_{REFINT} チャネルは無効です。

1：V_{REFINT} チャネルは有効です。

ビット 21:18 **PRESC[3:0]**：ADC プリスケーラ

これらのビットは、ADC へのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべての ADC に共通です。

0000：ADC 入力クロックは分周されません。
0001：ADC 入力クロックが 2 分周されます。
0010：ADC 入力クロックが 4 分周されます。
0011：ADC 入力クロックが 6 分周されます。
0100：ADC 入力クロックが 8 分周されます。
0101：ADC 入力クロックが 10 分周されます。
0110：ADC 入力クロックが 12 分周されます。
0111：ADC 入力クロックが 16 分周されます。
1000：ADC 入力クロックが 32 分周されます。
1001：ADC 入力クロックが 64 分周されます。
1010：ADC 入力クロックが 128 分周されます。
1011：ADC 入力クロックが 256 分周されます。
その他：予約済み

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、これらのビットに書き込むことができます。CKMODE[1:0] = 00 である場合のみ、A/D コンバータのプリスケーラ値が適用されます。

ビット 17:16 **CKMODE[1:0]**：ADC クロックモード

これらのビットは、ADC クロックスキーム（マスタとスレーブの両方の ADC に共通）を定義するために、ソフトウェアによってセット／クリアされます。

00：adc_ker_ck（x = 123）（非同期クロック･モード）。製品レベルで生成されます（セクション 6：リセットおよびクロック制御（RCC）を参照）。

01：adc_hclk/1（同期クロックモード）。この設定は、AHB クロック･プリスケーラが設定され（RCC_CFGR レジスタの HPRE[3:0] = 0xxx）、システム･クロックのデューティ･サイクルが 50% の場合にのみ有効にする必要があります。

10：adc_hclk/2（同期クロック･モード）。

11：adc_hclk/4（同期クロックモード）。

すべての同期クロックモードにおいて、タイマトリガから変換開始までの遅延にジッタはありません。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、これらのビットに書き込むことができます。

ビット 15:14 **MDMA[1:0]**：デュアル A/D コンバータ･モードのためのダイレクト･メモリ･アクセス･モード

このビットフィールドは、ソフトウェアによってセット／クリアされます。詳細については、DMA コントローラのセクションを参照してください。

00：MDMA モードを無効化します。

01：予約済み

10：MDMA モードは、12 ビットおよび 10 ビットの分解能で有効です。

11：MDMA モードは、8 ビットおよび 6 ビットの分解能で有効です。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 13 **DMACFG**：DMA 設定（デュアル ADC モード）

このビットは、2 つの DMA 動作モードのいずれかを選択するためにソフトウェアによってセットまたはクリアされ、DMAEN = 1 のときのみ有効です。

0：DMA ワンショット･モードが選択されます。

1：DMA サーキュラ･モードが選択されます。

詳細については、セクション ：DMA を使用した変換の管理を参照してください。

注： ソフトウェアは、ADSTART = 0 である（レギュラ変換が実行中でない）ときのみ、これらのビットに書き込むことができます。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **DELAY**：2 つのサンプリング･フェーズ間の遅延

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットはデュアルインタリーブモードで使用します。ADC 分解能の値対 DELAY ビット値については、表 176 を参照してください。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、これらのビットに書き込むことができます。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DUAL[4:0]**：デュアル ADC モード選択

これらのビットは、動作モードを選択するために、ソフトウェアによって書き込まれます。

すべての ADC が単独で動作します。

00000：独立モード

00001 から 01001：デュアルモード、マスタおよびスレーブ ADCは協調して動作

00001：レギュラ同時 + インジェクト同時モードの組み合わせ

00010：レギュラ同時 + オルタネートトリガモードの組み合わせ

00011：インタリーブモード + インジェクト同時モードの組み合わせ

00100：予約済みです。

00101：インジェクト同時モードのみ

00110：レギュラ同時モードのみ

00111：インタリーブモードのみ

01001：オルタネートトリガモードのみ

他のすべての組み合わせは予約済みであり、プログラムすることはできません。

注： ソフトウェアは、A/D コンバータが無効なときだけ（ADCAL = 0、JADSTART = 0、ADSTART = 0、ADSTP = 0、ADDIS = 0、および ADEN = 0）、これらのビットに書き込むことができます。

表 176. DELAY ビット対 A/D コンバータ分解能

DELAY ビット	12 ビット分解能	10 ビット分解能	8 ビット分解能	6 ビット分解能
0000	1 * $T_{adc_ker_ck}$	1 * $T_{adc_ker_ck}$	1 * $T_{adc_ker_ck}$	1 * $T_{adc_ker_ck}$
0001	2 * $T_{adc_ker_ck}$	2 * $T_{adc_ker_ck}$	2 * $T_{adc_ker_ck}$	2 * $T_{adc_ker_ck}$
0010	3 * $T_{adc_ker_ck}$	3 * $T_{adc_ker_ck}$	3 * $T_{adc_ker_ck}$	3 * $T_{adc_ker_ck}$
0011	4 * $T_{adc_ker_ck}$	4 * $T_{adc_ker_ck}$	4 * $T_{adc_ker_ck}$	4 * $T_{adc_ker_ck}$
0100	5 * $T_{adc_ker_ck}$	5 * $T_{adc_ker_ck}$	5 * $T_{adc_ker_ck}$	5 * $T_{adc_ker_ck}$
0101	6 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
0110	7 * $T_{adc_ker_ck}$	7 * $T_{adc_ker_ck}$	7 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
0111	8 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
1000	9 * $T_{adc_ker_ck}$	9 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
1001	10 * $T_{adc_ker_ck}$	10 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
1010	11 * $T_{adc_ker_ck}$	10 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
1011	12 * $T_{adc_ker_ck}$	10 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$
その他	12 * $T_{adc_ker_ck}$	10 * $T_{adc_ker_ck}$	8 * $T_{adc_ker_ck}$	6 * $T_{adc_ker_ck}$

21.8.3 A/D コンバータ x デュアル･モード用共通レギュラ･データ･レジスタ（ADCx_CDR）（x = 12 または 345）

アドレスオフセット：0x30C

リセット値：0x0000 0000

1 つのインタフェースは ADC1 と ADC2 を制御し、もう 1 つのインタフェースは ADC3、ADC4、および ADC5 を制御します。

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

ビット 31:16 **RDATA_SLV[15:0]**：スレーブ ADC のレギュラデータ

デュアルモードでは、これらのビットにはスレーブ ADC のレギュラデータが格納されます。セクション 21.4.30：デュアル ADC モード を参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）に示すように適用されています。

ビット 15:0 **RDATA_MST[15:0]**：マスタ ADC のレギュラデータ

デュアルモードでは、これらのビットにはマスタ ADC のレギュラデータが格納されます。セクション 21.4.30：デュアル ADC モード を参照してください。

データの配置は、セクション ：データレジスタ、データの配置およびデータオフセット（ADC_DR、OFFSETy、OFFSETy_CH、ALIGN）に示すように適用されています。

MDMA = 11 モードでは、ビット 15:8 には SLV_ADC_DR[7:0] が、ビット 7:0 には MST_ADC_DR[7:0] が含まれています。

21.9 A/D コンバータ･レジスタ･マップ

次の表に A/D コンバータのレジスタの一覧を示します。

表 177. A/D コンバータ･グローバル･レジスタ･マップ

オフセット	レジスタ
0x000 - 0x0FC	マスタ ADC1/ADC3
0x100 - 0x1FC	スレーブ ADC2/ADC4
0x200 - 0x2FC	予約済み/シングル ADC5
0x300 - 0x30C	マスタおよびスレーブ ADC に共通のレジスタ

表 178. A/D コンバータ･レジスタ･マップとリセット値（各 ADC、オフセット = 0x000（マスタ A/D コンバータ）、0x100（スレーブ A/D コンバータ））

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADC_ISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVF	AWD3	AWD2	AWD1	JEOS	JEOC	OVR	EOS	EOC	EOSMP	ADRDY
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x04	**ADC_IER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JQOVFIE	AWD3IE	AWD2IE	AWD1IE	JEOSIE	JEOCIE	OVRIE	EOSIE	EOCIE	EOSMPIE	ADRDYIE
	リセット値																						0	0	0	0	0	0	0	0	0	0	0
0x08	**ADC_CR**	ADCAL	ADCALDIF	DEEPPWD	ADVREGEN	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JADSTP	ADSTP	JADSTART	ADSTART	ADDIS	ADEN
	リセット値	0	0	1	0																							0	0	0	0	0	0
0x0C	**ADC_CFGR**	JQDIS	AWD1CH[4:0]					JAUTO	JAWD1EN	AWD1EN	AWD1SGL	JQM	JDISCEN	DISCNUM [2:0]			DISCEN	ALIGN	AUTDLY	CONT	OVRMOD	EXTEN[1:0]		EXTSEL4	EXTSEL3	EXTSEL2	EXTSEL1	EXTSEL0	RES [1:0]		Res.	DMACFG	DMAEN
	リセット値	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0
0x10	**ADC_CFGR2**	Res.	Res.	Res.	Res.	SMPTRIG	BULB	SWTRIG	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GCOMP	Res.	Res.	Res.	Res.	Res.	ROVSM	TROVS	OVSS[3:0]					OVSR [2:0]			JOVSE	ROVSE
	リセット値					0	0	0									0						0	0	0	0	0	0	0	0	0	0	0
0x14	**ADC_SMPR1**	SMPPLUS	Res.	SMP9 [2:0]			SMP8 [2:0]			SMP7 [2:0]			SMP6 [2:0]			SMP5 [2:0]			SMP4 [2:0]			SMP3 [2:0]			SMP2 [2:0]			SMP1 [2:0]			SMP0 [2:0]		
	リセット値	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**ADC_SMPR2**	Res.	Res.	Res.	Res.	Res.	SMP18 [2:0]			SMP17 [2:0]			SMP16 [2:0]			SMP15 [2:0]			SMP14 [2:0]			SMP13 [2:0]			SMP12 [2:0]			SMP11 [2:0]			SMP10 [2:0]		
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	予約済み	Res.																															
0x20	**ADC_TR1**	Res.	Res.	Res.	Res.	HT1 [11:0]												Res.	AWDFILT [2:0]			LT1[11:0]												
	リセット値					1	1	1	1	1	1	1	1	1	1	1	1					0	0	0	0	0	0	0	0	0	0	0	0
0x24	**ADC_TR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT2[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT2 [7:0]						
	リセット値									1	1	1	1	1	1	1	1									0	0	0	0	0	0	0	0
0x28	**ADC_TR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HT3[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LT3 [7:0]						
	リセット値									1	1	1	1	1	1	1	1									0	0	0	0	0	0	0	0
0x2C	予約済み	Res.																															
0x30	**ADC_SQR1**	Res.	Res.	Res.	SQ4 [4:0]					Res.	SQ3 [4:0]					Res.	SQ2 [4:0]					Res.	SQ1 [4:0]					Res.	Res.	L[3:0]			
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0			0	0	0	0

表 178. A/D コンバータ･レジスタ･マップとリセット値（各 ADC、オフセット = 0x000（マスタ A/D コンバータ）、0x100（スレーブ A/D コンバータ））（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x34	**ADC_SQR2**	Res.	Res.	Res.	SQ9 [4:0]					Res.	SQ8 [4:0]					Res.	SQ7 [4:0]					Res.	SQ6 [4:0]					Res.	SQ5 [4:0]				
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x38	**ADC_SQR3**	Res.	Res.	Res.	SQ14 [4:0]					Res.	SQ13 [4:0]					Res.	SQ12 [4:0]					Res.	SQ11 [4:0]					Res.	SQ10 [4:0]				
	リセット値				0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0
0x3C	**ADC_SQR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SQ16 [4:0]					Res.	SQ15 [4:0]				
	リセット値																						0	0	0	0	0		0	0	0	0	0
0x40	**ADC_DR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	レギュラ RDATA[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44-0x48	予約済み	Res.																															
0x4C	**ADC_JSQR**	JSQ4[4:0]					Res.	JSQ3 [4:0]					Res.	JSQ2 [4:0]			Res.	JSQ1 [4:0]				JEXTEN[1:0]	JEXTSEL [4:0]					JL[1:0]				
	リセット値	0	0	0	0	0		0	0	0	0	0		0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50-0x5C	予約済み	Res.																															
0x60	**ADC_OFR1**	OFFSET1_EN	OFFSET1_CH[4:0]					SATEN	OFFSETPOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET1 [11:0]											
	リセット値	0	0	0	0	0	0	0	0													0	0	0	0	0	0	0	0	0	0	0	0
0x64	**ADC_OFR2**	OFFSET2_EN	OFFSET2_CH[4:0]					SATEN	OFFSETPOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET2 [11:0]											
	リセット値	0	0	0	0	0	0	0	0													0	0	0	0	0	0	0	0	0	0	0	0
0x68	**ADC_OFR3**	OFFSET3_EN	OFFSET3_CH[4:0]					SATEN	OFFSETPOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET3 [11:0]											
	リセット値	0	0	0	0	0	0	0	0													0	0	0	0	0	0	0	0	0	0	0	0
0x6C	**ADC_OFR4**	OFFSET4_EN	OFFSET4_CH[4:0]					SATEN	OFFSETPOS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OFFSET4 [11:0]											
	リセット値	0	0	0	0	0	0	0	0													0	0	0	0	0	0	0	0	0	0	0	0
0x70-0x7C	予約済み	Res.																															
0x80	**ADC_JDR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA1 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x84	**ADC_JDR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA2 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x88	**ADC_JDR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA3 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C	**ADC_JDR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	JDATA4 [15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8C-0x9C	予約済み	Res.																															
0xA0	**ADC_AWD2CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD2CH [18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xA4	**ADC_AWD3CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AWD3CH [18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 178. A/D コンバータ･レジスタ･マップとリセット値（各 ADC、オフセット = 0x000（マスタ A/D コンバータ）、0x100（スレーブ A/D コンバータ））（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xA8-0xAC	予約済み	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0xB0	**ADC_DIFSEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIFSEL[18:0]																		
	リセット値														0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0xB4	**ADC_CALFACT**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT_D[6:0]							Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALFACT_S[6:0]						
	リセット値										0	0	0	0	0	0	0										0	0	0	0	0	0	0
0xC0	**ADC_GCOMP**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GCOMP[13:0]													
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 179. ADC レジスタマップとリセット値（マスタおよびスレーブ ADC 共通レジスタ）オフセット =0x300

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**ADCx_CSR**	Res.	Res.	Res.	Res.	Res.	JQOVF_SLV	AWD3_SLV	AWD2_SLV	AWD1_SLV	JEOS_SLV	JEOC_SLV	OVR_SLV	EOS_SLV	EOC_SLV	EOSMP_SLV	ADRDY_SLV	Res.	Res.	Res.	Res.	Res.	JQOVF_MST	AWD3_MST	AWD2_MST	AWD1_MST	JEOS_MST	JEOC_MST	OVR_MST	EOS_MST	EOC_MST	EOSMP_MST	ADRDY_MST
							スレーブ ADC2																マスタ ADC1										
	リセット値						0	0	0	0	0	0	0	0	0	0	0						0	0	0	0	0	0	0	0	0	0	0
0x04	予約済みです。	Res.																															
0x08	**ADCx_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VBATSEL	VSENSESEL	VREFEN	PRESC[3:0]				CKMODE[1:0]		MDMA[1:0]		DMACFG	Res.	DELAY[3:0]				Res.	Res.	Res.	DUAL[4:0]				
	リセット値								0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0				0	0	0	0	0
0x0C	**ADCx_CDR**	RDATA_SLV[15:0]																RDATA_MST[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

22 D/A コンバータ（DAC）

22.1 概要

DAC モジュールは、12 ビットの電圧出力デジタルアナログコンバータです。DAC は、8 または 12 ビットモードで設定でき、DMA コントローラと組み合わせて使用することもできます。12 ビットモードでは、データを左詰め右詰めのどちらにも配置できます。DAC には最大 2 つの出力チャネルがあり、それぞれがコンバータを搭載しています。デュアル DAC チャネルモードでは、変換は独立して行うか、両方のチャネルが同期更新操作のためにグループ化されているときには同時に行うことができます。精度を高めるために、入力基準ピン V_{REF+}（その他のアナログペリフェラルと共用）を使用することができます。内部基準も、同じ入力でセットできます。電圧基準バッファ（VREFBUF）セクションを参照してください。

D/A コンバータ出力が出力パッドから切断され、オンチップ･ペリフェラルに接続されている場合、DACx_OUTy ピンは汎用入出力（GPIO）として使用できます。駆動出力電流を多くするために、必要に応じて D/A コンバータの出力バッファを有効化できます。個別の較正を各 D/A コンバータ出力チャネルに適用可能です。D/A コンバータ出力チャネルは低電力モード（サンプルおよびホールドモード）をサポートしています。

22.2 DAC の主な機能

DAC の主な機能は以下のとおりです（図 156 : デュアル･チャネル D/A コンバータのブロック図を参照）。

- 最大 4 つの DAC インタフェースで、それぞれ最大 2 つの出力チャネル
- 12 ビットモードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波および三角波生成
- のこぎり波生成
- デュアル DAC チャネルの独立または同時変換
- DMA アンダーランエラー検出を含む各チャネルの DMA 機能
- バス活動を軽減するダブルデータ DMA 機能
- 変換外部トリガ
- DAC 出力チャネルバッファ／バッファ無モード
- バッファオフセット較正
- 各 D/A コンバータ出力を DACx_OUTy 出力ピンから切り離し可能
- オンチップペリフェラルへの D/A コンバータ出力接続
- STOP モードの低電力動作でのサンプル･アンド･ホールド･モード
- V_{REF+}ピン または内部 VREFBUF 基準からの入力電圧基準

図 156 に D/A コンバータ･チャネルのブロック図、表 181 にピンの概要を示します。

22.3 D/A コンバータの実装

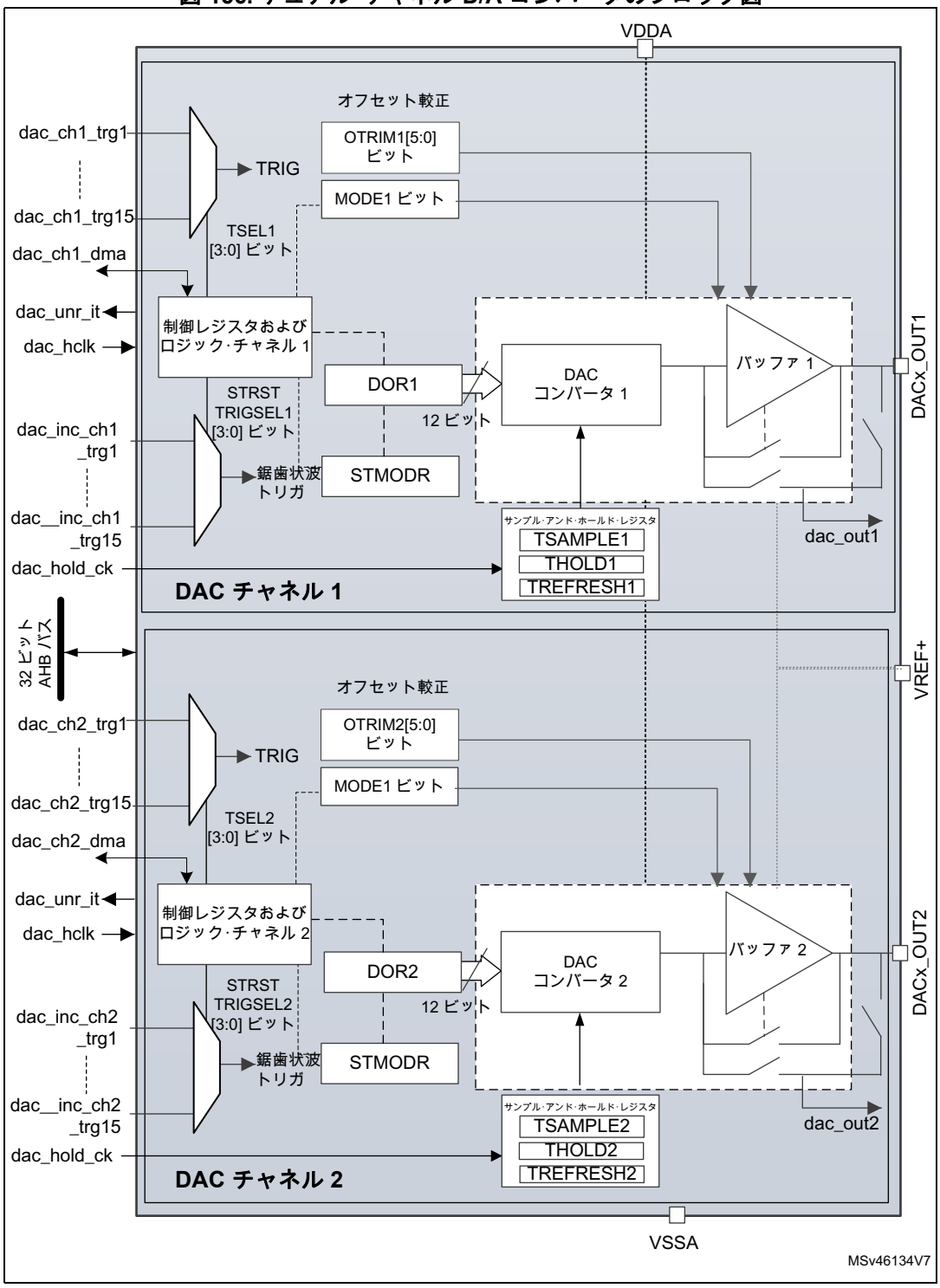
表 180. D/A コンバータの機能

D/A コンバータの機能	DAC1	DAC2	DAC3	DAC4
デュアル･チャネル	X	-	X	X
出力バッファ	X	X	-	-
I/O 接続	PA4 での DAC1_OUT1 PA5 での DAC1_OUT2	PA6 での DAC2_OUT1	GPIO への接続なし	
最大サンプリング時間	1MSPS		15MSPS	
自律モード	-			
VREF+ ピン	X			

22.4 D/A コンバータの機能詳細

22.4.1 D/A コンバータのブロック図

図 156. デュアル･チャネル D/A コンバータのブロック図



1. DAC_MCR の MODEx ビットによって、出力モードが制御され、バッファの有無設定の通常モードとサンプル･アンド･ホールド･モードを切り替えることができます。
2. チャネル2 の有無については、セクション 22.3：D/A コンバータの実装を参照してください。
3. DAC チャネル2 は、DAC1、DAC3、および DAC4 でのみ使用できます。

22.4.2 DAC ピンおよび内部信号

DAC には次の要素が含まれます。

- 最大 2 つの出力チャネル
- DACx_OUTy を出力ピンから切り離して標準の GPIO として使用可能
- dac_outx でコンパレータ、オペアンプ、および A/D コンバータ（使用できる場合）などのオンチップ･ペリフェラルへの内部ピン接続が使用可能
- D/A コンバータ出力チャネルのバッファ有りまたはバッファ無し
- スタティック変換のために LSI/LSE クロック･ソース（dac_hold_ck）を使用し、STOP モードで動作可能なサンプル･アンド･ホールド･ブロックとレジスタ。

D/A コンバータは最大で 2 つの個別出力チャネルを備えています。各出力チャネルは、コンパレータ、オペアンプ、および ADC（使用できる場合）などのオンチップペリフェラルに接続できます。この場合、D/A コンバータ出力チャネルを DACx_OUTy 出力ピンから切断でき、対応する GPIO を別の目的に使用できます。

D/A コンバータ出力にはバッファの有無を設定できます。サンプル･アンド･ホールド･ブロックと、それに関連するレジスタは、LSIまたは LSE クロック･ソース（dac_hold_ck）を使用して STOP モードで動作可能です。

表 181. D/A コンバータの入出力ピン

ピン名	信号タイプ	説明
VREF+	入力、アナログ基準電圧正	DAC のハイ レベル／正基準電圧 $V_{REF+} \leq V_{DDAmax}$ （データシートを参照）
VDDA	入力、アナログ電源	アナログ電源供給
VSSA	入力、アナログ電源グラウンド	アナログ電源供給のグラウンド
DACx_OUTy	アナログ出力信号	DACx チャネル y アナログ出力

表 182. D/A コンバータ入出力信号

内部信号名	信号タイプ	説明
dac_ch1_dma	双方向	D/A コンバータチャネル 1 DMA リクエスト／確認応答
dac_ch2_dma	双方向	D/A コンバータチャネル 2 DMA リクエスト／確認応答
dac_ch1_trgx（x = 1 ~ 15）	入力	DAC channel1 トリガ入力
dac_ch2_trgx（x = 1 ~ 15）	入力	DAC channel2 トリガ入力
dac_ch1_inc_trgx（x = 1 ~ 15）	入力	D/A コンバータ･チャネル 1、のこぎり波インクリメント･トリガ入力
dac_chn2_inc_trgx（x = 1 ~ 15）	入力	D/A コンバータ･チャネル 1、のこぎり波インクリメント･トリガ入力
dac_unr_it	出力	DAC アンダーラン割込み
dac_hclk	入力	DAC ペリフェラルクロック

表 182. D/A コンバータ入出力信号 （続き）

内部信号名	信号タイプ	説明
dac_hold_ck	入力	サンプルおよびホールドモードで使用される D/A コンバータ低電力クロック
dac_out1	アナログ出力	オンチップペリフェラル用 DAC channel1 出力
dac_out2	アナログ出力	オンチップペリフェラル用 DAC channel2 出力

表 183. DAC1 相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	ck_lsi or ck_lse（RCC で選択）	RCC で選択された LSI または LSE クロック
dac_chx_trg1（x = 1、2）	TIM8_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg6（x = 1、2）	EXTI9	外部ピン
dac_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ·タイマからの内部信号
dac_chx_trg9（x = 1、2）	hrtim_dac_reset_trg1	オンチップ·タイマからの内部信号
dac_chx_trg10（x = 1、2）	hrtim_dac_reset_trg2	オンチップ·タイマからの内部信号
dac_chx_trg11（x = 1、2）	hrtim_dac_reset_trg3	オンチップ·タイマからの内部信号
dac_chx_trg12（x = 1、2）	hrtim_dac_reset_trg4	オンチップ·タイマからの内部信号
dac_chx_trg13（x = 1、2）	hrtim_dac_reset_trg5	オンチップ·タイマからの内部信号
dac_chx_trg14（x = 1、2）	hrtim_dac_reset_trg6	オンチップ·タイマからの内部信号
dac_chx_trg15（x = 1、2）	hrtim_dac_trg1	オンチップ·タイマからの内部信号
dac_inc_chx_trg1（x = 1、2）	TIM8_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg6（x = 1、2）	EXTI10	外部ピン
dac_inc_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ·タイマからの内部信号
dac_inc_chx_trg9（x = 1、2）	hrtim_dac_step_trg1	オンチップ·タイマからの内部信号
dac_inc_chx_trg10（x = 1、2）	hrtim_dac_step_trg2	オンチップ·タイマからの内部信号
dac_inc_chx_trg11（x = 1、2）	hrtim_dac_step_trg3	オンチップ·タイマからの内部信号
dac_inc_chx_trg12（x = 1、2）	hrtim_dac_step_trg4	オンチップ·タイマからの内部信号
dac_inc_chx_trg13（x = 1、2）	hrtim_dac_step_trg5	オンチップ·タイマからの内部信号
dac_inc_chx_trg14（x = 1、2）	hrtim_dac_step_trg6	オンチップ·タイマからの内部信号

表 184. DAC2 相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	ck_lsi or ck_lse（RCC で選択）	RCC で選択された LSI または LSE クロック
dac_ch1_trg1	TIM8_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg2	TIM7_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg3	TIM15_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg4	TIM2_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg5	TIM4_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg6	EXTI9	外部ピン
dac_ch1_trg7	TIM6_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg8	TIM3_TRGO	オンチップ･タイマからの内部信号
dac_ch1_trg9	hrtim_dac_reset_trg1	オンチップ･タイマからの内部信号
dac_ch1_trg10	hrtim_dac_reset_trg2	オンチップ･タイマからの内部信号
dac_ch1_trg11	hrtim_dac_reset_trg3	オンチップ･タイマからの内部信号
dac_ch1_trg12	hrtim_dac_reset_trg4	オンチップ･タイマからの内部信号
dac_ch1_trg13	hrtim_dac_reset_trg5	オンチップ･タイマからの内部信号
dac_ch1_trg14	hrtim_dac_reset_trg6	オンチップ･タイマからの内部信号
dac_ch1_trg15	hrtim_dac_trg2	オンチップ･タイマからの内部信号
dac_inc_ch1_trg1	TIM8_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg2	TIM7_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg3	TIM15_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg4	TIM2_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg5	TIM4_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg6	EXTI10	外部ピン
dac_inc_ch1_trg7	TIM6_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg8	TIM3_TRGO	オンチップ･タイマからの内部信号
dac_inc_ch1_trg9	hrtim_dac_step_trg1	オンチップ･タイマからの内部信号
dac_inc_ch1_trg10	hrtim_dac_step_trg2	オンチップ･タイマからの内部信号
dac_inc_ch1_trg11	hrtim_dac_step_trg3	オンチップ･タイマからの内部信号
dac_inc_ch1_trg12	hrtim_dac_step_trg4	オンチップ･タイマからの内部信号
dac_inc_ch1_trg13	hrtim_dac_step_trg5	オンチップ･タイマからの内部信号
dac_inc_ch1_trg14	hrtim_dac_step_trg6	オンチップ･タイマからの内部信号

表 185. DAC3 相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	ck_lsi or ck_lse（RCC で選択）	RCC で選択された LSI または LSE クロック
dac_chx_trg1（x = 1、2）	TIM1_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg6（x = 1、2）	EXTI9	外部ピン
dac_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ・タイマからの内部信号
dac_chx_trg9（x = 1、2）	hrtim_dac_reset_trg1	オンチップ・タイマからの内部信号
dac_chx_trg10（x = 1、2）	hrtim_dac_reset_trg2	オンチップ・タイマからの内部信号
dac_chx_trg11（x = 1、2）	hrtim_dac_reset_trg3	オンチップ・タイマからの内部信号
dac_chx_trg12（x = 1、2）	hrtim_dac_reset_trg4	オンチップ・タイマからの内部信号
dac_chx_trg13（x = 1、2）	hrtim_dac_reset_trg5	オンチップ・タイマからの内部信号
dac_chx_trg14（x = 1、2）	hrtim_dac_reset_trg6	オンチップ・タイマからの内部信号
dac_chx_trg15（x = 1、2）	hrtim_dac_trg3	オンチップ・タイマからの内部信号
dac_inc_chx_trg1（x = 1、2）	TIM1_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg6（x = 1、2）	EXTI10	外部ピン
dac_inc_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ・タイマからの内部信号
dac_inc_chx_trg9（x = 1、2）	hrtim_dac_step_trg1	オンチップ・タイマからの内部信号
dac_inc_chx_trg10（x = 1、2）	hrtim_dac_step_trg2	オンチップ・タイマからの内部信号
dac_inc_chx_trg11（x = 1、2）	hrtim_dac_step_trg3	オンチップ・タイマからの内部信号
dac_inc_chx_trg12（x = 1、2）	hrtim_dac_step_trg4	オンチップ・タイマからの内部信号
dac_inc_chx_trg13（x = 1、2）	hrtim_dac_step_trg5	オンチップ・タイマからの内部信号
dac_inc_chx_trg14（x = 1、2）	hrtim_dac_step_trg6	オンチップ・タイマからの内部信号

表 186. DAC4 相互接続

信号名	転送元	ソースタイプ
dac_hold_ck	ck_lsi または ck_lse（RCC で選択）	RCC で選択された LSI または LSE クロック
dac_chx_trg1（x = 1、2）	TIM8_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg6（x = 1、2）	EXTI9	外部ピン
dac_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ･タイマからの内部信号
dac_chx_trg9（x = 1、2）	hrtim_dac_reset_trg1	オンチップ･タイマからの内部信号
dac_chx_trg10（x = 1、2）	hrtim_dac_reset_trg2	オンチップ･タイマからの内部信号
dac_chx_trg11（x = 1、2）	hrtim_dac_reset_trg3	オンチップ･タイマからの内部信号
dac_chx_trg12（x = 1、2）	hrtim_dac_reset_trg4	オンチップ･タイマからの内部信号
dac_chx_trg13（x = 1、2）	hrtim_dac_reset_trg5	オンチップ･タイマからの内部信号
dac_chx_trg14（x = 1、2）	hrtim_dac_reset_trg6	オンチップ･タイマからの内部信号
dac_chx_trg15（x = 1、2）	hrtim_dac_trg1	オンチップ･タイマからの内部信号
dac_inc_chx_trg1（x = 1、2）	TIM8_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg2（x = 1、2）	TIM7_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg3（x = 1、2）	TIM15_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg4（x = 1、2）	TIM2_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg5（x = 1、2）	TIM4_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg6（x = 1、2）	EXTI10	外部ピン
dac_inc_chx_trg7（x = 1、2）	TIM6_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg8（x = 1、2）	TIM3_TRGO	オンチップ･タイマからの内部信号
dac_inc_chx_trg9（x = 1、2）	hrtim_dac_step_trg1	オンチップ･タイマからの内部信号
dac_inc_chx_trg10（x = 1、2）	hrtim_dac_step_trg2	オンチップ･タイマからの内部信号
dac_inc_chx_trg11（x = 1、2）	hrtim_dac_step_trg3	オンチップ･タイマからの内部信号
dac_inc_chx_trg12（x = 1、2）	hrtim_dac_step_trg4	オンチップ･タイマからの内部信号
dac_inc_chx_trg13（x = 1、2）	hrtim_dac_step_trg5	オンチップ･タイマからの内部信号
dac_inc_chx_trg14（x = 1、2）	hrtim_dac_step_trg6	オンチップ･タイマからの内部信号

22.4.3 D/A コンバータチャネルイネーブル

各 DAC チャネルは、DAC_CR レジスタの対応する ENx ビットをセットすることによって起動できます。DAC チャネルは、t_{WAKEUP} のスタートアップ時間後に有効になります。

DAC インタフェースがデータを受信する準備ができているとき、DAC_SR レジスタの DAC x RDY ビットがセットされます。DACxRDY 信号がリセットされているときに ENx ビットがセットされている場合、新しいデータの書込みまたはトリガのアサートはできません。

注： **ENx ビットは、アナログ DAC チャネル x のみを有効にします。DAC チャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。**

22.4.4 DAC データフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- シングル DAC チャネル

  この場合、次の 3 つの設定が可能です。

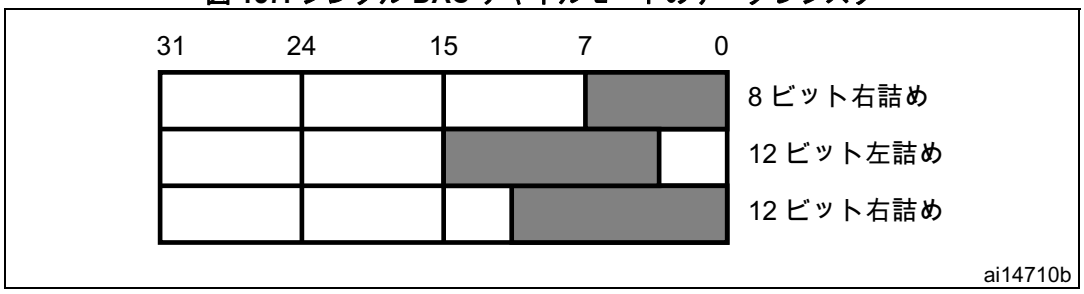
  – 8 ビット右詰め：ソフトウェアはデータを DAC_DHR8Rx[7:0] ビットにロードする必要があります（DHRx[11:4] ビットに格納）。

  – 12 ビット左詰め：ソフトウェアはデータを DAC_DHR12Lx [15:4] ビットにロードする必要があります（DHRx[11:0] ビットに格納）。

  – 12 ビット右詰め：ソフトウェアはデータを DAC_DHR12Rx [11:0] ビットにロードする必要があります（DHRx [11:0] ビットに格納）。

ユーザによって書き込まれたデータは、ロードされた DAC_DHRyyyx レジスタに応じて、シフトされてから、対応する DHRx（メモリマップされない内部レジスタであるデータ保持レジスタ x）に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガまたは外部イベントトリガによって、DORx レジスタにロードされます。

図 157. シングル DAC チャネルモードのデータレジスタ



- デュアル DAC チャネル（使用できる場合）

  この場合、次の 3 つの設定が可能です。

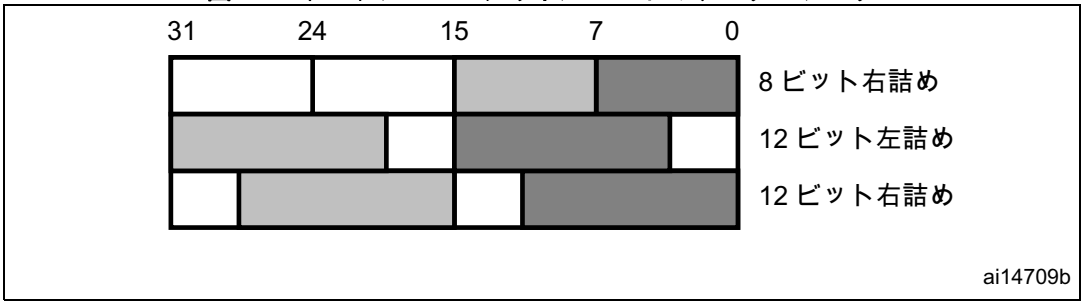
  – 8 ビット右詰め：DAC チャネル 1 のデータは、DAC_DHR8RD [7:0] ビット（DHR1[11:4] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR8RD [15:8] ビット（DHR2[11:4] ビットに格納）にロードされます。

  – 12 ビット左詰め：DAC チャネル 1 のデータは、DAC_DHR12LD [15:4] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12LD [31:20] ビット（DHR2[11:0]ビットに格納）にロードされます。

  – 12 ビット右詰め：DAC チャネル 1 のデータは、DAC_DHR12RD [11:0] ビット（DHR1[11:0] ビットに格納）に、DAC チャネル 2 のデータは、DAC_DHR12RD [27:16] ビット（DHR2[11:0] ビットに格納）にロードされます。

ロードされた DAC_DHRyyyD レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1 および DHR2（Data Holding Register、メモリマップされない内部レジスタ）に格納されます。その後、DHR1 および DHR2 レジスタは、自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、それぞれ DAC_DOR1 および DOR2 レジスタにロードされます。

図 158. デュアル DAC チャネルモードのデータレジスタ



符号付き/符号なしデータ

DAC 入力データは符号なし：12 ビットモードでは、0x000 は最小値に対応し、0xFFF は最大値に対応します。

DAC は 2 の補数形式の符号付き入力データも扱うことができます。このためには、DAC_MCR レジスタの SINFORMATx ビットをセットします。

SINFORMATx ビットがセットされると、DHRx レジスタに書き込まれたデータの MSB ビットは、DAC_DORx レジスタにコピーされるときに反転され、DAC インタフェースは符号付データを受け入れることができます（Q1.15、Q1.11、または Q1.7 形式）。DAC_DHR12Lx レジスタを使用して、16 ビット符号付きデータをデータ保持レジスタに格納することができます。16 ビットデータの 12 MSB は、DAC 出力データに使用され、MSB ビットは反転されます。4 つの LSB は単純に無視されます。

表 187. データフォーマット（12 ビットデータの場合）

SINFORMATx ビット	DHRx レジスタに書き込まれるデータ	DORx レジスタに転送されるデータ
0	0x000	0x000
0	0xFFF	0xFFF
1	0x7FF	0xFFF
1	0x000	0x800
1	0xFFF	0x7FF
1	0x800	0x000

22.4.5 D/A コンバータによる変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって（DAC_DHR8Rx、DAC_DHR12L x 、DAC_DHR12R x 、DAC_DHR8RD、DAC_DHR12RD、または DAC_DHR12LD への書込み操作）、DAC チャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない（DAC_CR レジスタの TEN x ビットがリセットされている）場合に、1 dac_hclk クロックサイクル後に DAC_DORx レジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている（DAC_CR レジスタの TENx ビットがセットされている）ときには、トリガが発生すると、転送はトリガ信号の 3 dac_hclk クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される $t_{SETTLING}$ 時間後にアナログ出力電圧が使用可能になります。

dac_hclk クロック速度が 80 MHz を超える場合は、DAC_MCR の HFSEL ビットをセットする必要があります。それによって、DAC_DHRx レジスタから DAC_DORx レジスタへの転送に余分の遅延が追加されます。

HFSEL ビットと dac_hclk クロック周波数に応じた DAC_DORx の更新速度の制限については、この後の HFSEL の説明を参照してください。

許されない期間にデータが更新された場合、またはソフトウェア/ハードウェア･トリガ･イベントが発生した場合のペリフェラルの動作は予期できません。

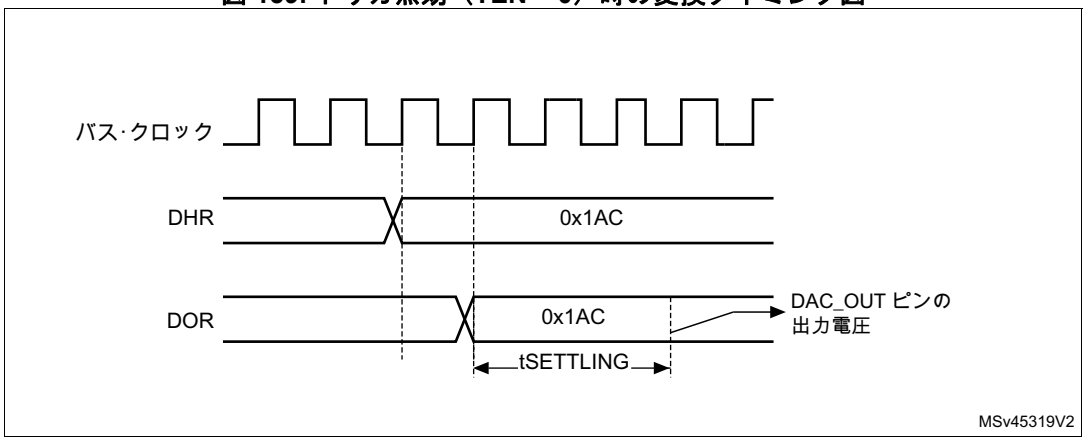
上記のタイミングは、DAC インタフェースの制限にのみ関係します。製品データシートの $t_{SETTLING}$ パラメータの値も参照してください。

表 188. HFSEL の説明

HFSEL[1:0]	AHB 周波数	機能
00	≤80 MHz	DAC_DOR の更新速度は最大 3 AHB クロックサイクル
01	>80 MHz(1)	DAC_DOR の更新速度は最大 5 AHB クロックサイクル
10	>160 MHz	DAC_DOR の更新速度は最大 7 AHB クロックサイクル
11	予約済みです。	-

1. 最大の AHB 周波数の値については、デバイスデータシートを参照してください。

図 159. トリガ無効（TEN = 0）時の変換タイミング図



22.4.6 D/A コンバータ出力電圧

デジタル入力は、0 から V_{REF+} までのリニア変換で出力電圧に変換されます。

各 DAC チャネルピンのアナログ出力電圧は、次の式によって求められます。

$$DACoutput = V_{REF} \times \frac{DOR}{4096}$$

22.4.7 DAC トリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割込みラインなど）によって変換をトリガできます。DAC_CR レジスタの TSELx[3:0] ビットに示すように、TSELx[3:0] 制御ビットは、変換をトリガするイベントを 16 のイベントから決定します。これらのイベントは、ソフトウェアトリガまたはハードウェアトリガのいずれかです。セクション 22.4.2：DAC ピンおよび内部信号の相互接続の表を参照してください。

DAC インタフェースが選択されたトリガソース（次の表を参照）で立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータがDAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから 3 dac_hclk サイクル後に更新されます。

ソフトウェアトリガが選択されている場合、変換は、SWTRIG ビットがセットされると開始されます。SWTRIG ビットは、DAC_DORx レジスタの内容が DAC_DHRx レジスタにロードされると、ハードウェアによってリセットされます。

のこぎり波生成のリセットトリガ選択およびインクリメントトリガ選択は、それぞれ STRSTTRIGSELx および STINCTRIGSELx 制御ビットによって行われます。STRSTTRIGSELx のマッピングは、TSELx と同様です。TSELx、STRSTTRIGSELx、STINCTRIGSELx のマッピングについては、セクション 22.4.2：DAC ピンおよび内部信号を参照してください。

注： **ENx ビットがセットされているときには、TSELx[3:0] ビットを変更できません。**

ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DORx レジスタへの転送は、わずか 1 dac_hclk クロックサイクルで行われます。

22.4.8 DMA リクエスト

各 DAC チャネルは、DMA 機能を備えています。DAC チャネルの DMA リクエストは、2 つの DMA チャネルを使用して処理されます。

DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガ以外）が発生した場合、DAC_DHRx レジスタの値は DAC_DORx レジスタに転送され、転送が終了すると、DMA リクエストが生成されます。

デュアルモードでは、両方の DMAENx ビットがセットされている場合、2 つの DMA リクエストが生成されます。1つの DMA リクエストしか必要ない場合には、対応する DMAENx ビットだけをセットする必要があります。これにより、アプリケーションは 1 つの DMA リクエストと一意な DMA チャネルを使用して、両方の DAC チャネルをデュアルモードで管理することがきます。

DAC_DHRx から DAC_DORx へのデータ転送は DMA リクエストの前に起きるので、最初のトリガイベントが発生する前に最初のデータを DAC_DHRx に書き込む必要があります。

DMA アンダーラン

DAC DMA リクエストはキューされないので、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に 2 つ目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタの DMA チャネル x アンダーランフラグ DMAUDRx がセットされてエラー条件を報告します。DAC チャネル x は、古いデータを変換し続けます。

ソフトウェアは、1を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし、DMA と D/A コンバータの両方のチャネル x を再初期化して転送を正しくリスタートさせる必要があります。また、D/A コンバータトリガ変換周波数を変更するか DMA の負荷を軽減して、新しい DMA アンダーランを回避する必要があります。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャネル x では、DAC_CR レジスタの対応する DMAUDRIEx ビットが有効にされている場合、割込みも生成されます。

DMA ダブルデータモード

DMA コントローラが通常モードで使用されるときには、DMA リクエストによって 12 ビット（または 8 ビット）データのみが転送されます。AHB 幅は 32 ビットなので、2 つの 12 ビットデータを同時に転送できます。このモードを使用するには、DAC_MCR レジスタの DMADOUBLEx ビットをセットします。

DAC DMA リクエストは、DMAENx ビットがセットされているとき、外部トリガ（ソフトウェアトリガでなく）が 2 つ発生するたびに生成されます。

1. 最初のトリガが検出されると、DAC_DHRx および DAC_DHRB x レジスタの値が DAC_DORx および DAC_DORB x レジスタに転送されます。実際の DAC データが DAC_DORx レジスタにロードされます。その後、DMA リクエストが生成されます。DMA は、新しいデータを DAC_DHR x および DAC_DHRB x データレジスタに書き込みます。
2. 次のトリガが検出されると、実際の DAC データが DAC_DHRBx レジスタにロードされます。この 2 つ目のトリガでは、DMA リクエストは生成されません。DORSTATx ビットは、アナログ DAC 入力に実際にロードされる DOR データを示します。

DMA ダブルデータモードでは、DMA アンダーラン機能もサポートされます。

DMA ダブルモードでは、DMA リクエストは 1 つの DAC チャネルのみを処理できます。DMA ダブルモードで 2 つのチャネル出力を使用するには、各 DMA チャネルを個別に設定する必要があります。

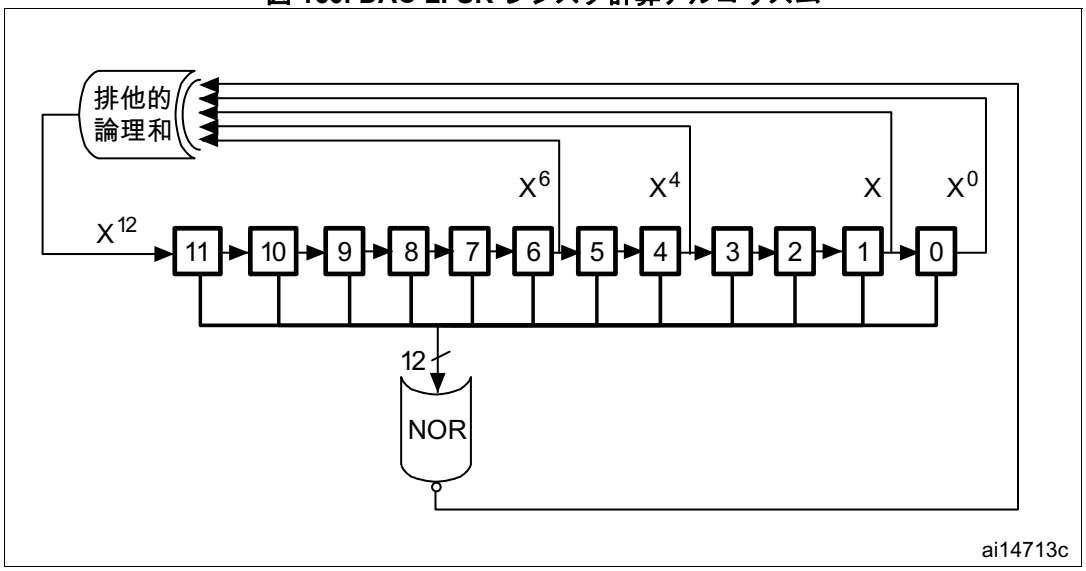
ダブルデータモードからシングルデータモードに、またはその逆に変更するには、以下の条件が満たされている必要があります。

- DAC を無効にする必要があります。
- DMAEN ビットをクリアする必要があります（ENx = 0 かつ DMAEN = 0）。

22.4.9 ノイズ生成

リニアフィードバックシフトレジスタ（LFSR）を使用して、可変振幅の擬似ノイズを生成することができます。D/A コンバータノイズ生成を選択するには、WAVEx[1:0] を 01にセットします。LFSR にプリロードされる値は 0xAAA です。このレジスタは、各トリガイベントの 3 dac_hclk クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図 160. DAC LFSR レジスタ計算アルゴリズム

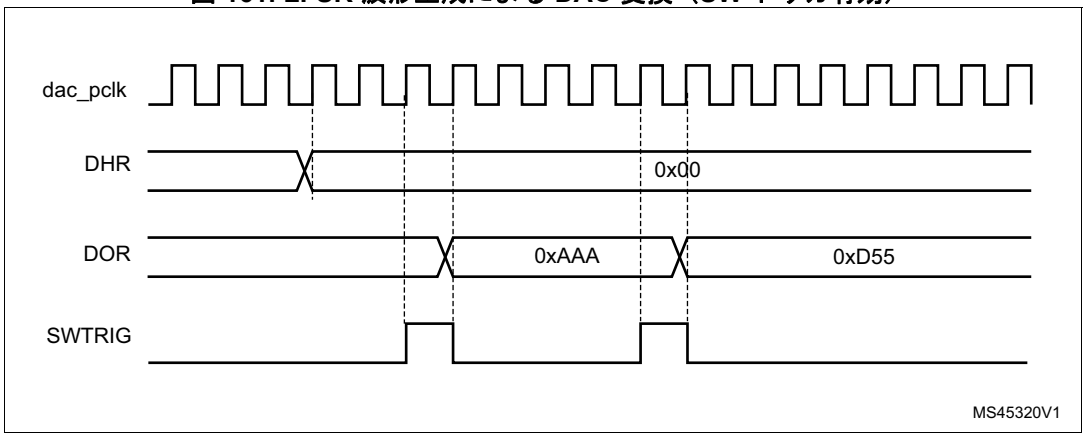


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算され、DAC_DORx レジスタに転送されます。

LFSR が 0x0000 の場合、“1”がインジェクトされます（アンチロックアップメカニズム）。

WAVEx[1:0] ビットをリセットすることによって、LFSR 波形生成をリセットできます。

図 161. LFSR 波形生成による DAC 変換（SW トリガ有効）



注： **ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。**

22.4.10 三角波生成

DC または低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC 三角波生成を選択するには、WAVEx[1:0] を 10 にセットします。振幅は、DAC_CR レジスタの MAMPx[3:0] ビットによって設定されます。内部三角波カウンタは、各トリガイベントの 3 dac_hclk クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしに DAC_DHRx レジスタに加えられ、合計は DAC_DORx レジスタに転送されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは 0 にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波形生成をリセットできます。

図 162. DAC 三角波生成

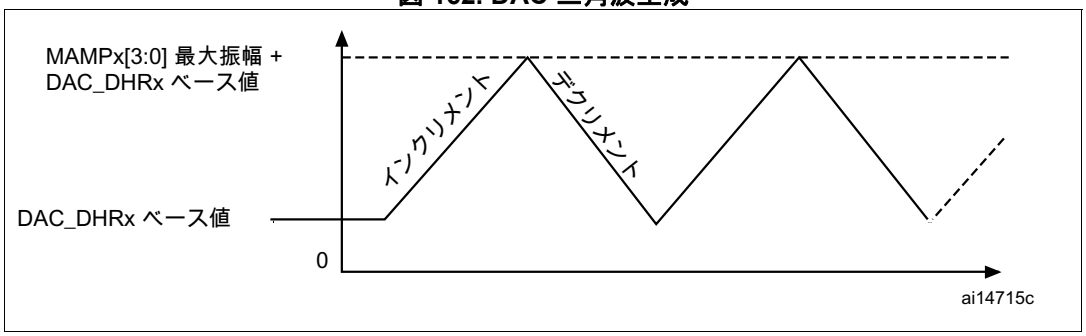
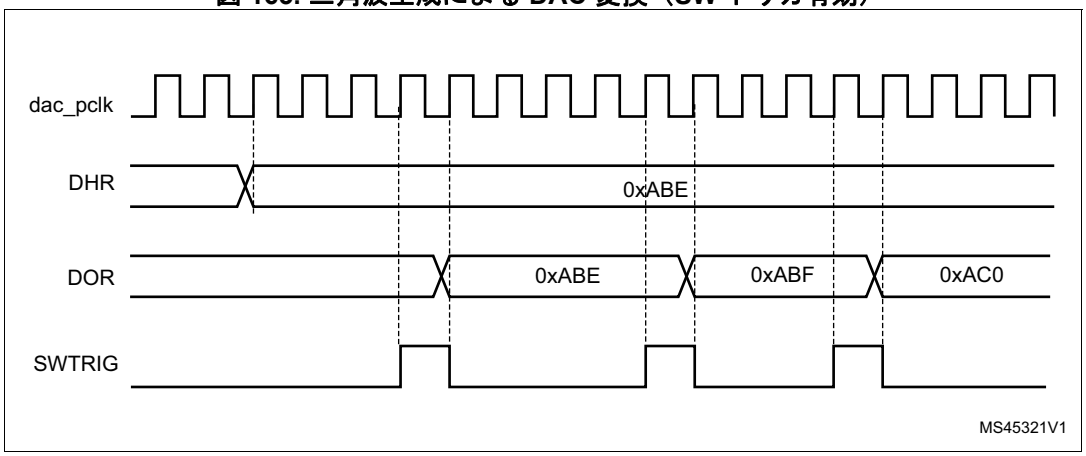


図 163. 三角波生成による DAC 変換（SW トリガ有効）



注： **三角波生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。**

DAC を有効にするには、その前に MAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

22.4.11 D/A コンバータによるのこぎり波生成

D/A コンバータでのこぎり波形を生成できます。初期値、インクリメント値、および方向制御について特定のレジスタ設定が必要です。

- DAC のこぎり波生成を選択するには、DAC_CR レジスタの WAVEx[1:0] を 11 にセットします。
- のこぎり波カウンタの初期値（リセット値）は、DAC_STR x レジスタの STRSTDATA x [11:0] ビットで設定されます。
- インクリメント値は、DAC_STR x レジスタの STINCDATA x [15:0] ビットによって定義されます。
- のこぎり波の方向は、DAC_STR x レジスタの STDIR x ビットによって定義されます。

のこぎり波カウンタは STRSTDATAx[11:0] から開始し（ビット 12～15 は 0000 にセットされます）、その後、各インクリメントトリガが STINCDATAx[15:0] 値をインクリメント（またはデクリメント）します。

DAC 出力は、それらのカウンタ値の 12 MSB から使用されます。カウンタが 0x0000 または 0xFFFF に達すると、値は飽和します。のこぎり波リセットトリガ信号は、カウンタ値を STRSTDATAx[11:0] 値に初期化します（ビット 12～15 は 0000 にセットされます）。

インクリメントトリガおよびリセットトリガは、STINCTRIGSELx[3:0] および STRSTTRIGSELx[3:0] ビットによって選択されなければなりません。

図 164. D/A コンバータによるのこぎり波生成（STDIR x = 0）

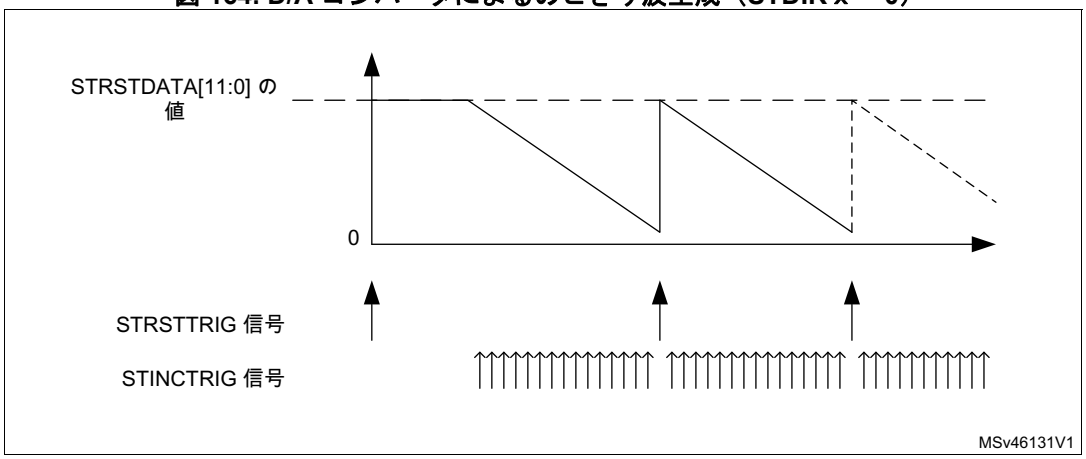
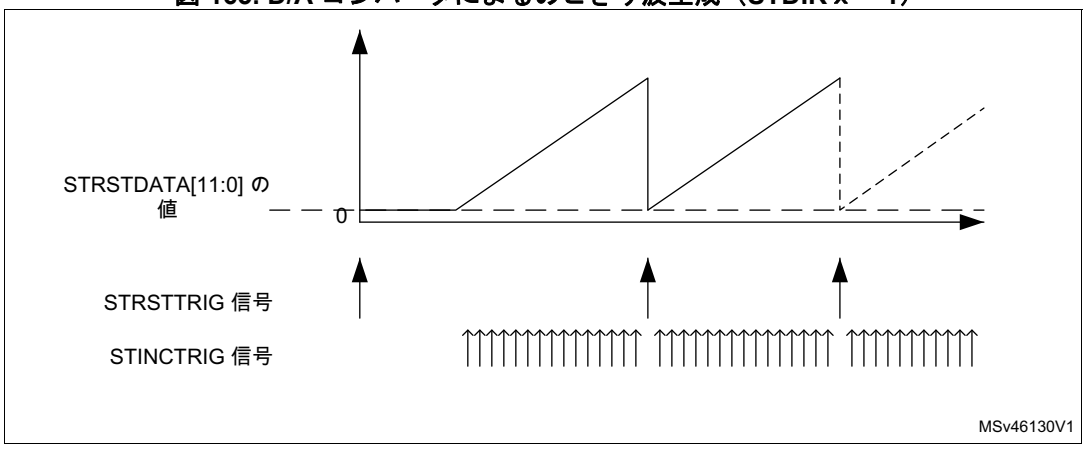
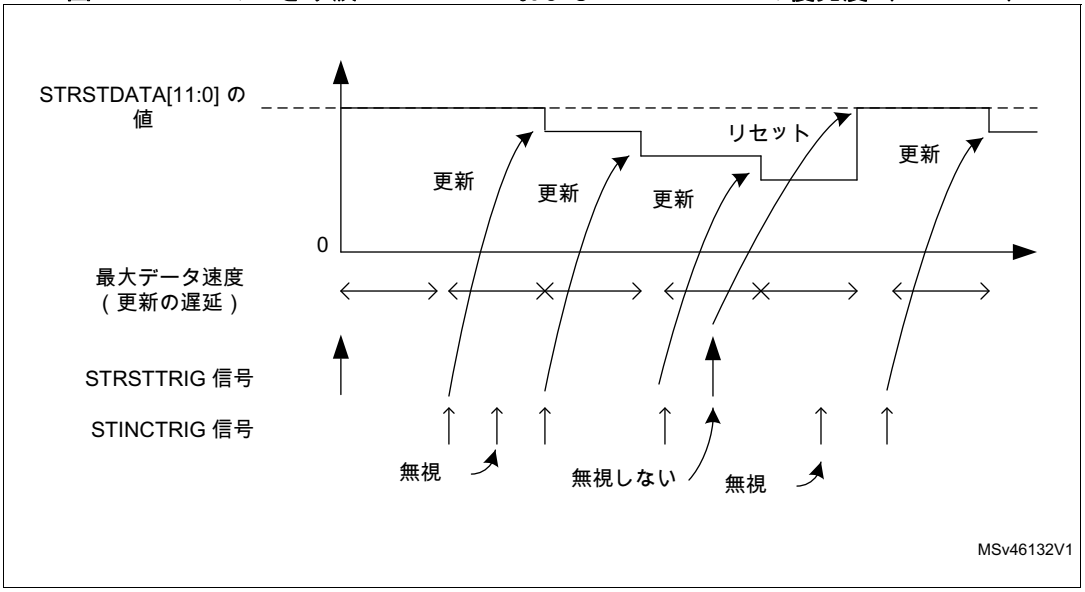


図 165. D/A コンバータによるのこぎり波生成（STDIR x = 1）



STRSTTRIG 信号は、STINCTRIG より優先されます。表 188 : HFSEL の説明で定義した DAC_DORx の更新速度よりトリガ信号を速くすることはできません。STINCTRIG が許されるデータ更新速度より速くアサートされた場合、STINCTRIG トリガは無視されます。STRSTTRIG 信号が STINCTRIG の後、DAC_DORx 更新速度の制限の前に適用された場合、STRSTTRIG は保留状態になります。その後、データインクリメントの直後に、リセットトリガが適用されます。

図 166. DAC のこぎり波 STINCTRIG および STRSTTRIG の優先度（STDIR=0）



22.4.12 DAC チャネルモード

各 D/A コンバータ･チャネルは、通常モードまたはサンプル･アンド･ホールド･モードのいずれかに設定できます。駆動力を高くするために出力バッファを有効にできます。出力バッファを有効にする前に、電圧オフセットを較正する必要があります。この較正は出荷時に実施され（リセット後にロードされ）、アプリケーション動作中にソフトウェアで調整できます。

通常モード

通常モードには、バッファ状態の変更と DACx_OUTy ピン相互接続の変更による 4 つの組み合わせがあります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のように設定する必要があります。

- 000：DAC を外部ピンに接続
- 001：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 010：DAC を外部ピンに接続
- 011：DAC をオンチップペリフェラルに接続

サンプルおよびホールドモード

サンプルおよびホールドモードでは、D/A コンバータコアがトリガ変換でデータを変換してから、コンデンサで変換された電圧を保持します。変換しない場合、DAC コアおよびバッファはサンプル間で完全にオフになり、DAC 出力はトライステートになるため、全体の消費電力を低減します。新しい変換を行う前ごとに、安定時間（その値はバッファ状態に応じて変わります）が必要になります。

このモードでは、D/A コンバータのコアおよびすべての対応するロジックとレジスタを dac_hclk クロックのほか、LSI または LSE 低速クロック（dac_hold_ck）でも駆動できるので、STOP モードなどの超低電力モードで D/A コンバータ･チャネルを使用できます。

サンプル･アンド･ホールド･モードが有効な場合、LSI または LSE 低速クロック（dac_hold_ck）を停止しないでください。

サンプル･アンド･ホールド･モードの動作は、3 つのフェーズに分けられます。

1. サンプルフェーズ : サンプル／ホールド要素が対象の電圧までチャージされます。チャージ時間はコンデンサ値（ユーザによって選択された内部または外部）によって異なります。サンプリング時間は、DAC_SHSRx レジスタの TSAMPLEx[9:0] ビットで設定されます。TSAMPLEx[9:0] ビットの書込み中、両方のクロックドメイン（AHB および低速クロック）を同期するために DAC_SR レジスタの BWSTx ビットが 1にセットされ、D/A コンバータチャネル動作中にソフトウェアによってサンプルフェーズの値を変更できるようになります。
2. ホールドフェーズ : DAC 出力チャネルはトライステートになり、DAC コアおよびバッファはオフになって消費電流を低減します。ホールド時間は、DAC_SHHR レジスタの THOLDx[9:0] ビットで設定されます。
3. リフレッシュフェーズ : リフレッシュ時間は、DAC_SHRR レジスタの TREFRESHx[7:0] ビットで設定されます。

上記 3 つのフェーズのタイミングは、LSI/LSE クロックの周期単位で定義されます。たとえば、LSI/LSE の周波数として 32 KHz 以下を選択した場合、350 µs のサンプル時間、2 ms のホールド時間、100 µs のリフレッシュ時間は次のように設定します。

12 サイクルがサンプルフェーズで必要になります。TSAMPLEx[9:0] = 11

62 サイクルがホールド･フェーズで必要になります。THOLDx[9:0] = 62

4 サイクルがリフレッシュ期間で必要になります。TREFRESHx[7:0] = 4

この例では、消費電力が通常モードに対してほぼ 15 分の 1 に低減されます。

適切なサンプルおよびリフレッシュタイミングを計算する公式は、以下の表に記載しており、ホールド時間はリーク電流によって異なります。

表 189. サンプルおよびリフレッシュタイミング

バッファ状態	t_{SAMP}(1)(2)	$t_{REFRESH}$ (2)(3)
イネーブル	7 µs + (10*R_{BON}*C_{SH})	7 µs + (R_{BON}*C_{SH})*ln(2*N_{LSB})
無効化	3 µs + (10*R_{BOFF}*C_{SH})	3 µs + (R_{BOFF}*C_{SH})*ln(2*N_{LSB})

1. 上記の公式で、½ LSBエラー精度での目標のコード値までの整定には、12 ビットの分解能の場合、10 定数時間が必要です。8 ビット分解能の場合、整定時間は 7 定数時間です。
2. C_{SH} はサンプルおよびホールドモードでのコンデンサ容量です。
3. ホールドフェーズ中に許容される電圧低下「Vd」は、コンデンサが出力リーク電流で放電した後の LSB の数で表されます。½ LSB エラー精度で目標のコード値まで整定し直すには、DAC の ln(2*Nlsb) 定数時間が必要です。

出力バッファがオンの場合のサンプルおよびリフレッシュ時間の計算例

以下に使用される値は、例示のみを目的としています。製品データについては、製品データシートを参照してください。

C_{SH} = 100 nF

V_{DDA} = 3.0 V

サンプリング･フェーズ：

t_{SAMP} = 7 µs + (10 * 2000 * 100 * 10^{-9}) = 2.007 ms

（ここで、R_{BON} = 2 kΩ）

リフレッシュ･フェーズ：

$t_{REFRESH}$ = 7 µs + (2000 * 100 * 10^{-9}) * ln(2*10) = 606.1 µs

（ここで、N_{LSB} = 10（ホールド･フェーズ中に 10 個の LSB をドロップ））

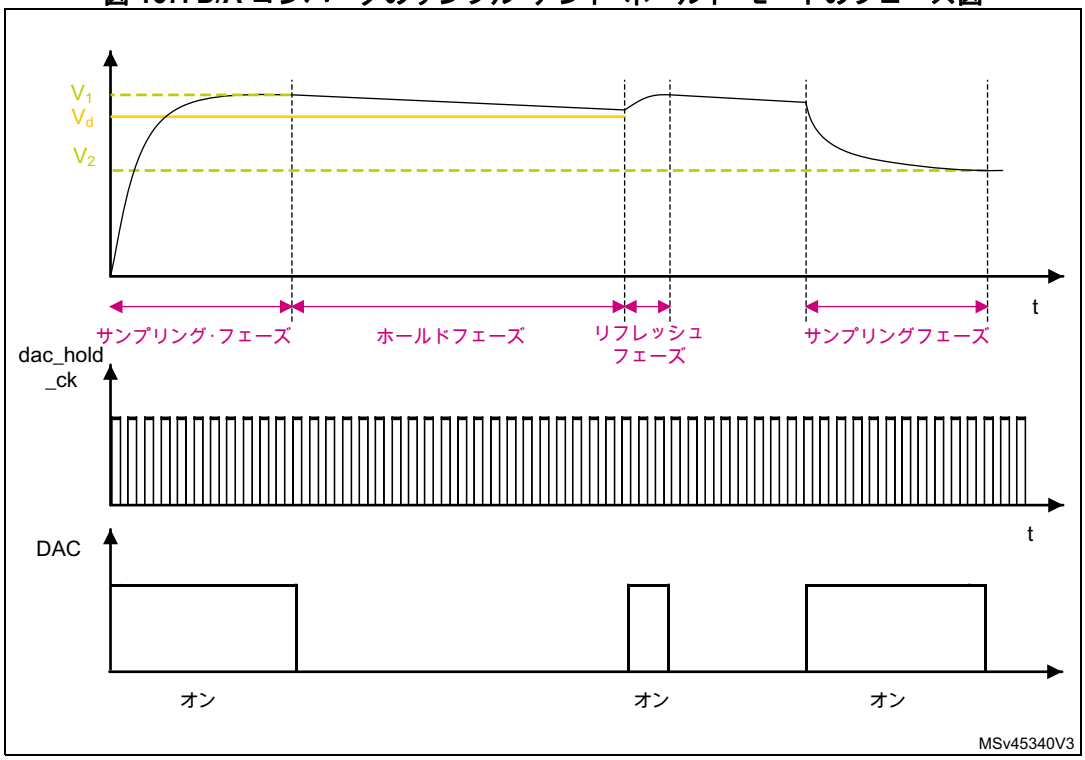
ホールド･フェーズ：

$D_v = i_{leak} * t_{hold} / C_{SH}$ = 0.0073 V（3 V で 12 ビットの 10 LSB）

i_{leak} = 150 nA（全温度範囲にわたる IO リークの最悪値）

t_{hold} = 0.0073 * 100 * 10^{-9} / (150 * 10^{-9}) = 4.867 ms

図 167. D/A コンバータのサンプル･アンド･ホールド･モードのフェーズ図



通常モードと同様に、サンプルおよびホールドモードには個別の設定があります。

出力バッファを有効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 100：DAC を外部ピンに接続
- 101：DAC を外部ピンとオンチップペリフェラルに接続

出力バッファを無効にするには、DAC_MCR レジスタの MODEx[2:0] ビットを次のようにセットする必要があります。

- 110：D/A コンバータを外部ピンとオンチップ･ペリフェラルに接続
- 111：DAC をオンチップペリフェラルに接続

  MODEx[2:0] ビットが 111と等しい場合、内部コンデンサ C_{Lint} は、D/A コンバータコアの電圧出力を保持し、それによってオンチップペリフェラルを駆動します。

すべてのサンプルおよびホールドフェーズは割込み可能であり、DAC_DHRx のあらゆる変更は新しいサンプルフェーズを即座にトリガします。

注：　のこぎり波生成は、サンプル･アンド･ホールド･モード動作ではサポートされません。

表 190. チャネル出力モードの概要

MODEx[2:0]			モード	バッファ	出力接続
0	0	0	通常モード	有効	外部ピンに接続
0	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
0	1	0		無効	外部ピンに接続
0	1	1			オンチップペリフェラルに接続（コンパレータなど）
1	0	0	サンプルおよびホールドモード	有効	外部ピンに接続
1	0	1			外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	0		無効	外部ピンとオンチップペリフェラルに接続（コンパレータなど）
1	1	1			オンチップペリフェラルに接続（コンパレータなど）

22.4.13 DAC 出力バッファ較正

N ビットのデジタルアナログコンバータ（DAC）の転送関数は次のとおりです。

$$V_{out} = ((D/2^N) \times G \times V_{ref}) + V_{OS}$$

ここで、V_{OUT} はアナログ出力、D はデジタル入力、G はゲイン、V_{ref} は公称フルスケール電圧、V_{os} はオフセット電圧です。理想的な D/A コンバータ･チャネルでは、G = 1、V_{os} = 0 となります。

出力バッファの特性により、オフセット電圧が部品間で異なり、アナログ出力に絶対オフセット誤差が生じる場合があります。V_{os} を補正するには、トリミング技法による較正が必要です。

この較正は、D/A コンバータ･チャネル x がバッファ有効状態で動作している場合（MODEx[2:0] = 000b または 001b または 100b または 101b）にのみ有効です。バッファがオフの他のモードで適用しても効果はありません。較正中は次のようになります。

- バッファ出力がピンの内部／外部接続から切断され、トライステートモード（HiZ）になります。
- バッファがミドルコード値 0x800を検出して、内部ブリッジによる VREF+/2信号と比較するためにコンパレータとして機能し、比較結果（CAL_FLAGx ビット）に応じて出力信号を 0または 1にトグルします。

次の 2 つの較正技法があります。

- 出荷時トリミング（デフォルト設定）

  DAC バッファオフセットが出荷時にトリミングされます。DAC_CCR レジスタの OTRIMx[4:0] ビットのデフォルト値は出荷時トリミング値となり、DAC デジタルインタフェースがリセットされるとロードされます。

- ユーザトリミング

  動作条件が出荷時トリミング条件と異なる場合、特に V_{DDA} 電圧、温度、VREF+ の値が変化する場合に、ユーザトリミングはソフトウェアによって適用時にいつでも実行できます。

注： **出荷時トリミング条件の詳細については、データシートを参照してください。**

また、VDD が削除される場合（デバイスが STANDBY モードや VBAT モードに移行する場合など）、較正が必要になります。

ユーザトリミング較正を実行する手順は次のとおりです。

1. DAC チャネルがアクティブな場合、DAC_CR の ENx ビットに 0 を書き込み、チャネルを無効にします。
2. DAC_MCR レジスタの MODEx[2:0] = 000b または 001b または 100b または 101b を書き込んで、バッファが有効なモードを選択します。
3. DAC_CR レジスタの CENx ビットを 1 に設定して、D/A コンバータ･チャネル x の較正を開始します。
4. トリミングアルゴリズムを適用します。
   a) OTRIMx[4:0] ビットにコード（00000b で開始する）を書き込みます。
   b) tTRIM の遅延を待ちます。
   c) DAC_SR の CAL_FLAGx ビットが 1 にセットされているか確認します。
   d) CAL_FLAGx が 1にセットされている場合、OTRIMx[4:0] トリミングコードが検索され、これを出力値を補正するデバイスの操作中に使用できます。それ以外は OTRIMx[4:0] をインクリメントし、（a）から（d）のサブステップを再び繰り返します。

ソフトウェアアルゴリズムによって、逐次比較や二分岐法を使用して、より早く OTRIMx[4:0] ビットの内容を計算してセットできます。

CAL_FLAGx ビットの転流／トグルは、オフセットが正しく補正されたことを示し、対応するトリミングコードを DAC_CCR レジスタの OTRIMx[4:0] ビットに保持しておく必要があります。

注： **tTRIM 遅延が、OTRIMx[4:0] ビットへの書込みと DAC_SR レジスタの CAL_FLAGx ビットの読出しの間で守られなければ、正しい値を取得できません。このパラメータは、データシートの電気的特性のセクションで指定されています。**

V_{DDA}、VREF+、および温度条件がデバイス動作中に変化せず、STANDBY および VBAT モードに頻繁に移行する場合、ソフトウェアは最初のユーザ較正時に見つかった OTRIMx[4:0] ビットを Flash またはバックアップレジスタに格納して、後でデバイスの電源が復帰したときに直接ロード／書き込むことができ、新しい較正時間を待たなくて済むようになります。

CENx ビットがセットされているとき、ENx ビットをセットすることはできません。

22.4.14 デュアル D/A コンバータチャネル変換モード（デュアルチャネルを使用できる場合）

同時に 2 つの DAC チャネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、および DHR12LD の 3 つのデュアルレジスタが搭載されています。両方の DAC チャネルを同時に駆動するには、一意なレジスタアクセスが必要です。波形の生成の場合は、DHRxxxD レジスタへのアクセスは必要ありません。したがって、2 つの出力チャネルは独立に、または同時に使用できます。

2 つの DAC チャネルとこれらのデュアルレジスタを使用することで、15 の変換モードが使用可能です。すべての変換モードは、必要な場合には、別の DHRx レジスタを使っても利用できます。

すべてのモードについて、以下の節で説明します。

波形生成なしの独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、DHR1 レジスタが DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DHR2 レジスタが DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。

1 つの LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMP1[3:0] ビットと MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 1x に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1および TSEL2ビットに異なる値をセットすることによって、異なるトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 1x に、異なる最大振幅値を MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

DAC チャネル 1 トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

DAC チャネル 2 トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

1 つののこぎり波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. STRSTTRIGSEL1[3:0]、STRSTTRIGSEL2[3:0]、STINCTRIGSEL2[3:0]、および STINCTRIGSEL1[3:0] ビットに異なる値をセットすることによって、異なるトリガソースを設定します。
2. 2 つの DAC チャネル WAVEx[1:0] ビットを 11 に設定、各レジスタについて同じ STRSTDATAx[11:0]、STINCDATAx[15:0]、および STDIRx 値をセットします。

DAC チャネル 1 トリガが発生すると、DAC チャネル 1 のこぎり波カウンタが DHR1 レジスタを更新して、DAC_DOR1 に転送します（3 AHB クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DAC チャネル 2 のこぎり波カウンタが DHR1 レジスタを更新して、DAC_DOR1 に転送します（3 AHB クロックサイクル後）。

異なるのこぎり波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. STRSTTRIGSEL1[23:0]、STRSTTRIGSEL2[23:0]、STINCTRIGSEL2[3:0]、および STINCTRIGSEL1[3:0] ビットに異なる値をセットすることによって、異なるトリガソースを設定します。
2. 2 つの DAC チャネル WAVEx[1:0] ビットを 11 に設定し、各レジスタについて異なる STRSTDATAx[11:0]、STINCDATAx[15:0]、および STDIRx 値をセットします。

DAC チャネル 1 トリガが発生すると、DAC チャネル 1 のこぎり波カウンタが DHR1 レジスタを更新して、DAC_DOR1 に転送します（3 AHB クロックサイクル後）。

DAC チャネル 2 トリガが発生すると、DAC チャネル 2 のこぎり波カウンタが DHR2 レジスタを更新して、DAC_DOR1 に転送します（3 AHB クロックサイクル後）。

同時ソフトウェア開始

DAC をこの変換モードに設定するには、次の手順が必要です。

- デュアル DAC チャネルデータを目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にロードします。

この設定では、1 dac_hclk クロックサイクル後に、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます。

波形生成なしの同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. デュアル DAC チャネルデータを目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にロードします。

トリガが発生すると、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます（3 dac_hclk クロックサイクル後）。

1 つの LFSR 生成による同時トリガ

1. DAC をこの変換モードに設定するには、次の手順が必要です。
2. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
3. TSEL1 および TSEL2 ビットフィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
4. 2 つの DAC チャネル WAVEx[1:0] ビットを 01 に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
5. デュアル DAC チャネルデータを目的の DHR レジスタ（DHR12RD、DHR12LD、または DHR8RD）にロードします。

トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。同時に、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

異なる LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 01 に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる LFSR マスク値をセットします。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR1 カウンタが更新されます。
同時に、MAMP2[3:0] で設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、LFSR2 カウンタが更新されます。

1 つの三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネルの WAVEx[1:0] ビットを 1x に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、同じ三角波振幅を持つ DAC チャネル1 の三角波カウンタが DHR 1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、同じ三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

異なる三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. TEN1 と TEN2 の 2 つの DAC チャネルトリガイネーブルビットをセットします。
2. TSEL1 および TSEL2 ビットフィールドに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
3. 2 つの DAC チャネル WAVEx[1:0] ビットを 1x に、異なる最大振幅値を MAMP1[3:0] と MAMP2[3:0] ビットに設定します。
4. 目的の DHR レジスタ（DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD）にデュアル DAC チャネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャネル 1 の三角波カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます（3 AHB クロックサイクル後）。その後、DAC チャネル1 の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つ DAC チャネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます（3 dac_hclk クロックサイクル後）。その後、DAC チャネル2 の三角波カウンタが更新されます。

1 つののこぎり波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. STRSTTRIGSEL1[3:0]、STRSTTRIGSEL2[3:0]、STINCTRIGSEL2[3:0]、および STINCTRIGSEL1[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルに同じトリガソースを設定します。
2. 2 つの DAC チャネル WAVEx[1:0] ビットを 11 に設定し、各レジスタについて同じ STRSTDATAx[11:0]、STINCDATAx[15:0]、および STDIRx 値をセットします。

トリガが発生すると、DAC チャネル 1/2 のこぎり波カウンタが DHR1 および DHR2 レジスタを更新して、DAC_DOR1/2 に転送します（3 AHB クロックサイクル後）。

異なるのこぎり波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

1. STRSTTRIGSEL1[3:0]、STRSTTRIGSEL2[3:0]、STINCTRIGSEL2[3:0]、および STINCTRIGSEL1[3:0] ビットに同じ値をセットすることによって、両方の DAC チャネルについて同じトリガソースを設定します。
2. 2 つの DAC チャネル WAVEx[1:0] ビットを 11 に設定し、各レジスタについて異なる STRSTDATAx[11:0]、STINCDATAx[15:0]、および STDIRx 値をセットします。

トリガが到達すると、D/A コンバータ･チャネルの 1/2 のこぎり波カウンタによって DHR1 レジスタと DHR2 レジスタが更新され、それらの値が DAC_DOR1/2 に個別にロードされます（3 AHB クロック･サイクル後）。

22.5 低電力モードでの D/A コンバータ

表 191. 低電力モードが D/A コンバータに与える影響

モード	説明
SLEEP	影響はありません。D/A コンバータが DMA と使用されます。
LPRun	影響はありません。
LPSleep	影響はありません。DAC が DMA と使用されます。
STOP 0 / STOP 1	LSI/LSE クロックによるサンプル･アンド･ホールド･モードを選択している場合、D/A コンバータはスタティック値でアクティブな状態を維持します。
STANDBY	DAC ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

22.6 D/A コンバータの割込み

表 192. D/A コンバータの割込み

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
D/AC	DMA アンダーラン	DMAUDRx	DMAUDRIEx	DMAUDRx = 1を書き込む	可能	不可	不可

22.7 DAC レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1 を参照してください。

ペリフェラル・レジスタには、ワード（32 ビット）単位でアクセスする必要があります。

22.7.1 DAC 制御レジスタ（DAC_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2 [1:0]		TSEL2[3]	TSEL2[2]	TSEL2[1]	TSEL2[0]	TEN2	EN2
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1 [1:0]		TSEL1[3]	TSEL1[2]	TSEL1[1]	TSEL1[0]	TEN1	EN1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **CEN2**：DAC チャネル 2 較正イネーブル

このビットは、DAC チャネル 2 の較正を有効化/無効化するためにソフトウェアによってセット/クリアされます。DAC_CR で EN2 ビットが 0 にセットされている場合のみ書込み可能で（DAC チャネルが無効化された場合のみ較正モードに移行／終了できます）、それ以外の書込み操作は無視されます。

0：DAC チャネル 2 は通常動作モードです。

1：DAC チャネル 2 は較正モードです。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 29 **DMAUDRIE2**：DAC チャネル 2 DMA アンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 DMA アンダーラン割込みは無効です。

1：DAC チャネル 2 DMA アンダーラン割込みは有効です。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 28 **DMAEN2**：DAC チャネル2 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 DMA モードは無効です。

1：DAC チャネル2 DMA モードは有効です。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 27:24 **MAMP2[3:0]**：DAC チャネル2 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。
0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。
0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。
0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。
0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。
0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。
0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。
0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。
1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。
1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。
1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。
≥1011: LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

注： **これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。**

ビット 23:22 **WAVE2[1:0]**：DAC チャネル2 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／リセットされます。

00：波形生成は無効です。
01：ノイズ波生成は有効です。
10：三角波生成は有効です。
11：のこぎり波生成は有効です。

注： **ビット TEN2＝1（DAC チャネル2 トリガ有効）の場合のみ使用されます。**
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 21:18 **TSEL2[3:0]**：DAC チャネル2 トリガ選択

これらのビットは、DAC チャネル2 をトリガするために使用される外部イベントを選択します。

0000：SWTRIG2
0001：dac_ch2_trg1
0010：dac_ch2_trg2
.....
1111：dac_ch2_trg15

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

注： **ビット TEN2 = 1（D/A コンバータ･チャネル 2 トリガ有効）の場合のみ使用されます。**
これらのビットは、デュアル･チャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 17 **TEN2**：DAC チャネル2 トリガイネーブル

このビットは、DAC チャネル2 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 2 トリガは無効であり、DAC_DHR2 レジスタに書き込まれたデータは、1 dac_hclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。
1：DAC チャネル 2 トリガは有効であり、DAC_DHR2 レジスタからのデータは、3 dac_hclk クロックサイクル後に DAC_DOR2 レジスタに転送されます。

注： **ソフトウェアトリガが選択されているときには、DAC_DHR2 から DAC_DOR2 レジスタへの転送は、わずか 1 dac_hclk クロックサイクルで行われます。**
これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 16 **EN2**：DAC チャネル2 イネーブル

このビットは、DAC チャネル2 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル2 は無効です。

1：DAC チャネル2 は有効です。

注： これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CEN1**：DAC チャネル 1 較正イネーブル

このビットは、D/A コンバータ･チャネル 1 の較正を有効化または無効化するためにソフトウェアでセットまたはクリアされます。DAC_CR のビット EN1 = 0 の場合のみ書込み可能で（較正モードへの移行とその終了は D/A コンバータ･チャネルが無効な場合にのみ可能）、それ以外の書込み操作は無視されます。

0：DAC チャネル 1 は通常動作モードです。

1：DAC チャネル 1 は較正モードです。

ビット 13 **DMAUDRIE1**：DAC チャネル 1 DMA アンダーラン割込みイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA アンダーラン割込みは無効です。

1：DAC チャネル 1 DMA アンダーラン割込みは有効です。

ビット 12 **DMAEN1**：DAC チャネル 1 DMA イネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 DMA モードは無効です。

1：DAC チャネル 1 DMA モードは有効です。

ビット 11:8 **MAMP1[3:0]**：DAC チャネル 1 マスク／振幅セレクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000：LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。

0001：LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。

0010：LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。

0011：LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。

0100：LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。

0101：LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。

0110：LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。

0111：LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。

1000：LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。

1001：LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。

1010：LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。

≥1011: LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 7:6 **WAVE1[1:0]**：DAC チャネル 1 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00：波形生成は無効です。

01：ノイズ波生成は有効です。

10：三角波生成は有効です。

11：のこぎり波生成は有効です。

ビット TEN1 = 1（D/A コンバータ･チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 5:2 **TSEL1[3:0]**：DAC チャネル 1 トリガ選択

これらのビットは、DAC チャネル 1 をトリガするために使用される外部イベントを選択します。

0000：SWTRIG1

0001：dac_ch1_trg1

0010：dac_ch1_trg2

.....

1111：dac_ch1_trg15

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

注： ビット TEN1 = 1（D/A コンバータ・チャネル 1 トリガ有効）の場合のみ使用されます。

ビット 1 **TEN1**：DAC チャネル 1 トリガイネーブル

このビットは、DAC チャネル 1 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 トリガは無効であり、DAC_DHR1 レジスタに書き込まれたデータは、1 dac_hclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

1：DAC チャネル 1 トリガは有効であり、DAC_DHR1 レジスタからのデータは、3 dac_hclk クロックサイクル後に DAC_DOR1 レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHR1 レジスタから DAC_DOR1 レジスタへの転送は、わずか 1 dac_hclk クロックサイクルで行われます。

ビット 0 **EN1**：DAC チャネル 1 イネーブル

このビットは、DAC チャネル 1 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：DAC チャネル 1 は無効です。

1：DAC チャネル 1 は有効です。

22.7.2 DAC ソフトウェアトリガレジスタ（DAC_SWTRGR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIGB2	SWTRIGB1
														w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
														w	w

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **SWTRIGB2**：DAC チャネル 2 ソフトウェアトリガ B

このビットは、ソフトウェアトリガモードで DAC をトリガするために（のこぎり波生成）、ソフトウェアによってセットされます。このビットはハードウェアによってクリアされます。

0：トリガなし

1：のこぎり波インクリメント用のトリガ

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 16 **SWTRIGB1**：DAC チャネル 1 ソフトウェアトリガ B

このビットは、ソフトウェアトリガモードで DAC をトリガするために（のこぎり波生成）、ソフトウェアによってセットされます。このビットはハードウェアによってクリアされます。

0：トリガなし

1：のこぎり波インクリメント用のトリガ

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **SWTRIG2**：DAC チャネル2 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR2 レジスタの値が DAC_DOR2 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_hclk クロックサイクル後）。

このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 0 **SWTRIG1**：DAC チャネル 1 ソフトウェアトリガ

このビットは、ソフトウェアトリガモードで DAC をトリガするために、ソフトウェアによってセットされます。

0：トリガなし

1：トリガ

注： このビットは、DAC_DHR1 レジスタの値が DAC_DOR1 レジスタにロードされると、ハードウェアによってクリアされます（1 dac_hclk クロックサイクル後）。

22.7.3 DAC チャネル 1 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R1）

アドレス･オフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC1DHRB [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC1DHRB[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ B

これらのビットは、ソフトウェアによって書き込まれます。DAC がダブルデータモードで動作しているとき、これらには DAC チャネル 1 の 12 ビットのデータを指定します。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 1 の 12 ビットデータを指定します。

22.7.4 DAC チャネル 1 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L1）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC1DHRB [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC1DHRB[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ B

これらのビットは、ソフトウェアによって書き込まれます。DAC がダブルデータモードで動作しているとき、これらには DAC チャネル 1 の 12 ビットのデータを指定します。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。

これらには、DAC チャネル 1 の 12 ビットデータを指定します。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

22.7.5 DAC チャネル 1 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R1）

アドレス･オフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHRB[7:0]								DACC1DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC1DHRB[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DAC がダブルデータモードで動作しているとき、これらには DAC チャネル 1 の 8 ビットのデータを指定します。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 1 の 8 ビットデータを指定します。

22.7.6 DAC チャネル 2 の 12 ビット右詰めデータ保持レジスタ（DAC_DHR12R2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHRB [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHRB[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DAC が DMA ダブルデータモードで動作しているとき、これらには DAC チャネル 2 の 12 ビットのデータを指定します。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。これらには、DAC チャネル 2 の 12 ビットデータを指定します。

22.7.7 DAC チャネル 2 の 12 ビット左詰めデータ保持レジスタ（DAC_DHR12L2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHRB [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC2DHRB[11:0]**：DAC チャネル 2 の 12 ビット左詰めデータ B

これらのビットは、ソフトウェアによって書き込まれます。DAC がダブルデータモードで動作しているとき、これらには DAC チャネル 2 の 12 ビットのデータを指定します。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

22.7.8 DAC チャネル 2 の 8 ビット右詰めデータ保持レジスタ（DAC_DHR8R2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHRB[7:0]								DACC2DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC2DHRB[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、ソフトウェアによって書き込まれます。DAC がダブルデータモードで動作しているとき、これらには DAC チャネル 2 の 8 ビットのデータを指定します。

ビット 7:0 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

22.7.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ（DAC_DHR12RD）

アドレス･オフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHR[11:0]**：DAC チャネル2 12ビット右詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

22.7.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ（DAC_DHR12LD）

アドレス･オフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DACC2DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR [11:0]												Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

ビット 31:20 **DACC2DHR[11:0]**：DAC チャネル2 12ビット左詰めデータ

これらのビットは、DAC チャネル2 の 12ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]**：DAC チャネル 1 の 12 ビット左詰めデータ

これらのビットは、DAC チャネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

22.7.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ（DAC_DHR8RD）

アドレス･オフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR [7:0]								DACC1DHR [7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **DACC2DHR[7:0]**：DAC チャネル2 8ビット右詰めデータ

これらのビットは、DAC チャネル2 の 8ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 7:0 **DACC1DHR[7:0]**：DAC チャネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

22.7.12 DAC チャネル 1 データ出力レジスタ（DAC_DOR1）

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC1DORB [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC1DORB[11:0]**：DAC チャネル 1 データ出力
これらのビットは読出し専用です。これらには、DAC チャネル 1 B のデータ出力が含まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DOR[11:0]**：DAC チャネル 1 データ出力
これらのビットは読出し専用であり、DAC チャネル 1 のデータ出力を含みます。

22.7.13 DAC チャネル 2 データ出力レジスタ（DAC_DOR2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	DACC2DORB [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **DACC2DORB[11:0]**：DAC チャネル2 データ出力
これらのビットは読出し専用です。これらには、DAC チャネル 2 B のデータ出力が含まれます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC2DOR[11:0]**：DAC チャネル2 データ出力
これらのビットは読出し専用であり、DAC チャネル 2 のデータ出力を含みます。

22.7.14 DAC ステータスレジスタ（DAC_SR）

アドレスオフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BWST2	CAL_FLAG2	DMAUDR2	DORSTAT2	DAC2RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1	r	r											
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BWST1	CAL_FLAG1	DMAUDR1	DORSTAT1	DAC1RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r	r	rc_w1	r	r											

ビット 31 **BWST2**：DAC チャネル 2 ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによってセットされます。このビットは、ソフトウェアが DAC_SHSR2 レジスタに書き込むたびにセットされ、DAC_SHSR2 の書込み操作が完了するとハードウェアによってクリアされます（同期に約 3 LSI/LSE 周期かかります）。

0：進行中の DAC_SHSR2 の書込み操作はありません。DAC_SHSR2 は書込み可能です。

1：進行中の DAC_SHSR2 の書込み操作があります。DAC_SHSR2 は書込みできません。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 30 **CAL_FLAG2**：DAC チャネル 2 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 29 **DMAUDR2**：DAC チャネル2 DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル2 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル 2 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 2 変換を駆動しています）。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 28 **DORSTAT2**：DAC チャネル 2 出力レジスタステータスビット

このビットは、ハードウェアによってセット／クリアされます。これは、DAC がダブルデータモードで動作している場合のみ適用されます。

0：DOR[11:0] が実際の DAC 出力に使用されます。

1：DORB[11:0] が実際の DAC 出力に使用されます。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 27 **DAC2RDY**：DAC チャネル 2 レディステータスビット

このビットは、ハードウェアによってセット／クリアされます。

0：DAC チャネル 2 は、まだトリガも出力データも受け入れる準備ができていません。

1：DAC チャネル 2 は、トリガまたは出力データを受け入れる準備ができています。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 26:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **BWST1**：DAC チャネル 1 ビジー書込みサンプル時間フラグ

このビットは、サンプルおよびホールドモードが有効になった直後にシステムによって、およびソフトウェアが DAC_SHSR1レジスタに書き込むたびにセットされます。DAC_SHSR1の書込み操作が完了するとハードウェアによってクリアされます（同期に約 3 LSI/LSE 周期かかります）。

0：進行中の DAC_SHSR1 の書込み操作はありません。DAC_SHSR1 は書込み可能です。

1：進行中の DAC_SHSR1 の書込み操作があります。DAC_SHSR1 は書込みできません。

ビット 14 **CAL_FLAG1**：DAC チャネル 1 較正オフセットステータス

このビットは、ハードウェアによってセット／クリアされます。

0：較正トリミング値はオフセット較正値未満です。

1：較正トリミング値がオフセット較正値以上です。

ビット 13 **DMAUDR1**：DAC チャネル 1 の DMA アンダーランフラグ

このビットは、ハードウェアによってセットされ、（1 を書き込むことによって）ソフトウェアによってクリアされます。

0：DAC チャネル 1 に DMA アンダーランエラー条件は発生しませんでした。

1：DAC チャネル1 に DMA アンダーランエラー条件が発生しました（現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャネル 1 変換を駆動しています）。

ビット 12 **DORSTAT1**：DAC チャネル 1 出力レジスタステータスビット

このビットは、ハードウェアによってセット／クリアされます。これは、DAC がダブルデータモードで動作している場合のみ適用されます。

0：DOR[11:0] が実際の DAC 出力に使用されます。

1：DORB[11:0] が実際の DAC 出力に使用されます。

ビット 11 **DAC1RDY**：DAC チャネル 1 レディステータスビット

このビットは、ハードウェアによってセット／クリアされます。

0：DAC チャネル 1は、まだトリガも出力データも受け入れる準備ができていません。

1：DAC チャネル 1 は、トリガまたは出力データを受け入れる準備ができています。

ビット 10:0 予約済みであり、リセット値に保持する必要があります。

22.7.15 DAC 較正制御レジスタ（DAC_CCR）

アドレスオフセット：0x38

リセット値：0x00XX 00XX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2 [4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1 [4:0]				
											rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **OTRIM2[4:0]**：DAC チャネル 2 オフセットトリミング値

これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3 : D/A コンバータの実装を参照してください。

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **OTRIM1[4:0]**：DAC チャネル 1 オフセットトリミング値

22.7.16 DAC モード制御レジスタ（DAC_MCR）

アドレスオフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SINFORMAT2	DMADOUBLE2	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]		
						rw	rw						rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HFSEL1	HFSEL0	Res.	Res.	Res.	Res.	SINFORMAT1	DMADOUBLE1	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
rw	rw					rw	rw						rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **SINFORMAT2**：DAC チャネル 2 の符号付きフォーマットの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：入力データは符号なしフォーマットです。

1：入力データは符号付きフォーマット（2 の補数）です。MSB ビットは、符号を表します。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 24 **DMADOUBLE2**：DAC チャネル 2 DMA ダブルデータモード

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA 通常モードが選択されています。

1：DMA ダブルデータモードが選択されています。

注： このビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 23:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **MODE2[2:0]：**DAC チャネル 2 モード

これらのビットは、D/A コンバータが無効になっていて、較正モードでない場合（DAC_CR レジスタのビット EN2 = 0 および CEN2 = 0）にのみ書込み可能です。EN2 = 1 または CEN2 = 1 の場合、書込み操作は無視されます。

これらは、D/A コンバータ･チャネル 2 モードを選択するために、ソフトウェアによってセットまたはクリアされます。

– DAC チャネル 2 は通常動作モードです。

000：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

001：D/A コンバータ･チャネル 2 はバッファ有効で外部ピンとオンチップ･ペリフェラルに接続されます。

010：DAC チャネル 2 はバッファ無効で外部ピンに接続されています。

011：D/A コンバータ･チャネル 2 はバッファ無効でオンチップ･ペリフェラルに接続されます。

– D/A コンバータ･チャネル 2はサンプル･アンド･ホールド･モードです。

100：DAC チャネル 2 はバッファ有効で外部ピンに接続されています。

101：D/A コンバータ･チャネル 2 はバッファ有効で外部ピンとオンチップ･ペリフェラルに接続されます。

110：D/A コンバータ･チャネル 2 はバッファ無効で外部ピンとオンチップ･ペリフェラルに接続されます。

111：D/A コンバータ･チャネル 2 はバッファ無効でオンチップ･ペリフェラルに接続されます。

注： このレジスタは、EN2 = 0 のときにのみ変更できます。

D/A コンバータ･チャネル 2 の有無については、セクション 22.3：D/A コンバータの実装を参照してください。

ビット 15:14 **HFSEL[1:0]**：高周波数インタフェースモードの選択

00：高周波数インタフェースモードは無効です。

01：AHB>80 MHz に適合する高周波数インタフェースモードは有効です。

10：AHB>160 MHz に適合する高周波数インタフェースモードは有効です。

11：予約済みです。

ビット 13:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **SINFORMAT1**：DAC チャネル 1 の符号付きフォーマットの有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：入力データは符号なしフォーマットです。

1：入力データは符号付きフォーマット（2 の補数）です。MSB ビットは、符号を表します。

ビット 8 **DMADOUBLE1**：DAC チャネル 1 DMA ダブルデータモード

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA 通常モードが選択されています。

1：DMA ダブルデータモードが選択されています。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **MODE1[2:0]**：DAC チャネル 1 モード

これらのビットは、D/A コンバータが無効になっていて、較正モードでない場合（DAC_CR レジスタのビット EN1 = 0 および CEN1 = 0）にのみ書込み可能です。EN1 = 1 または CEN1 = 1 の場合、書込み操作は無視されます。

これらは、D/A コンバータ･チャネル 1 モードを選択するために、ソフトウェアによってセットまたはクリアされます。

– DAC チャネル 1 は通常動作モードです。

000：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

001：D/A コンバータ･チャネル 1 はバッファ有効で外部ピンとオンチップ･ペリフェラルに接続されます。

010：DAC チャネル 1 はバッファ無効で外部ピンに接続されています。

011：D/A コンバータ･チャネル 1 はバッファ無効でオンチップ･ペリフェラルに接続されます。

– D/A コンバータ･チャネル 1 はサンプル･アンド･ホールド･モードです。

100：DAC チャネル 1 はバッファ有効で外部ピンに接続されています。

101：D/A コンバータ･チャネル 1 はバッファ有効で外部ピンとオンチップ･ペリフェラルに接続されます。

110：D/A コンバータ･チャネル 1 はバッファ無効で外部ピンとオンチップペリフェラルに接続されます。

111：D/A コンバータ･チャネル 1 はバッファ無効でオンチップ･ペリフェラルに接続されます。

注： このレジスタは、EN1 = 0 のときにのみ変更できます。

22.7.17 D/A コンバータ･チャネル 1 サンプル･アンド･ホールドのサンプル時間レジスタ（DAC_SHSR1）

アドレス･オフセット：0x40

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE1[9:0]**：D/A コンバータチャネル 1サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは D/A コンバータ･チャネル 1 が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SR レジスタの BWST1 がローのときのみ書込み可能であり、BWST1 = 1 の場合、書込み操作は無視されます。

注： **これは、サンプル･フェーズを実行する LSI/LSE クロック･サイクル数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x LSI/LSE クロック周期です。**

22.7.18 DAC チャネル 2 サンプルおよびホールドのサンプル時間レジスタ（DAC_SHSR2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x44

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TSAMPLE2[9:0]**：D/A コンバータチャネル 2サンプル時間（サンプルおよびホールドモードでのみ有効）

これらのビットは D/A コンバータ･チャネル 2 が無効である場合や通常動作中にも書込み可能です。後者の場合、DAC_SR レジスタの BWST2 がローのときのみ書込み可能であり、BWST2 = 1 の場合、書込み操作は無視されます。

注： **これは、サンプル･フェーズを実行する LSI/LSE クロック･サイクル数を表します。サンプリング時間 = (TSAMPLE1[9:0] + 1) x LSI/LSE クロック周期です。**

22.7.19 DAC サンプルおよびホールド時間レジスタ（DAC_SHHR）

アドレスオフセット：0x48

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:16 **THOLD2[9:0]**：D/A コンバータチャネル 2ホールド時間（サンプルおよびホールドモードでのみ有効）
ホールド時間 = (THOLD[9:0]) x LSI/LSE クロック周期

注： このレジスタは、EN2 = 0 のときにのみ変更できます。
これらのビットは、デュアル･チャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **THOLD1[9:0]**：D/A コンバータチャネル 1ホールド時間（サンプルおよびホールドモードでのみ有効）
ホールド時間 = (THOLD[9:0]) x LSI/LSE クロック周期

注： このレジスタは、EN1 = 0 のときにのみ変更できます。

注： これらのビットは、D/A コンバータ･チャネルが無効になっていて、通常動作モードの場合（DAC_CR レジスタのビット ENx = 0 およびビット CENx = 0）にのみ書込み可能です。ENx =1 または CENx =1 の場合、書込み操作は無視されます。

22.7.20 D/A コンバータ･サンプル･アンド･ホールドのリフレッシュ時間レジスタ（DAC_SHRR）

アドレスオフセット：0x4C

リセット値：0x0001 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **TREFRESH2[7:0]**：D/A コンバータチャネル 2リフレッシュ時間（サンプルおよびホールドモードでのみ有効）
リフレッシュ時間 = (TREFRESH[7:0]) x LSI/LSE クロック周期

注： このレジスタは、EN2 = 0 のときにのみ変更できます。
これらのビットは、デュアル･チャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TREFRESH1[7:0]**：D/A コンバータチャネル 1リフレッシュ時間（サンプルおよびホールドモードでのみ有効）

リフレッシュ時間 = (TREFRESH[7:0]) x LSI/LSE クロック周期

注： このレジスタは、EN1 = 0 のときにのみ変更できます。

注： これらのビットは、D/A コンバータ･チャネルが無効になっていて、通常動作モードの場合（DAC_CR レジスタのビット ENx = 0 およびビット CENx = 0）にのみ書込み可能です。ENx =1 または CENx =1 の場合、書込み操作は無視されます。

22.7.21 D/A コンバータ･チャネル 1 のこぎり波レジスタ（DAC_STR1）

アドレス･オフセット：0x58

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STINCDATA1 [15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	STDIR1	STRSTDATA1[11:0]											
			rw	rw											

ビット 31:16 **STINCDATA1[15:0]**：DAC チャネル 1 のこぎり波インクリメント値（12.4 ビット形式）

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **STDIR1**：DAC チャネル 1 のこぎり波方向設定

このビットは、のこぎり波ステップの方向を選択するために、ソフトウェアによって書き込まれます。

0：デクリメント

1：インクリメント

ビット 11:0 **STRSTDATA1[11:0]**：DAC チャネル 1 のこぎり波リセット値

22.7.22 DAC チャネル 2 のこぎり波レジスタ（DAC_STR2）

このレジスタは、デュアルチャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

アドレスオフセット：0x5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
STINCDATA2 [15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	STDIR2	STRSTDATA2[11:0]											
			rw	rw											

ビット 31:16 **STINCDATA2[15:0]**：DAC チャネル 2 のこぎり波インクリメント値（12.4 ビット形式）

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **STDIR2**：DAC チャネル 2 のこぎり波方向設定

このビットは、のこぎり波ステップの方向を選択するために、ソフトウェアによって書き込まれます。

0：デクリメント

1：インクリメント

ビット 11:0 **STRSTDATA2[11:0]**：DAC チャネル 2 のこぎり波リセット値

22.7.23 DAC のこぎり波モードレジスタ（DAC_STMODR）

アドレスオフセット：0x60

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	STINCTRIGSEL2 [3:0]				Res.	Res.	Res.	Res.	STRSTTRIGSEL2 [3:0]			
				rw								rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	STINCTRIGSEL1 [3:0]				Res.	Res.	Res.	Res.	STRSTTRIGSEL1 [3:0]			
				rw								rw			

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **STINCTRIGSEL2[3:0]**：DAC チャネル 2 のこぎり波インクリメントトリガ選択

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

0000：SWTRIGB2

0001：dac_inc_ch2_trg1

....

1111：dac_inc_ch2_trg15

注： これらのビットは、デュアル･チャネル D/A コンバータでのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 19:16 **STRSTTRIGSEL2[3:0]**：DAC チャネル 2 のこぎり波リセットトリガ選択

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

0000：SWTRIGB2

0001：dac_ch2_trg1

....

1111：dac_ch2_trg15

マッピングは TSEL2[3:0] と同じです。

注： これらのビットは、デュアルチャネル DAC でのみ使用できます。セクション 22.3：D/A コンバータの実装を参照してください。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **STINCTRIGSEL1[3:0]**：DAC チャネル 1 のこぎり波インクリメントトリガ選択

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

0000：SWTRIGB1

0001：dac_inc_ch1_trg1

....

1111：dac_inc_ch1_trg15

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **STRSTTRIGSEL1[3:0]**：DAC チャネル 1 のこぎり波リセットトリガ選択

トリガの設定とマッピングの詳細については、セクション 22.4.2：DAC ピンおよび内部信号のトリガ選択表を参照してください。

0000：SWTRIGB1

0001：dac_ch1_trg1

....

1111：dac_ch1_trg15

マッピングは TSEL1[3:0] と同じです。

22.7.24 D/A コンバータ・レジスタ・マップ

表 193 に DAC レジスタの要約を示します。

表 193. D/A コンバータ・レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	DAC_CR	Res.	CEN2	DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2 [2:0]		TSEL23	TSEL22	TSEL21	TSEL20	TEN2	EN2	Res.	CEN1	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE1 [1:0]		TSEL13	TSEL12	TSEL11	TSEL10	TEN1	EN1
	リセット値		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	DAC_SWTRGR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIGB2	SWTRIGB1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWTRIG2	SWTRIG1
	リセット値															0	0															0	0
0x08	DAC_DHR12R1	Res.	Res.	Res.	Res.	DACC1DHRB [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x0C	DAC_DHR12L1	DACC1DHRB[11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x10	DAC_DHR8R1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC1DHRB [7:0]								DACC1DHR [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	DAC_DHR12R2	Res.	Res.	Res.	Res.	DACC2DHRB [11:0]												Res.	Res.	Res.	Res.	DACC2DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x18	DAC_DHR12L2	DACC2DHRB[11:0]												Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	DAC_DHR8R2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHRB [7:0]								DACC2DHR [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	DAC_DHR12RD	Res.	Res.	Res.	Res.	DACC2DHR [11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x24	DAC_DHR12LD	DACC2DHR[11:0]												Res.	Res.	Res.	Res.	DACC1DHR [11:0]												Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0				
0x28	DAC_DHR8RD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DACC2DHR [7:0]								DACC1DHR [7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	DAC_DOR1	Res.	Res.	Res.	Res.	DACC1DORB [11:0]												Res.	Res.	Res.	Res.	DACC1DOR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x30	DAC_DOR2	Res.	Res.	Res.	Res.	DACC2DORB [11:0]												Res.	Res.	Res.	Res.	DACC2DOR [11:0]											
	リセット値					0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x34	DAC_SR	BWST2	CAL_FLAG2	DMAUDR2	DORSTAT2	DAC2RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BWST1	CAL_FLAG1	DMAUDR1	DORSTAT1	DAC1RDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0												0	0	0	0	0											

表 193. D/A コンバータ・レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x38	**DAC_CCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM2 [4:0]					Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OTRIM1 [4:0]				
	リセット値												X	X	X	X	X												X	X	X	X	X
0x3C	**DAC_MCR**	Res.	Res.	Res.	Res.	Res.	Res.	SINFORMAT2	DMADOUBLE2	Res.	Res.	Res.	Res.	Res.	MODE2 [2:0]			HFSEL1	HFSEL0	Res.	Res.	Res.	Res.	SINFORMAT1	DMADOUBLE1	Res.	Res.	Res.	Res.	Res.	MODE1 [2:0]		
	リセット値							0	0						0	0	0	0	0					0	0						0	0	0
0x40	**DAC_SHSR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE1 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x44	**DAC_SHSR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSAMPLE2 [9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x48	**DAC_SHHR**	Res.	Res.	Res.	Res.	Res.	Res.	THOLD2 [9:0]										Res.	Res.	Res.	Res.	Res.	Res.	THOLD1 [9:0]									
	リセット値							0	0	0	0	0	0	0	0	0	1							0	0	0	0	0	0	0	0	0	1
0x4C	**DAC_SHRR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH2 [7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TREFRESH1 [7:0]							
	リセット値									0	0	0	0	0	0	0	1									0	0	0	0	0	0	0	1
0x58	DAC_STR1	STINCDATA1[15:0]																Res.	Res.	Res.	Res.	STRSTDATA1 [11:0]											
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x5C	DAC_STR2	STINCDATA2[15:0]																Res.	Res.	Res.	Res.	STRSTDATA2 [11:0]											
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x60	DAC_STMODR	Res.	Res.	Res.	Res.	STINCTRIGSEL2 [3:0]				Res.	Res.	Res.	Res.	STRSTTRIGSEL2 [3:0]				Res.	Res.	Res.	Res.	STINCTRIGSEL1 [3:0]		Res.	Res.	Res.	Res.	STRSTTRIGSEL1 [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

23 電圧基準バッファ（VREFBUF）

23.1 概要

デバイスには電圧基準バッファが搭載されており、A/D コンバータや D/Aコンバータの電圧基準や、VREF+ ピンを介して外部コンポーネント用の電圧基準としても使用できます。VREF+ ピンがパッケージで VDDA ピンとダブルボンディングされている場合、電圧基準バッファは使用できず、無効にしておく必要があります（パッケージのピン配置の説明については、データシートを参照してください）。

23.2 VREFBUF の機能説明

内部電圧基準バッファは次の 3 つの電圧[a] をサポートしており、VREFBUF_CSR レジスタの VRS ビットで設定します。

- VRS = 00：約 2.048 V.
- VRS = 01：約 2.5 V.
- VRS = 10：約 2.90 V.

内部電圧基準は ENVR および HIZ ビットの設定に応じて 4 つの個別モードに設定できます。これらのモードを以下の表に示します。

表 194. VREF バッファモード

ENVR	ハイ･インピーダンス	VREF バッファ設定
0	0	VREFBUF バッファ･オフ･モード： – V_{REF+} ピンを V_{SSA} にプルダウン
0	1	外部電圧基準モード（デフォルト値）： – VREFBUF バッファ･オフ – V_{REF+} ピンは入力モード
1	0	内部電圧基準モード – VREFBUF バッファ･オン – V_{REF+} ピンを VREFBUF バッファ出力に接続
1	1	ホールドモード： – 出力バッファなしで VREF を有効化します。VREF+ ピンの電圧を外付けのコンデンサで保持します。 – VRR の検出は無効で、VRR ビットは最後の状態を保持します。

VREFBUF_CSR レジスタの ENVR ビットをセットし、HIZ ビットをクリアすることで VREFBUF を有効化した後、ユーザは電圧基準出力が期待値に達したことを示す VRR ビットがセットされるまで待つ必要があります。

a. V_{DDA} の最小電圧は VRS の設定に依存します。製品データシートを参照してください。

23.3 VREFBUF のトリミング

VREFBUF の出力電圧は、工場出荷時に較正されています。リセットするたび、および VRS 設定を変更するたびに較正データが自動的に TRIM レジスタに読み込まれます。

必要に応じ、TRIM レジスタのビットを直接変更して出力電圧をトリミングできます。その場合、VRS の設定にレジスタの変更を反映するにはデバイスをリセットする必要があります。

23.4 VREFBUF レジスタ

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

アドレスオフセット：0x00

リセット値：0x0000 0002

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **VRS[1:0]**：電圧基準スケール

これらのビットは、電圧基準バッファによって生成される値を選択します。

00：電圧基準が 2.048 V にセットされます。

01：電圧基準が 2.5 V にセットされます。

10：電圧基準が 2.90 V にセットされます。

11：予約済みです。

注： VREFBUF が無効（ENVR = 0）の場合にのみ、このビットフィールドをソフトウェアでプログラムできます。

ビット 3 **VRR**：電圧基準バッファ･レディ

0：電圧基準バッファ出力がレディ状態ではありません。

1：電圧基準バッファ出力がリクエストされたレベルに達しました。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **HIZ**：ハイ･インピーダンス･モード

このビットによって、V_{REF+} ピンを接続／切断するためのアナログスイッチを制御します。

0：V_{REF+} ピンが 電圧基準バッファ出力に内部接続されます。

1：V_{REF+} ピンをハイインピーダンスにします。

ENVR ビットの設定に応じたモード説明については、表 194:VREF バッファモードを参照してください。

ビット 0 **ENVR**：電圧基準バッファ･モード有効化

このビットは、電圧基準バッファモードの有効化に使用されます。

0：内部電圧基準モード無効化（外部電圧基準モード）

1：内部電圧基準モード（基準バッファ有効化またはホールドモード）有効化

23.4.2 VREFBUF 較正制御レジスタ（VREFBUF_CCR）

アドレスオフセット：0x04

リセット値：0x0000 00XX

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:0 **TRIM[5:0]**：トリミング･コード

TRIM コードは、次に示す機能に従って設定および更新される 6 ビットの符号なしデータです（最小値 000000、最大値 111111）。

リセット：

TRIM[5:0] は、生産試験中に Flash メモリに格納されたトリミング値 VRS = 0で自動的に初期化されます。

VRS の変化：

TRIM[5:0] は、生産試験中に Flash メモリに格納された（VRS 設定に対応する）トリミング値で自動的に初期化されます。

TRIM[5:0] への書込み：

ユーザは TRIM[5:0] を任意の値で変更できます。これにより、VRS によるトリミング値の制御は恒久的に無効化されます（デバイスがリセットされるまで）。

注： ユーザ･アプリケーションでトリミングを実行する場合、トリミング･コードを 000000 から 111111 まで昇順で設定する必要があります。

23.4.3 VREFBUF レジスタ･マップ

次の表に、VREFBUF レジスタマップとリセット値を示します。

表 195. VREFBUF レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	VREFBUF_CSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	VRR	VRS[1:0]		ハイ･インピーダンス	ENVR
	リセット値																												0	0	0	1	0
0x04	VREFBUF_CCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRIM[5:0]					
	リセット値																											x	x	x	x	x	x

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

24 コンパレータ（COMP）

24.1 COMP の概要

デバイスは最大 7 つの超高速アナログコンパレータを内蔵します。

コンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低電力モードからのウェイクアップ
- アナログ信号調節
- タイマからの PWM 出力を組み合わせた場合のサイクルごとの電流制御ループ。

24.2 COMP の主な機能

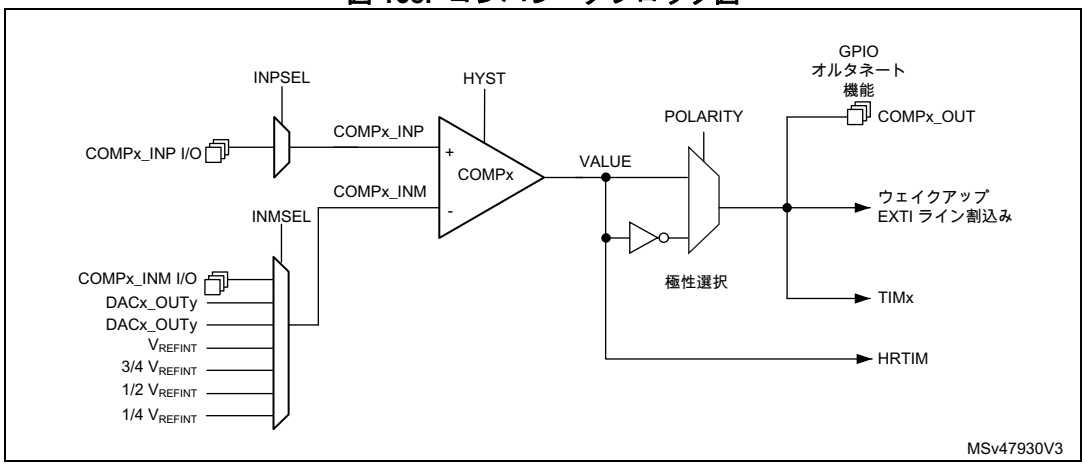
- 各コンパレータは、電圧を柔軟に選択できるように、次のような設定可能な正入力と負入力を備えています。
  - マルチプレクス I/O ピン
  - DAC チャネル
  - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および 3 つの約数（1/4、1/2、3/4）
- プログラム可能なヒステリシス
- ブレークイベントをトリガして高速 PWM シャットダウンするための I/O またはタイマ入力への出力リダイレクション
- 出力ブランキングによるスイッチノイズへの耐性
- SLEEP および STOP モードからのウェイクアップによるチャネルごとの割込み生成

24.3 COMP の機能詳細

24.3.1 COMP ブロック図

1 つのコンパレータ･チャネル･フロントエンドのブロック図を図 168 : コンパレータブロック図に示します。

図 168. コンパレータブロック図



24.3.2 COMP ピンおよび内部信号

コンパレータ入力として使用される I/O は、GPIO レジスタのアナログモードで設定する必要があります。

コンパレータの出力は、データシートの「オルタネート機能配置」表に示されているオルタネート機能チャネルを使用して I/O に接続することができます。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- BKIN および BKIN2 入力を使用した PWM 信号の緊急停止
- OCREF_CLR 入力を使用したサイクル-バイ-サイクル電流制御
- タイミング測定のためのキャプチャ入力

コンパレータの出力先を内部および外部から一斉に変更することが可能です。

表 196. COMPx 非反転入力割付け

INPSEL	COMP1_INP	COMP2_INP	COMP3_INP	COMP4_INP	COMP5_INP	COMP6_INP	COMP7_INP
0	PA1	PA7	PA0	PB0	PB13	PB11	PB14
1	PB1	PA3	PC1	PE7	PD12	PD11	PD14

表 197. COMPx 反転入力割付け

INMSEL [2:0]	COMP1_INM	COMP2_INM	COMP3_INM	COMP4_INM	COMP5_INM	COMP6_INM	COMP7_INM
000	1/4 V_{REFINT}						
001	1/2 V_{REFINT}						
010	3/4 V_{REFINT}						
011	V_{REFINT}						
100	DAC3_CH1	DAC3_CH2	DAC3_CH1	DAC3_CH2	DAC4_CH1	DAC4_CH2	DAC4_CH1
101	DAC1_CH1	DAC1_CH2	DAC1_CH1	DAC1_CH1	DAC1_CH2	DAC2_CH1	DAC2_CH1
110	PA4	PA5	PF1	PE8	PB10	PD10	PD15
111	PA0	PA2	PC0	PB2	PD13	PB15	PB12

24.3.3 COMP のリセットおよびクロック

クロックコントローラによって提供される COMP クロックは、APB2 クロックと同期しています。

RCC コントローラには、COMP 専用クロックイネーブル制御ビットはありません。リセットおよびクロックイネーブルビットは COMP と SYSCFG に共通です。

注： **重要：極性選択ロジックおよびポートへの出力先変更は、APB クロックとは独立して機能します。これにより、コンパレータは STOP モードでも機能することができます。**

24.3.4 COMP ロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるようなアプリケーションの場合、万一誤ったレジスタへのアクセスやプログラムカウンタの破壊が起こった場合に、コンパレータのプログラミングが絶対に変更されないようにすることが必要です。

このような目的で、コンパレータの制御レジスタおよびステータスレジスタを書込み保護することができます（読出し専用）。

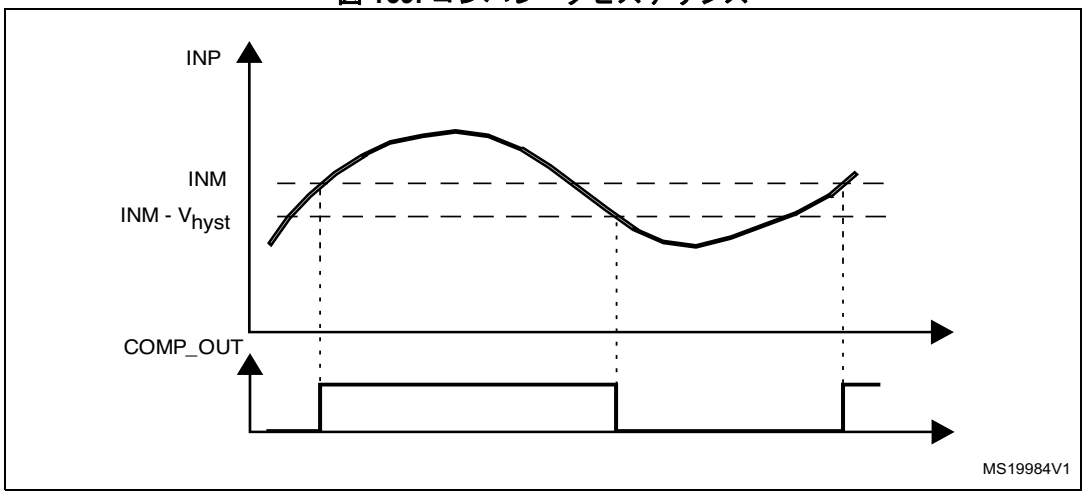
プログラミングが完了したら、COMPx の LOCK ビットをセットできます。これにより、COMPx の LOCK ビットを含め、レジスタ全体が読出し専用になります。

書込み保護は MCU リセットによってのみ解除可能です。

24.3.5 COMP ヒステリシス

コンパレータには、ノイズの多い入力信号での疑似出力遷移を避けるために、プログラム可能なヒステリシスが含まれています。非対称であり、コンパレータ出力の立ち下がりエッジに対してのみ作用します。内部ヒステリシス機能を無効にして、外部コンポーネントによるヒステリシスの量をセットすることができ、これは、たとえば、低電力モードを終了するときに役立ちます。

図 169. コンパレータヒステリシス



24.3.6 COMP 出力のブランキング

ブランキング機能の目的は、PWM 周期の開始時に電流規制が短絡電流スパイクでトリップするのを防ぐことです（通常は電源スイッチのアンチパラレルダイオードのリカバリ電流）。これは、タイマ出力比較信号で定義されたデッドウィンドウの設定を通過します。ブランキングソースは、表 198 : ブランキング ソースに示すように、対応する COMP_CxCSR レジスタの BLANKSEL[2:0] ビットフィールドにより、コンパレータチャネルごとにソフトウェアによって選択されます。反転されたブランキング信号はコンパレータステージ出力と論理 AND されて、コンパレータチャネル x 出力を生成します。次の図に示す例を参照してください。

図 170. コンパレータの出力のブランキング

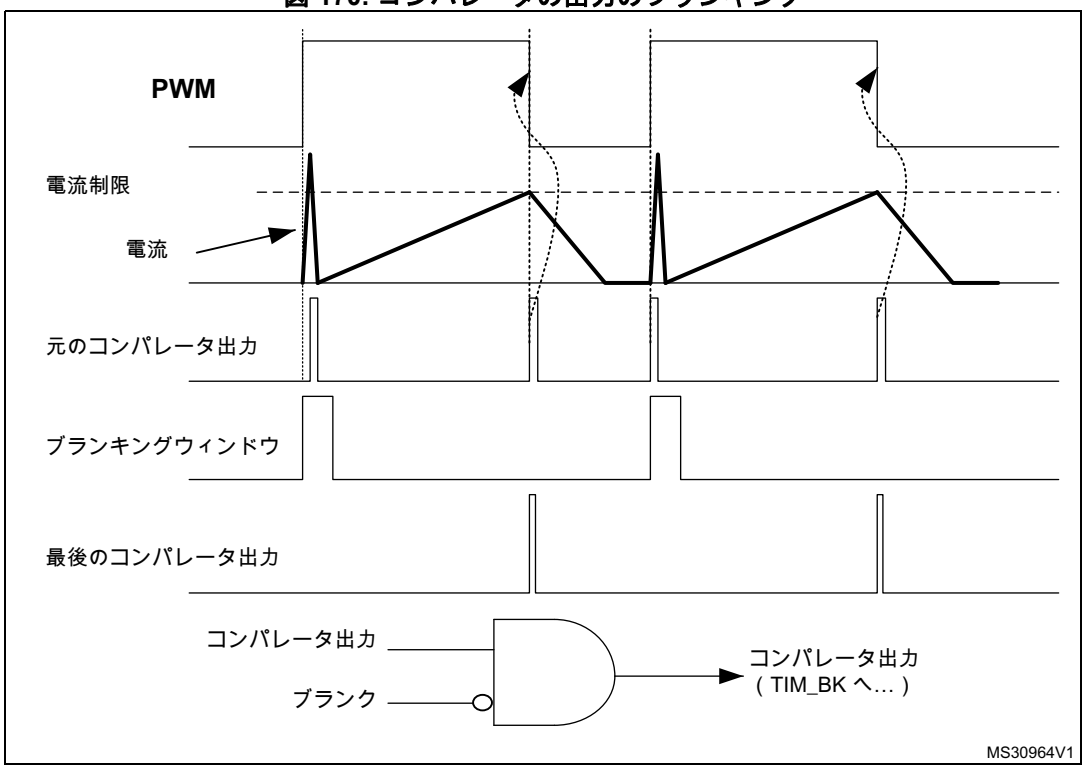


表 198. ブランキング ソース

BLANKSEL [2:0]	COMP1	COMP2	COMP3	COMP4	COMP5	COMP6	COMP7
001	TIM1_OC5	TIM1_OC5	TIM1_OC5	TIM3_OC4	TIM2_OC3	TIM8_OC5	TIM1_OC5
010	TIM2_OC3	TIM2_OC3	TIM3_OC3	TIM8_OC5	TIM8_OC5	TIM2_OC4	TIM8_OC5
011	TIM3_OC3	TIM3_OC3	TIM2_OC4	TIM15_OC1	TIM3_OC3	TIM15_OC2	TIM3_OC3
100	TIM8_OC5	TIM8_OC5	TIM8_OC5	TIM1_OC5	TIM1_OC5	TIM1_OC5	TIM15_OC2
101	TIM20_OC5	TIM20_OC5	TIM20_OC5	TIM20_OC5	TIM20_OC5	TIM20_OC5	TIM20_OC5
110	TIM15_OC1	TIM15_OC1	TIM15_OC1	TIM15_OC1	TIM15_OC1	TIM15_OC1	TIM15_OC1
111	TIM4_OC3	TIM4_OC3	TIM4_OC3	TIM4_OC3	TIM4_OC3	TIM4_OC3	TIM4_OC3

24.4 COMP 低電力モード

表 199. 低電力モードでのコンパレータの動作

モード	説明
SLEEP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスは SLEEP モードから復帰します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。COMP 割込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP	コンパレータへの影響はありません。 コンパレータ割込みによって、デバイスは STOP モードから復帰します。
STANDBY、SHUTDOWN	COMP レジスタはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再度初期化する必要があります。

24.5 COMP 割込み

コンパレータの出力は拡張割込み／イベントコントローラに内部的に接続されます。各コンパレータには専用の EXTI ラインがあり、割込みまたはイベントを生成することができます。SLEEP および STOP 低電力モードを終了するときにも同じ方法が使用されます。

詳細については、割込みおよびイベントのセクションを参照してください。

COMPx 割込みを有効にするには、次のシーケンスが必要です。

1. 割込みモードで COMPx 出力イベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジの感度を選択します。
2. 対応する EXTI ラインにマップされた NVIC IRQ チャネルを設定し、有効にします。
3. COMPx を有効にします。

COMP_CxCSR フラグによって割込みイベントにフラグを立てます。

24.6 COMP レジスタ

24.6.1 コンパレータ x 制御/ステータスレジスタ（COMP_CxCSR）

x = 1 ~ 7 について、COMP_CxCSR レジスタはコンパレータ x に関するすべてのビットとフラグを含みます。

アドレスオフセット：4(x-1)、ここで x = 1～7

システムリセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]		
rw	r							rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
rw							rw		rw	rw	rw				rw

ビット 31 **LOCK**：COMP_CxCSR レジスタ·ロック

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。コンパレータ x 制御レジスタ COMP_CxCSR[31:0] の内容全体をロックします。ロックされると、すべての制御ビットとフラグは読出し専用になり、書込みできなくなります。アンロックされると、制御ビットはソフトウェアによる書込みが可能です。

0：アンロック

1：ロック状態

ビット 30 **VALUE**：コンパレータ x 出力ステータス

この読出し専用フラグは、コンパレータ x の出力レベルを反映しています（図 168 に示されているように、極性セレクタおよびブランキング前）。

ビット 29:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **SCALEN**：V_{REFINT} スケーラ·イネーブル

このビットはソフトウェアによって制御され、すべてのコンパレータの反転入力時の V_{REFINT} スケーラの動作を有効にします。V_{REFINT} スケーラを無効にするには、すべての COMP_CxCSR レジスタの SCALEN ビットをディスエーブル状態にセットする必要があります。V_{REFINT} スケーラが無効のときには、マルチプレクサの 1/4 V_{REFINT}、1/2 V_{REFINT}、3/4 V_{REFINT}、および V_{REFINT} 入力を選択しないでください。

0：無効化

1：イネーブル

ビット 22 **BRGEN**：V_{REFINT} スケーラの抵抗ブリッジ有効化

このビットはソフトウェアによって制御され、V_{REFINT} スケーラのレジスタブリッジの動作を有効にします。レジスタブリッジを無効にするには、すべての COMP_CxCSR レジスタの BRGEN ビットをディスエーブル状態にセットする必要があります。レジスタブリッジが無効のときには、入力セレクタの 1/4 V_{REFINT}、1/2 V_{REFINT}、および3/4 V_{REFINT} 入力は、V_{REFINT} 電圧を受信します。

0：無効化

1：イネーブル

ビット 21:19 **BLANKSEL[2:0]**：コンパレータ x ブランキング信号選択

このビットフィールドはソフトウェアによって制御され、表 198：ブランキング ソースに示されているように、コンパレータチャネル x のブランキング信号を選択します。

ビット 18:16 **HYST[2:0]**：コンパレータ x ヒステリシス

このビットフィールドはソフトウェアによって制御され、コンパレータ x のヒステリシスを選択します。

000：ヒステリシスなし

001：10 mV ヒステリシス

010：20mv ヒステリシス

011：30mV ヒステリシス

100：40mV ヒステリシス

101：50mV ヒステリシス

110：60mV ヒステリシス

111：70mV ヒステリシス

ビット 15 **POL**：コンパレータ x 極性

このビットフィールドはソフトウェアによって制御され、コンパレータ x の出力極性を選択します。

0：反転されません。

1：反転されます。

ビット 14:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **INPSEL**：非反転入力のためのコンパレータ x の信号選択

このビットフィールドはソフトウェアによって制御され、表 196：COMPx 非反転入力割付けに示されているように、コンパレータ x の非反転入力 COMPx_INP の信号を選択します。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **INMSEL[2:0]**：反転入力のためのコンパレータ x の信号選択

このビットフィールドはソフトウェアによって制御され、表 197：COMPx 反転入力割付けに示されているように、コンパレータ x の反転入力 COMPx_INM の信号を選択します。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EN**：コンパレータ x イネーブル

このビットはソフトウェアによって制御され、コンパレータ x の動作を有効にします。

0：無効化

1：イネーブル

24.6.2 COMP レジスタマップ

次の表に コンパレータ レジスタの一覧を示します。

コンパレータ レジスタは、SYSCFG ペリフェラルレジスタのベースアドレスを共用しています。

表 200. COMP レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**COMP_C1CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x04	**COMP_C2CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x08	**COMP_C3CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x0C	**COMP_C4CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x10	**COMP_C5CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x14	**COMP_C6CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0
0x18	**COMP_C7CSR**	LOCK	VALUE	Res.	Res.	Res.	Res.	Res.	Res.	SCALEN	BRGEN	BLANKSEL[2:0]			HYST[2:0]			POL	Res.	Res.	Res.	Res.	Res.	Res.	INPSEL.	Res.	INMSEL[2:0]			Res.	Res.	Res.	EN
	リセット値	0	0							0	0	0	0	0	0	0	0	0							0		0	0	0				0

レジスタ境界アドレスについては、セクション 2.2.2 : メモリマップとレジスタ境界アドレスを参照してください。

25 オペアンプ（OPAMP）

25.1 概要

このデバイスは、6 個のオペアンプを内蔵しており、それぞれ入力 2 本と出力 1 本を持っています。3 つの I/O は外部ピンにつなげることができ、あらゆる種類の外部相互接続が可能です。オペアンプは、フォロアとして、または 2～64 の範囲の非反転ゲインもしくは -1～-63 の範囲の反転ゲインを持つアンプとして、内部的に設定できます。

正入力は、内部 DAC に接続できます。

出力は、内部 ADC に接続できます。

25.2 OPAMP の主な機能

- レールツーレール入出力電圧範囲
- 低入力バイアス電流
- 低入力オフセット電圧
- 高周波数 ゲイン帯域幅
- スルーレートが改善されるハイスピードモード

注： **OPAMP 特性の詳細については、製品データシートを参照してください。**

25.3 OPAMP の機能詳細

OPAMP にはいくつかのモードがあります。

各 OPAMP は個別に有効化でき、無効になっているときの出力はハイインピーダンスです。

有効時は、較正モードにしてから OPAMP のすべての入出力を切断したり、機能モードにしたりできます。

機能モードでは、OPAMP の入出力はセクション 25.3.4：信号接続に示すように接続されます。

25.3.1 内部 ADC チャネルへの OPAMP 出力リダイレクション

オペアンプ出力は、OPAMPx_CSR レジスタの OPAINTOEN ビットをセットすることにより、ADC チャネルに内部でリダイレクトできます。この場合、指定された OPAMP x _VOUT 出力がマップされる GPIO は未使用になり、別の目的に使用できます。ADC は OPAMPx_VOUT 電圧を内部で測定できます。OPAMPx 内部出力と ADC チャネル間の割り当てについては、図 201：オペアンプの可能な接続を参照してください。

25.3.2 OPAMP のリセットおよびクロック

クロックコントローラによって提供される OPAMP クロックは、PCLK2（APB2 クロック）と同期しています。RCC コントローラには、クロックイネーブル制御ビットは提供されていません。OPAMP にクロックソースを使用するには、RCC コントローラで SYSCFG クロックイネーブル制御ビットをセットする必要があります。

OPAEN ビットは、OPAMP の動作を有効化または無効化します。OPAMP レジスタ設定は、出力に誤った影響を及ぼさないように OPAEN ビットを有効化する前に変更する必要があります。

オペアンプの出力が必要なくなったときは、オペアンプを無効にして電力を低減することができます。以前にセットしたすべての設定（較正を含む）は、OPAMP が無効化されても保持されます。

25.3.3 初期設定

オペアンプのデフォルト設定は、3 つの入出力が外部ピンに接続されている機能モードです。デフォルトのモードでは、オペアンプはオフセット較正に出荷時のトリミング値を使用します。トリミング値は調整できます。トリミング値の変更については、セクション 25.3.7 : 較正を参照してください。デフォルト設定が通常モードを使用する場合、標準のパフォーマンスを提供します。オペアンプをハイスピードモードに切り替えてスルーレートを改善するように、OPAHSM ビットでセットできます。通常モードとハイスピードモードの両方の特性は、データシートで定義されています。

OPAMPx_CSR レジスタの OPAEN ビットがセットされるとすぐに、オペアンプは機能します。2 個の入力ピンおよび出力ピンは、セクション 25.3.4 : 信号接続に定義されているように接続されており、デフォルト接続設定は変更できます。

注 : **入力ピンおよび出力ピンは、対応する GPIOx_MODER レジスタでアナログモード（デフォルト状態）で設定する必要があります。**

25.3.4 信号接続

オペアンプピンのルーティングは、OPAMPx_CSR および OPAMP x _TCMR レジスタで決まります。

6 個のオペアンプ（OPAMPx（x = 1 ~ 6））の接続を以下の表に示します。

表 201. オペアンプの可能な接続

信号	ピン	内部	コメント
OPAMP1_VINM	PA3 (VINM0) PC5 (VINM1)	OPAMP1_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP1_VINP	PA1 (VINP0) PA3 (VINP1) PA7 (VINP2)	DAC3_CH1	VP_SEL ビットによって制御されます。
OPAMP1_VOUT	PA2	ADC1_IN3 ADC1_IN13(1)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC 入力は、ADC によって制御されます。
OPAMP2_VINM	PA5 (VINM0) PC5 (VINM1)	OPAMP2_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP2_VINP	PA7 (VINP0) PB14 (VINP1) PB0 (VINP2) PD14 (VINP3)	-	ビット VP_SEL で制御します。

表 201. オペアンプの可能な接続 （続き）

信号	ピン	内部	コメント
OPAMP2_VOUT	PA6	ADC2_IN3 ADC2_IN16 (1)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC 入力は、ADC によって制御されます。
OPAMP3_VINM	PB2 (VINM0) PB10 (VINM1)	OPAMP3_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP3_VINP	PB0 (VINP0) PB13 (VINP1) PA1 (VINP2)	DAC3_CH2(2)(3)	VP_SEL ビットによって制御されます。
OPAMP3_VOUT	PB1	ADC3_IN1 (2) (3) /ADC1_IN12 ADC2_IN18 (1) / ADC3_IN13 (1) (2) (3)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC 入力は、ADC によって制御されます。
OPAMP4_VINM	PB10 (VINM0) PD8 (VINM1)	OPAMP4_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP4_VINP	PB13 (VINP0) PD11 (VINP1) PB11 (VINP2)	DAC4_CH1	VP_SEL ビットによって制御されます。
OPAMP4_VOUT	PB12	ADC4_IN3/ADC1_IN11 ADC5_IN5 (1)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC入力は、ADC で制御されます。
OPAMP5_VINM	PB15 (VINM0) PA3 (VINM1)	OPAMP5_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP5_VINP	PB14 (VINP0) PD12 (VINP1) PC3 (VINP2)	DAC4_CH2	VP_SEL ビットによって制御されます。
OPAMP5_VOUT(4)	PA8	ADC5_IN1 ADC5_IN3 (1)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC入力は、ADC で制御されます。
OPAMP6_VINM	PA1 (VINM0) PB1 (VINM1)	OPAMP6_VOUT または PGA	PGA_GAIN ビットおよび VM_SEL ビットによって制御されます。
OPAMP6_VINP(5)	PB12 (VINP0) PD9 (VINP1) PB13 (VINP2)	DAC3_CH1	VP_SEL ビットによって制御されます。
OPAMP6_VOUT	PB11	ADC12_IN14 ADC4_IN17 (1) (2) ADC3_IN17 (1) (3)	OPAMP が有効であり、OPAMP 内部出力が無効なとき、このピンが接続されます。ADC入力は、ADC で制御されます。

1. この ADC チャネルは、OPAINTOEN ビットをセットすると内部で OPAMPx_VOUT に接続されます。この場合、OPAMP x _VOUT として使用できる I/O を別の目的に使用できます。
2. カテゴリ 3 デバイスの場合のみ
3. カテゴリ 4 デバイスの場合のみ
4. OPAMP4/5 はカテゴリ 3 デバイスでのみサポートされます。
5. OPAMP6 はカテゴリ 3 およびカテゴリ 4 デバイスでのみサポートされます。

25.3.5 OPAMP モード

すべてのオペアンプの入出力に、端子からアクセスできます。アンプは複数の設定環境で使用できます。

- スタンドアロンモード（外部ゲイン設定モード）
- フォロア設定モード
- PGA モード

注： **信号のインピーダンスを、（ソースの抵抗性電圧降下が原因で）重大な乱れをつくる入力リークが発生しないレベル以下に維持する必要があります。詳細については、データシートの電気的特性のセクションを参照してください。**

スタンドアロンモード（外部ゲイン設定モード）

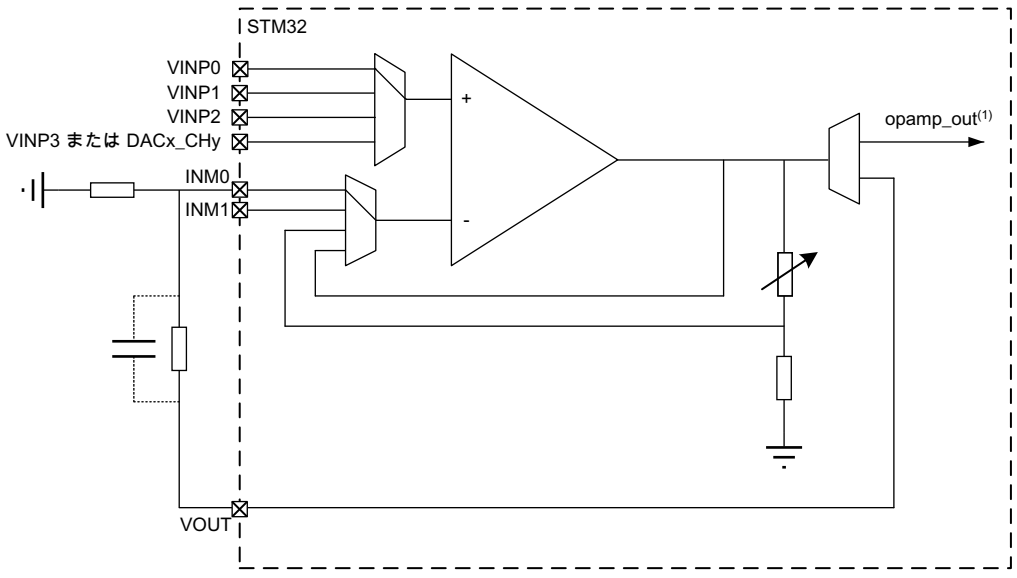
スタンドアロンモードで OPAMP を使用する手順をここで説明します。

OPAMPx_CSR のデフォルト値と GPIOx_MODER のデフォルト状態から始めて、OPAEN ビットがセットされるとすぐに、入力ピン2つと出力ピン1つがオペアンプに接続されます。

デフォルト設定が出荷時トリミング値を使用して、通常モード（最高のパフォーマンス）で動作します。OPAMP の動作は、次のように変更できます。

- 高スルーレートとするために OPAHSM を「オペアンプハイスピード」モードにセットできます。
- USERTRIM をセットして、入力オフセットのためのトリミング値を変更できます。

図 171. スタンドアロンモード：外部ゲイン設定モード



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で ADC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

MSv48039V1

フォロア設定モード

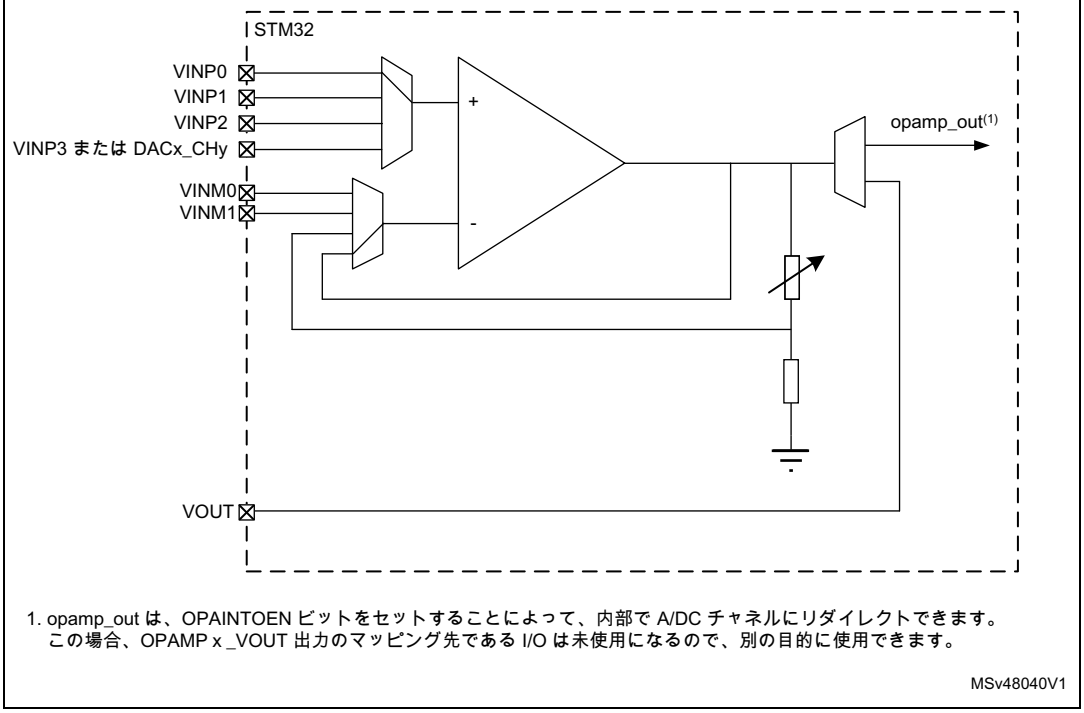
フォロアモードで OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「OPAMPx_VINM 入力に接続された opamp_out」である 11 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。
- OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が OPAMPx_VOUT ピンにバッファリングされます。

注： **OPAMPx_VINM に対応するピンは、他の用途のために解放されます。**

OPAMP 出力の信号も ADC 入力とみなされます。結果として、フォロアモードに設定された OPAMP を使用して、入力信号周波数がオペアンプのゲイン周波数の仕様と互換性があると想定して、入力信号を ADC 入力に供給する前にインピーダンス適合を実行できます。

図 172. フォロア設定



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で A/DC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

MSv48040V1

プログラマブルゲインアンプモード

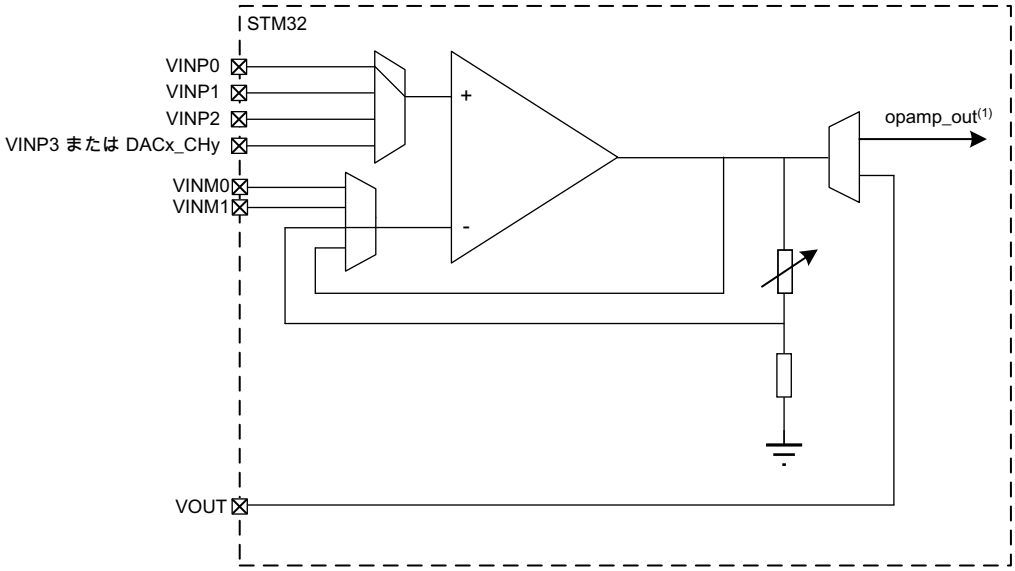
プログラム可能なゲインアンプとして OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「内部ゲイン 2、4、8、16、32、または 64」である 00000～00101 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」である 00 に設定します。

  OPAEN ビットがセットされるとすぐに、OPAMPx_VINP ピンの電圧が選択されたゲインで増幅し、OPAMPx_VOUT ピンで確認できます。

注： **飽和を避けるために、入力電圧は選択されたゲインで分圧された V_{DDA} 未満にしておく必要があります。**

図 173. PGA モード、内部ゲイン設定（x2/x4/x8/x16/x32/x64）、反転入力は使用されません。



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で A/DC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

MSv48041V1

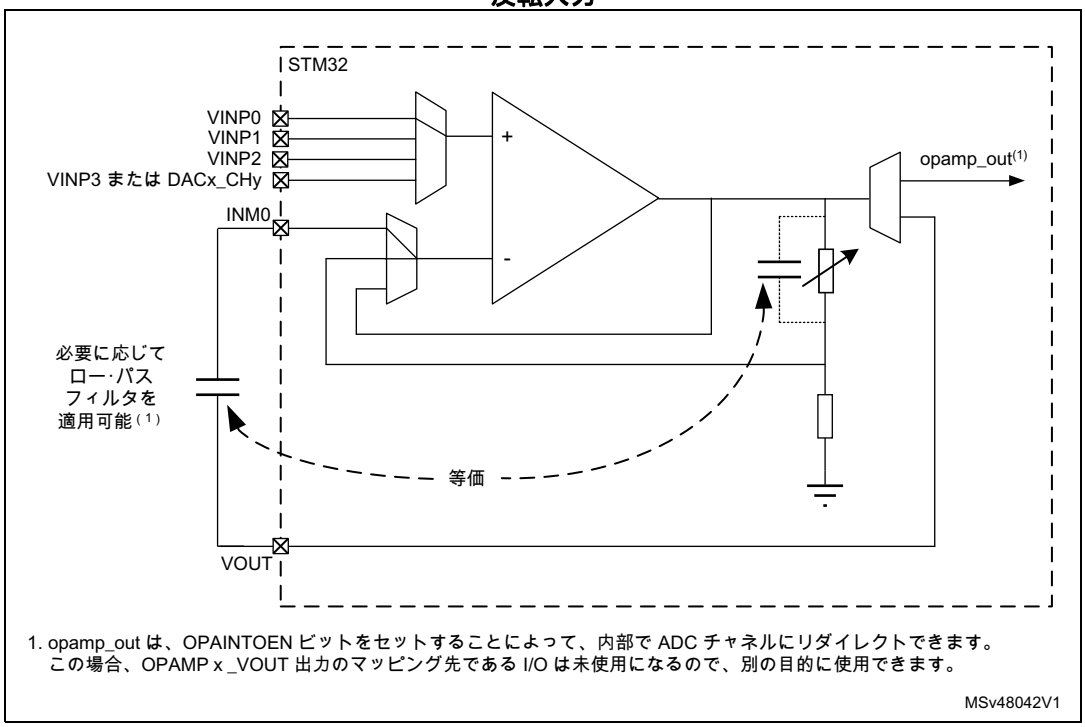
外部フィルタリングでのプログラム可能なゲインアンプモード

入力信号を増幅するために外部フィルタリングで OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「VINM0 のフィルタリング付き内部ゲイン 2、4、8、16、32、または 64」である 10000～10101 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

  VINM であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と VINM の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 174. PGA モード、内部ゲイン設定（x2/x4/x8/x16/x32/x64）、フィルタリング用の反転入力



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で ADC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

MSv48042V1

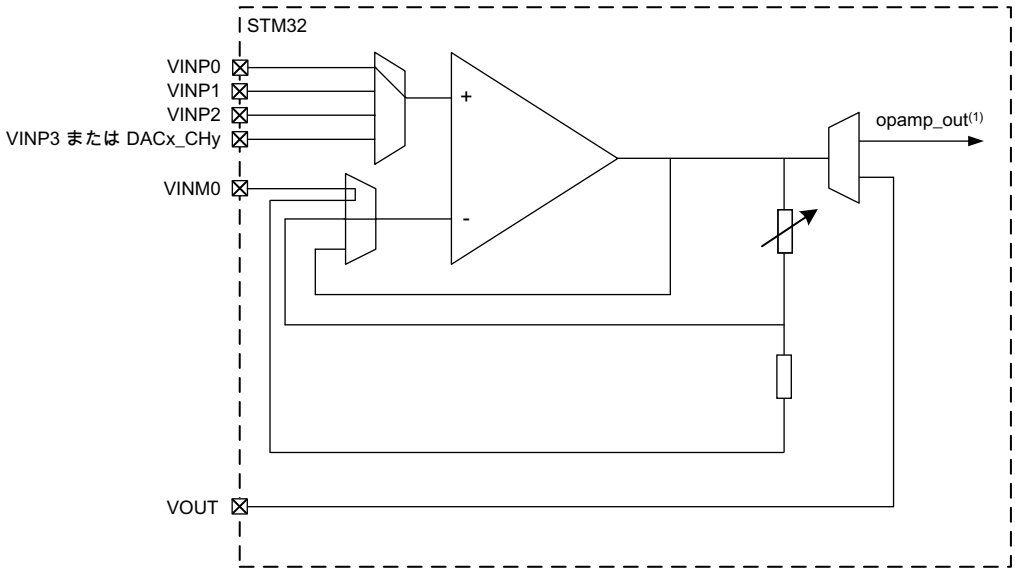
1. ゲインはカットオフ周波数によって決まります。

プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたは反転モード

非反転モードまたは反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「VINM0の反転ゲイン = -1、-3、-7、-15、-31、-63/非反転ゲイン = 2、4、8、16、32、64」である 01000～01101 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

図 175. PGA モード、非反転ゲイン設定（x2/x4/x8/x16/x32/x64）または反転ゲイン設定（x-1/x-3/x-7/x-15/x-31/x-63）



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で A/DC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

MSv48043V1

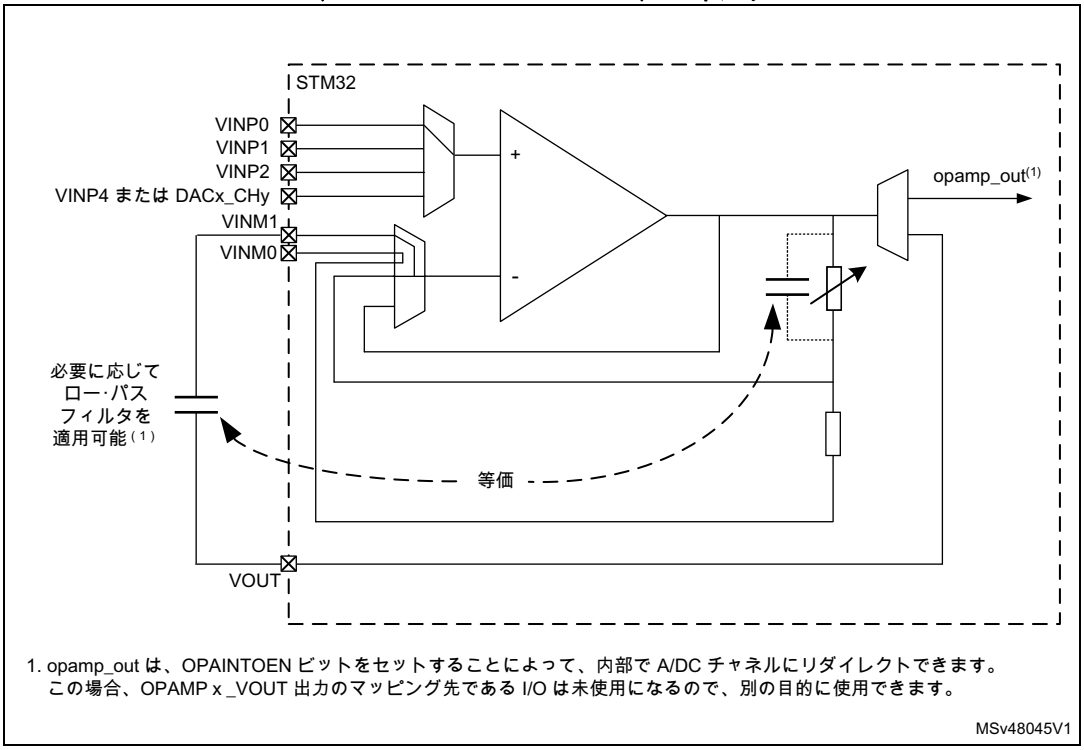
プログラム可能なゲインアンプ、外部バイアス付き非反転モードまたはフィルタリング付き反転モード

非反転モードまたはフィルタリング付き反転モードのバイアス電圧を含んでいる入力信号の振幅を増幅するために、OPAMP を使用する手順をここで説明します。

- VM_SEL ビットを「フィードバック抵抗はOPAMPx_VINM 入力に接続」である 10 に設定します。
- PGA_GAIN ビットを「フィルタリング用 VINM0、VINM1 ピンの反転ゲイン = -1、-3、-7、-15、-31、-63/非反転ゲイン = 2、4、8、16、32、64」である 11000～11101 に設定します。
- VP_SEL ビットを「OPAMPx_VINP に接続された GPIO」に設定します。

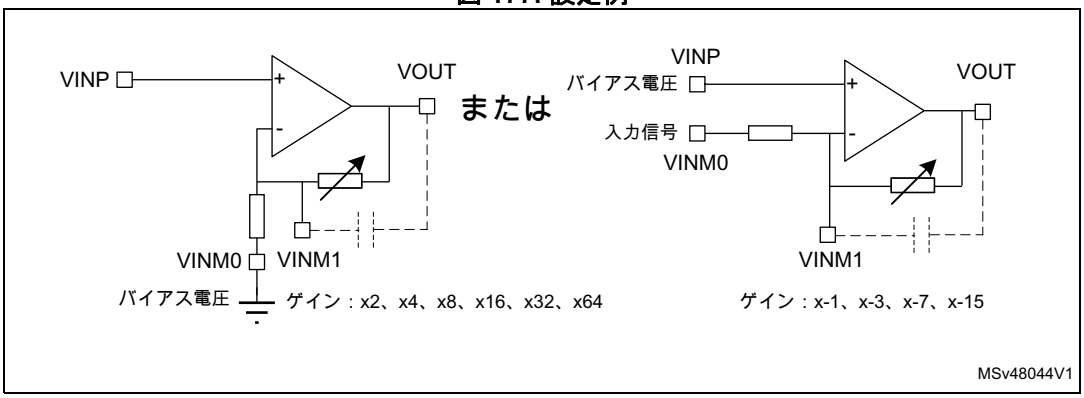
  VINM1 であらゆる外部接続が内部 PGA と並列で使用できます。たとえば、フィルタリング目的でコンデンサを opamp_out と VINM1 の間に接続できます（PGA レジスタネットワークで使用されるレジスタの値についてはデータシートを参照）。

図 176. PGA モード、非反転ゲイン設定（x2/x4/x8/x16/x32/x64）または反転ゲイン設定（x-1/x-3/x-7/x-15/x-31/x-63）フィルタ



1. opamp_out は、OPAINTOEN ビットをセットすることによって、内部で A/DC チャネルにリダイレクトできます。この場合、OPAMP x _VOUT 出力のマッピング先である I/O は未使用になるので、別の目的に使用できます。

図 177. 設定例



25.3.6 OPAMP PGA ゲイン

OPAMP が PGA モードとして設定された場合には、ゲインは非反転モードでは x2、x4、x8、x16、x32、x64、反転モードでは x-1、x-3、x-7、x-15、x-31、x-63 にプログラムできます。

OPAMP が非反転モードとして設定された場合には、ゲインは内部抵抗分圧器にのみ依存します。反転モードとして設定された場合には、ゲイン係数は、オンチップのフィードバック抵抗によってだけではなく、信号源の出力インピーダンスによっても定まります。PGA の入力フィードバック抵抗と比較して、信号源の出力インピーダンスが無視できない場合には、ゲイン誤差が生じます。

25.3.7 較正

OPAMP オフセット値は、トリミング回路を使用して最小化されます。起動時、トリミング値はプリセットされた「出荷時」トリミング値に初期化されます。オペアンプが工場のトリミング条件と異なる条件で使用される場合には、各オペアンプをユーザがトリミングすることもできます。

各オペアンプを、ユーザがトリミングすることもできます。特定のレジスタでは、通常モードとハイスピードモードで異なるトリミング値を持つことができます。

較正の目的は、OPAMP 入力オフセット電圧をできる限りキャンセルすることです。較正回路を使えば、安定した電圧条件と温度条件の範囲で、入力オフセット電圧を +/-3 mV 未満まで低減できます。

各オペアンプで、N 差動ペアについて 1 つ、P 差動ペアについて 1 つ、合計 2 つのトリミング値（OPAMPx_CSR レジスタの TRIMOFFSETP および TRIMOFFSETN）をトリミングする必要があります。

ユーザは、OPAMPx_CSR レジスタの USERTRIM ビットを使用して、「出荷時」の値から「ユーザ」がトリミングした値に切り替えることができます。このビットは起動時にリセットされるので、「出荷時」の値がデフォルトで OPAMP オプションレジスタに適用されます。

オフセットトリミング TRIMOFFSETP および TRIMOFFSETN ビットは、一般に、ビット CALON を 1 にセットすることによって、較正操作が初期化された後に設定されます。CALON = 1 の場合、オペアンプの入力は I/O から切断されます。

- CALSEL を 01 にセットすると、P 差動ペアのオフセット較正が初期化されます（低電圧基準使用）。
- CALSEL を 11 にリセットすると、N 差動ペアのオフセット較正が初期化されます（高電圧基準使用）。

CALON = 1 の場合、ビット CALOUT は CALSEL および OPAHSM で選択されたトリミング値の影響を反映します。ソフトウェアによって OPAMP 制御レジスタの TRIMOFFSETN ビットを 0x00 からインクリメントし、OPAMP 制御レジスタの CALOUT ビットが 1 から 0 に変わる最初の値に設定する必要があります。CALOUT ビットがリセットされると、オフセットが正しく較正されるので対応するトリミング値を格納する必要があります。CALOUT フラグは、トリミング値が変更され安定した後、最大 2 ms 必要とします（データシートの電気的特性セクションにある $t_{OFFTRIM}$ 最大遅延仕様を参照）。

注： **トリミング値が最適なトリミング値に近いほど、安定化にかかる時間が長くなります（いずれの場合でも最大安定時間が 2 ms 未満のままの状態）。**

表 202. 動作モードと較正

モード	制御ビット				出力	
	OPAEN	OPAHSM	CALON	CALSEL	V_{OUT}	CALOUT フラグ
通常動作モード	1	0	0	X	アナログ	0
ハイスピードモード	1	1	0	X	アナログ	0
パワーダウン	0	X	X	X	Z	0
オフセット較正 N	1	X	1	11	アナログ	X
オフセット較正 P 差分	1	X	1	01	アナログ	X

較正手順

オペアンプのいずれか 1 つを完全に較正するための手順を示します。

1. OPAMPx_CSR の OPAEN ビットを 1 にセットしてオペアンプを有効にします。
2. OPAMPx_CSR レジスタのUSERTRIM ビットを 1 にセットします。
3. 較正モードを選択します（表 202 : 動作モードと較正を参照）。手順 3 と 4 は 4 回繰り返す必要があります。繰り返しの 1 回目では通常モードと N 差動ペアを選択します。この較正モードは、OPAMPx_CSR レジスタの OPAHSM = 0 および CALSEL= 11 に対応します。
4. OPAMPx_CSR の CALOUT が 0 に変わるまで、OPAMPx_OTR の TRIMOFFSETN[4:0] を 0b00000 からインクリメントします。

注： **TRIMOFFSETP および TRIMOFFSETN ビットへの書込みと CALOUT 値の読出しの間は、正しい CALOUT 値を得るには、データシートの電気的特性セクションで指定された $t_{OFFTRIM}$ 最大遅延を待ってください。**

転流は、オフセットが正しく補正されることと、対応するトリミングコードを TRIMOFFSETP および TRIMOFFSETN ビットに保存する必要があることを示します。

次の場合、手順 3～4 を繰り返します。

- 通常モード、P 差動ペア、CALSEL = 01
- ハイスピード･モードと N 差動ペア
- ハイスピードモードと P 差動ペア

モードを使用しない場合は、対応する較正を実行する必要はありません。

すべてのオペアンプは、同時に較正できます。

注： **較正フェーズ全体において、オペアンプ出力の外部接続によって、500 µA を超えるプルアップまたはプルダウンはしないでください。**

OPAMP 出力が内部で ADC チャネルに接続され、出力ピンから切り離されている場合（OPAMPx_CSR レジスタの OPAINTOEN = 1）、OPAMP 出力が出力ピンに接続されている場合（OPAINTOEN = 0）とオフセット･トリミングの手順が異なります。較正手順は上記と同様ですが、手順 4 の CALOUT ビットの変化の検出は使用できません。代わりに、OPAMP 出力変化の指標として ADC の出力データを使用する必要があります。CALOUT の 1 から 0 への変化は、ADC データの最大出力に近い値から最小出力に近い値への変化に対応します（ADC が OPAMP 出力に接続されたコンパレータとして機能します）。もう一つの解決策は、OPAINTOEN = 0 で較正を実行してから、OPAINTOEN を 1 に変更する方法です。この場合、較正中に OPAMP 出力の GPIO に切り替えが発生するので、競合が発生しないように注意する必要があります。

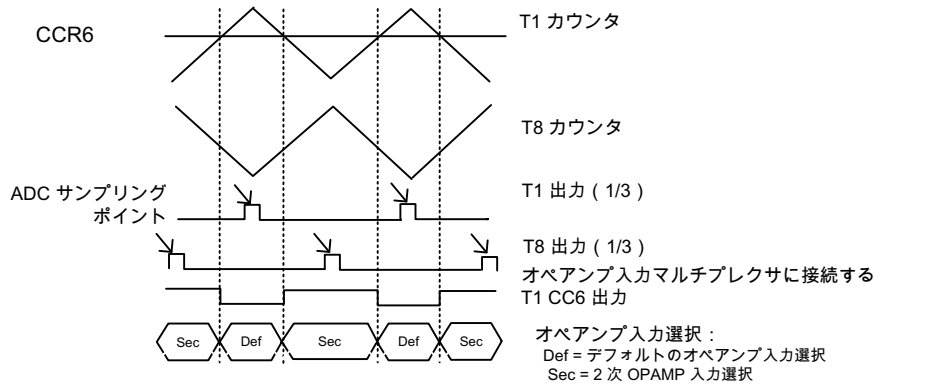
25.3.8 タイマによって制御されたマルチプレクサ･モード

OPAMP の反転および非反転入力を、自動で選択できます。この場合、1 つの入力から別の入力へ自動で切り替わります。この自動切替えは、OPAMP 入力マルチプレクサで発生する TIM1 CC6 または TIM8 CC6 または TIM20 CC6 出力によってトリガされます。

これは、最初のモータに次いで 2 番目のモータで 3 つのフェーズの電流を同時に測定する必要があるデュアルモータ制御で有用です。

自動切替えは、OPAMP 切替え制御レジスタの TxCMEN ビット(x = 1、8、20）をセットすることによって有効になります。反転および非反転入力の選択には、OPAMP タイマ制御モード･レジスタの VPS_SEL および VMS_SEL ビットフィールドを使用します。TxCMEN ビットがクリアされると、OPAMP 制御/ステータス･レジスタの VP_SEL および VM_SEL ビットフィールドに応じて入力が選択されます。

図 178. タイマによって制御されたマルチプレクサモード



25.4 OPAMP 低電力モード

表 203. 低消費電力モードが OPAMP に与える影響

モード	説明
SLEEP	影響はありません。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。
STOP 0 / STOP 1	影響はありません。OPAMP レジスタの内容は保たれます。
STANDBY	OPAMP レジスタはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

25.5 OPAMP レジスタ

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

アドレス･オフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP1_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP1_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP1_CSR は読み書き用です。

1：OPAMP1_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。
0：OPAMP 出力は出力ピンに接続されます。
1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード
この設定を変更するには、オペアンプを無効化する必要があります。
0：オペアンプは通常モードです。
1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択
00：VINM0 ピンを OPAMP VINM 入力に接続します。
01：VINM1 ピンを OPAMP VINM 入力に接続します。
10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。
11：opamp_out を OPAMP VINM 入力に接続します（フォロア･モード）。

ビット 4 **USERTRIM**：ユーザトリミング有効
このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。
このビットは、通常と高電力の両方のモードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択
00：VINP0 ピンを OPAMP1 VINP 入力に接続します。
01：VINP1 ピンを OPAMP1 VINP 入力に接続します。
10：VINP2 ピンを OPAMP1 VINP 入力に接続します。
11：DAC3_CH1 を OPAMP1 VINP 入力に接続します。

ビット 1 **FORCE_VP**：OPAMP VINP 入力の内部リファレンスに接続（テスト用に予約済み）
0：通常動作モード。非反転入力は入力に接続されます。
1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

25.5.2 OPAMP2 制御／ステータス･レジスタ（OPAMP2_CSR）

アドレス･オフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP2_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP2_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP2_CSR は読み書き用です。

1：OPAMP2_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。

0：OPAMP 出力は出力ピンに接続されます。

1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード

この設定を変更するには、オペアンプを無効化する必要があります。

0：オペアンプは通常モードです。

1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択

00：VINM0 ピンを OPAMP VINM 入力に接続します。

01：VINM1 ピンを OPAMP VINM 入力に接続します。

10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out は OPAMP VINM 入力（フォロアモード）に接続されています。

ビット 4 **USERTRIM**：ユーザトリミング有効

このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択

00：VINP0 ピンを OPAMP2 VINP 入力に接続します。

01：VINP1 ピンを OPAMP2 VINP 入力に接続します。

10：VINP2 ピンを OPAMP2 VINP 入力に接続します。

11：VINP3 ピンを OPAMP2 VINP 入力に接続します。

ビット 1 **FORCE_VP**：VP の 強制内部基準（テスト用に予約済み）

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

25.5.3 OPAMP3 制御／ステータス･レジスタ （OPAMP3_CSR）

アドレス･オフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP3_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP3_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP3_CSR は読み書き用です。

1：OPAMP3_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。
0：OPAMP 出力は出力ピンに接続されます。
1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード
この設定を変更するには、オペアンプを無効化する必要があります。
0：オペアンプは通常モードです。
1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択
00：VINM0 ピンを OPAMP VINM 入力に接続します。
01：VINM1 ピンを OPAMP VINM 入力に接続します。
10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。
11：opamp_out を OPAMP VINM 入力に接続します（フォロア･モード）。

ビット 4 **USERTRIM**：ユーザトリミング有効
このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。
このビットは、通常と高電力の両方のモードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択
00：VINP0 ピンを OPAMP3 VINP 入力に接続します。
01：VINP1 ピンを OPAMP3 VINP 入力に接続します。
10：VINP2 ピンを OPAMP3 VINP 入力に接続します。
11：DAC3_CH2 を OPAMP3 VINP 入力に接続します。

ビット 1 **FORCE_VP**：VP の 強制内部基準（テスト用に予約済み）
0：通常動作モード。非反転入力は入力に接続されます。
1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

25.5.4 OPAMP4 制御／ステータス･レジスタ（OPAMP4_CSR）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP4_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP4_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP4_CSR は読み書き用です。

1：OPAMP4_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。

0：OPAMP 出力は出力ピンに接続されます。

1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード

この設定を変更するには、オペアンプを無効化する必要があります。

0：オペアンプは通常モードです。

1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択

00：VINM0 ピンを OPAMP VINM 入力に接続します。

01：VINM1 ピンを OPAMP VINM 入力に接続します。

10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out を OPAMP VINM 入力に接続します（フォロア･モード）。

ビット 4 **USERTRIM**：ユーザトリミング有効

このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択

00：VINP0 ピンを OPAMP4 VINP 入力に接続します。

01：VINP1 ピンを OPAMP4 VINP 入力に接続します。

10：VINP2 ピンを OPAMP4 VINP 入力に接続します。

11：DAC4_CH1 を OPAMP4 VINP 入力に接続します。

ビット 1 **FORCE_VP**：VP の 強制内部基準（テスト用に予約済み）

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

25.5.5 OPAMP5 制御／ステータス･レジスタ （OPAMP5_CSR）

アドレス･オフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP5_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP5_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP5_CSR は読み書き用です。

1：OPAMP5_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。
0：OPAMP 出力は出力ピンに接続されます。
1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード
この設定を変更するには、オペアンプを無効化する必要があります。
0：オペアンプは通常モードです。
1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択
00：VINM0 ピンを OPAMP VINM 入力に接続します。
01：VINM1 ピンを OPAMP VINM 入力に接続します。
10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。
11：opamp_out は OPAMP VINM 入力（フォロアモード）に接続されています。

ビット 4 **USERTRIM**：ユーザトリミング有効
このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。
このビットは、通常と高電力の両方のモードでアクティブです。
0：「出荷時」トリミングコードを使用します。
1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択
00：VINP0 ピンを OPAMP5 VINP 入力に接続します。
01：VINP1 ピンを OPAMP5 VINP 入力に接続します。
10：VINP2 ピンを OPAMP5 VINP 入力に接続します。
11：DAC4_CH2 を OPAMP5 VINP 入力に接続します。

ビット 1 **FORCE_VP**：VP の 強制内部基準（テスト用に予約済み）
0：通常動作モード。非反転入力は入力に接続されます。
1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化
0：オペアンプ無効
1：オペアンプ有効

25.5.6 OPAMP6 制御／ステータス･レジスタ（OPAMP6_CSR）

アドレス･オフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	CAL OUT	Res.	TRIMOFFSETN[4:0]					TRIMOFFSETP[4:0]					PGA_GAIN[4:2]		
rw	r		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PGA_GAIN[1:0]		CALSEL[1:0]		CALON	Res.	Res.	OPA INTOEN	OPA HSM	VM_SEL[1:0]		USER TRIM	VP_SEL[1:0]		FORCE _VP	OPAEN
rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP6_CSR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP6_CSR レジスタを読出し専用に設定するために使用します。

0：OPAMP6_CSR は読み書き用です。

1：OPAMP6_CSR は読出し専用です。

ビット 30 **CALOUT**：オペアンプ較正出力

このビットは、OPAMP 出力のデジタル値を示し、較正オフセットモード時の較正出力ステータスです（CALOUT が 1 から 0 に切り替わると、較正は成功です。）

ビット 29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **TRIMOFFSETN[4:0]**：NMOS 差動ペアのトリミング

ビット 23:19 **TRIMOFFSETP[4:0]**：PMOS 差動ペアのトリミング

リセット時、これらのビットは出荷時トリミング値によってロードされます。それらは USER_TRIM=1 のときのみ変更できます。

ビット 18:14 **PGA_GAIN[4:0]**：オペアンプのプログラム可能なアンプ･ゲイン値

00000：非反転内部ゲイン = 2
00001：非反転内部ゲイン = 4
00010：非反転内部ゲイン = 8
00011：非反転内部ゲイン = 16
00100：非反転内部ゲイン = 32
00101：非反転内部ゲイン = 64
00110：未使用
00111：未使用
01000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアスに使用
01001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアスに使用
01010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアスに使用
01011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアスに使用
01100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアスに使用
01101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアスに使用
01110：未使用
01111：未使用
10000：非反転ゲイン = 2、VINM0 ピンでフィルタリング
10001：非反転ゲイン = 4、VINM0 ピンでフィルタリング
10010：非反転ゲイン = 8、VINM0 ピンでフィルタリング
10011：非反転ゲイン = 16、VINM0 ピンでフィルタリング
10100：非反転ゲイン = 32、VINM0 ピンでフィルタリング
10101：非反転ゲイン = 64、VINM0 ピンでフィルタリング
10110：未使用
10111：未使用
11000：反転ゲイン = -1／非反転ゲイン = 2、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11001：反転ゲイン = -3／非反転ゲイン = 4、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11010：反転ゲイン = -7／非反転ゲイン = 8、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11011：反転ゲイン = -15／非反転ゲイン = 16、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11100：反転ゲイン = -31／非反転ゲイン = 32、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11101：反転ゲイン = -63／非反転ゲイン = 64、VINM0 ピンを入力またはバイアス、フィルタリングに VINM1 ピンを使用
11110：未使用
11111：未使用

ビット 13:12 **CALSEL[1:0]**：較正選択

CALON = 1 または FORCE_VP= 1 の場合に、内部基準電圧の生成に使用するオフセット較正バスを選択するために使用されます。

00：OPAMP 入力に印加される 0.033*VDDA
01：OPAMP 入力に印加される 0.1*VDDA（PMOS較正用）
10：OPAMP 入力に印加される 0.5*VDDA
11：OPAMP 入力に印加される 0.9*VDDA（NMOS較正用）

ビット 11 **CALON**：較正モード有効化

0：通常モード
1：較正モード（すべてのスイッチはハードウェアによって開かれます）

ビット 10:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **OPAINTOEN**：オペアンプ内部出力イネーブル。

0：OPAMP 出力は出力ピンに接続されます。

1：OPAMP 出力は内部で ADC チャネルに接続され、出力ピンから切断されます。

ビット 7 **OPAHSM**：オペアンプ･ハイスピード･モード

この設定を変更するには、オペアンプを無効化する必要があります。

0：オペアンプは通常モードです。

1：オペアンプはハイ･スピード･モードです。

ビット 6:5 **VM_SEL[1:0]**：反転入力選択

00：VINM0 ピンを OPAMP VINM 入力に接続します。

01：VINM1 ピンを OPAMP VINM 入力に接続します。

10：フィードバック抵抗は OPAMP VINM 入力（PGA モード）に接続されています。反転入力の選択は PGA_GAIN 設定に依存します。

11：opamp_out は OPAMP VINM 入力（フォロアモード）に接続されています。

ビット 4 **USERTRIM**：ユーザトリミング有効

このビットによって、「出荷時」AOP オフセット･トリミング値から「ユーザ」AOP オフセット･トリミング値に切り替えることができます。

このビットは、通常と高電力の両方のモードでアクティブです。

0：「出荷時」トリミングコードを使用します。

1：「ユーザ」トリミングコードを使用します。

ビット 3:2 **VP_SEL[1:0]**：非反転入力選択

00：VINP0 ピンを OPAMP6 VINP 入力に接続します。

01：VINP1 ピンを OPAMP6 VINP 入力に接続します。

10：VINP2 ピンを OPAMP6 VINP 入力に接続します。

11：DAC3_CH1 を OPAMP6 VINP 入力に接続します。

ビット 1 **FORCE_VP**：VP の 強制内部基準（テスト用に予約済み）

0：通常動作モード。非反転入力は入力に接続されます。

1：較正検証モード：非反転入力は較正基準電圧に接続されます。

ビット 0 **OPAEN**：オペアンプ有効化

0：オペアンプ無効

1：オペアンプ有効

25.5.7 OPAMP1 タイマ制御モードレジスタ（OPAMPx_TCMR）（x = 1...6）

アドレス･オフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP1_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP1_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP1_TCMR は読み書き用です。

1：OPAMP1_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMPx の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMPx の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMPx の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP1 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMPx 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP1 VINP 入力に接続します。

01：VINP1 ピンを OPAMP1 VINP 入力に接続します。

10：VINP2 ピンを OPAMP1 VINP 入力に接続します。

11：DAC3_CH1 を OPAMP1 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP1 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMPx 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.8 OPAMP2 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6）

アドレス･オフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP2_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP2_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP2_TCMR は読み書き用です。

1：OPAMP2_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP2 の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP2 の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP2 の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP2 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP2 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP2 VINP 入力に接続します。

01：VINP1 ピンを OPAMP2 VINP 入力に接続します。

10：VINP2 ピンを OPAMP2 VINP 入力に接続します。

11：VINP3 ピンを OPAMP2 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP2 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMPx 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.9 OPAMP3 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6）

アドレス･オフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP3_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP3_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP3_TCMR は読み書き用です。

1：OPAMP3_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP3 の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP3 の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP3 の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP3 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP3 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP3 VINP 入力に接続します。

01：VINP1 ピンを OPAMP3 VINP 入力に接続します。

10：VINP2 ピンを OPAMP3 VINP 入力に接続します。

11：DAC3_CH2 を OPAMP3 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP3 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMPx 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.10 OPAMP4 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6）

アドレス･オフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP4_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP4_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP4_TCMR は読み書き用です。

1：OPAMP4_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP4 の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP4 の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP4 の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP4 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP4 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP4 VINP 入力に接続します。

01：VINP1 ピンを OPAMP4 VINP 入力に接続します。

10：VINP2 ピンを OPAMP4 VINP 入力に接続します。

11：DAC4_CH1 を OPAMP4 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP4 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP4 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.11 OPAMP5 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6）

アドレス･オフセット：0x28

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP5_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP5_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP5_TCMR は読み書き用です。

1：OPAMP5_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP5 の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP5 の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP5 の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP5 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP5 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP5 VINP 入力に接続します。

01：VINP1 ピンを OPAMP5 VINP 入力に接続します。

10：VINP2 ピンを OPAMP5 VINP 入力に接続します。

11：DAC4_CH2 を OPAMP5 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP5 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP5 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.12 OPAMP6 タイマ制御モード･レジスタ（OPAMPx_TCMR）（x = 1 ~ 6）

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL[1:0]		VMS_SEL
										rw	rw	rw	rw	rw	rw

ビット 31 **LOCK**：OPAMP6_TCMR ロック

このビットは 1 回のみ書込み可能です。ソフトウェアによってセットされます。システムリセットによってのみクリアされます。

このビットは、OPAMP6_TCMR レジスタを読出し専用に設定するために使用します。

0：OPAMP6_TCMR は読み書き用です。

1：OPAMP6_TCMR は読出し専用です。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **T20CM_EN**：TIM20 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP6 の入力マルチプレクサで発生する TIM20 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 4 **T8CM_EN**：TIM8 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP6 の入力マルチプレクサで発生する TIM8 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 3 **T1CM_EN**：TIM1 制御マルチプレクスモードイネーブル

このビットは、ソフトウェアによってセット／クリアされます。反転および非反転入力のデフォルト選択（VP_SEL および VM_SEL）と 2 次選択（VPS_SEL および VMS_SEL）の切り替えを自動的に制御するために使用します。この自動切替えは、OPAMP6 の入力マルチプレクサで発生する TIM1 CC6 出力によってトリガされます。

0：自動切替えは無効です。

1：自動切替えは有効です。

ビット 2:1 **VPS_SEL[1:0]**：OPAMP6 非反転入力の 2 次選択

これらのビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP6 非反転入力を選択するために使用されます。

00：VINP0 ピンを OPAMP6 VINP 入力に接続します。

01：VINP1 ピンを OPAMP6 VINP 入力に接続します。

10：VINP2 ピンを OPAMP6 VINP 入力に接続します。

11：DAC3_CH1 を OPAMP6 VINP 入力に接続します。

ビット 0 **VMS_SEL**：OPAMP6 反転入力の 2 次選択

このビットは、ソフトウェアによってセット／クリアされます。制御マルチプレクサモードが有効なとき（T1CM_EN = 1 または T8CM_EN = 1 または T20CM_EN = 1）、OPAMP6 反転入力を選択するために使用されます。

スタンドアロンモードが使用されるとき（VM_SEL = 00 または 01）：

0：VINM0 から入力

1：VINM1 から入力

PGA（VM_SEL = 10）またはフォロアモード（VM_SEL = 11）が使用されるとき：

0：抵抗フィードバック出力が選択されます（PGA モード）

1：V_{OUT} が入力マイナスとして選択されます（フォロア･モード）。

25.5.13 OPAMP レジスタ･マップ

表 204. OPAMP レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**OPAMP1_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0x04	**OPAMP2_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0x08	**OPAMP3_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0x0C	**OPAMP4_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**OPAMP5_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0x14	**OPAMP6_CSR**	LOCK	CALOUT	Res.	TRIMOFFSETN					TRIMOFFSETP					PGA_GAIN					CALSEL		CALON	Res.	Res.	OPAINTOEN	OPAHSM	VM_SEL		USERTRIM	VP_SEL		FORCE_VP	OPAEN
	リセット値	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0
0x18	**OPAMP1_ TCMR**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																									0	0	0	0	0	0
0x1C	**OPAMP2_ TCMR**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																									0	0	0	0	0	0
0x20	**OPAMP3_ TCMR**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																									0	0	0	0	0	0
0x24	**OPAMP4_ TCMR**	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																									0	0	0	0	0	0

表 204. OPAMP レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	OPAMP5_ TCMR	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																										0	0	0	0	0	0
0x2C	OPAMP6_ TCMR	LOCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	T20CM_EN	T8CM_EN	T1CM_EN	VPS_SEL		VMS_SEL
	リセット値	0																										0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

26 真性乱数発生器（RNG）

26.1 概要

RNG は、完全エントロピー出力を 32 ビットサンプルとしてアプリケーションに提供する真性乱数発生器です。ライブエントロピーソース（アナログ）と内部conditioning component(条件付けコンポーネント)から構成されています。

RNG は、ライブ･エントロピー･ソースとして作用する NIST 準拠の決定論的乱数発生器（DRBG）の構築に使用できます。

RNG の真性乱数発生器は、ドイツ BSI 文書の統計的検定 AIS-31 (T0 ～T8) 規格に従って試験を行いました。

26.2 RNG の主な機能

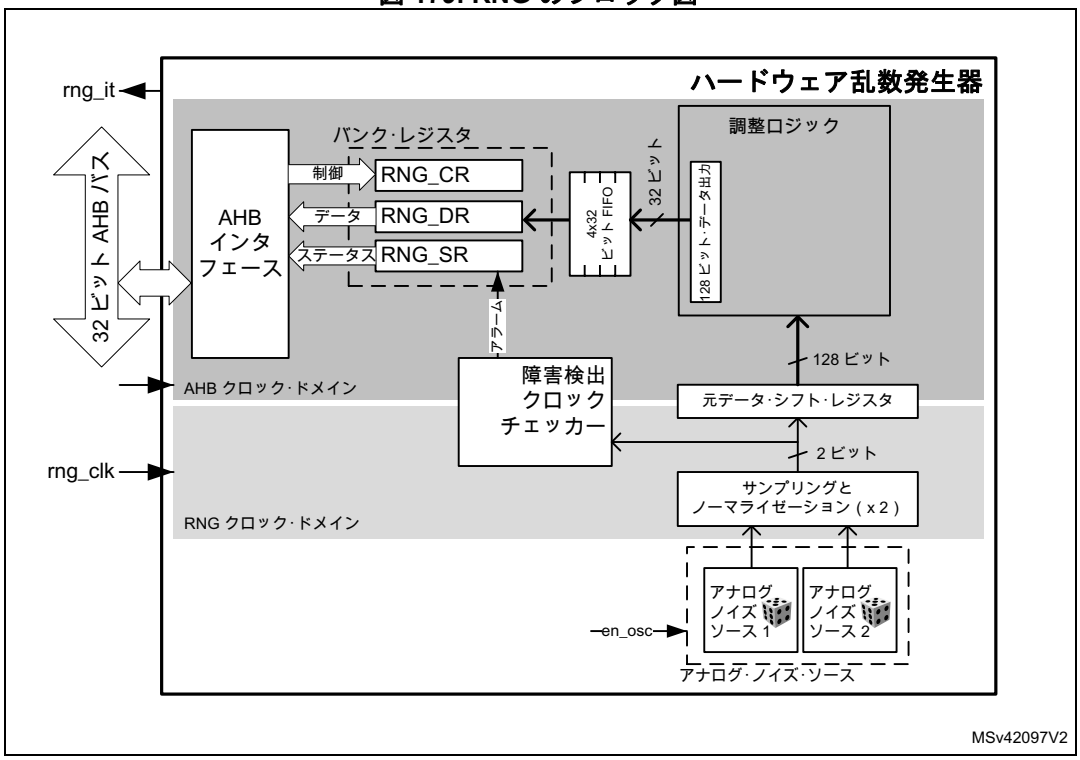
- RNG は、アナログ･エントロピー･ソースを高品質の調整技術によって処理して得られる 32 ビットの真性乱数を供給します。
- NIST 設定では、4 つの 32 ビット･ランダム･サンプルを16x $\frac{f_{AHB}}{f_{RNG}}$ AHB クロック･サイクルごとに生成します（ただし、このサイクル数の計算値が 213 サイクルを超える場合。この計算値が 213 サイクル以下になる場合は 213 サイクルごとに生成します）。
- 関連のエラー管理による基本的な健全性試験が組み込まれ、継続的に動作しています。
  - 非常に低いサンプリングクロック検出および反復カウント試験が搭載されています。
- 無効化して消費電力を低減することができます。
- 32 ビット･ワードのシングル･アクセスでのみアクセス可能な AMBA AHB スレーブ･ペリフェラルが搭載されています（32 ビット･ワード以外でのアクセスでは AHB バス･エラーが発生し、書込みアクセスは無視されます）。

26.3 RNG の機能説明

26.3.1 RNG のブロック図

図 179 に RNG のブロック図を示します。

図 179. RNG のブロック図



26.3.2 RNG 内部信号

表 205 では、RNG をより良く理解するために内部の信号のリストを記載しております。STM32（パッド上）の入出力の信号ではありません。

表 205. RNG 内部入力／出力信号

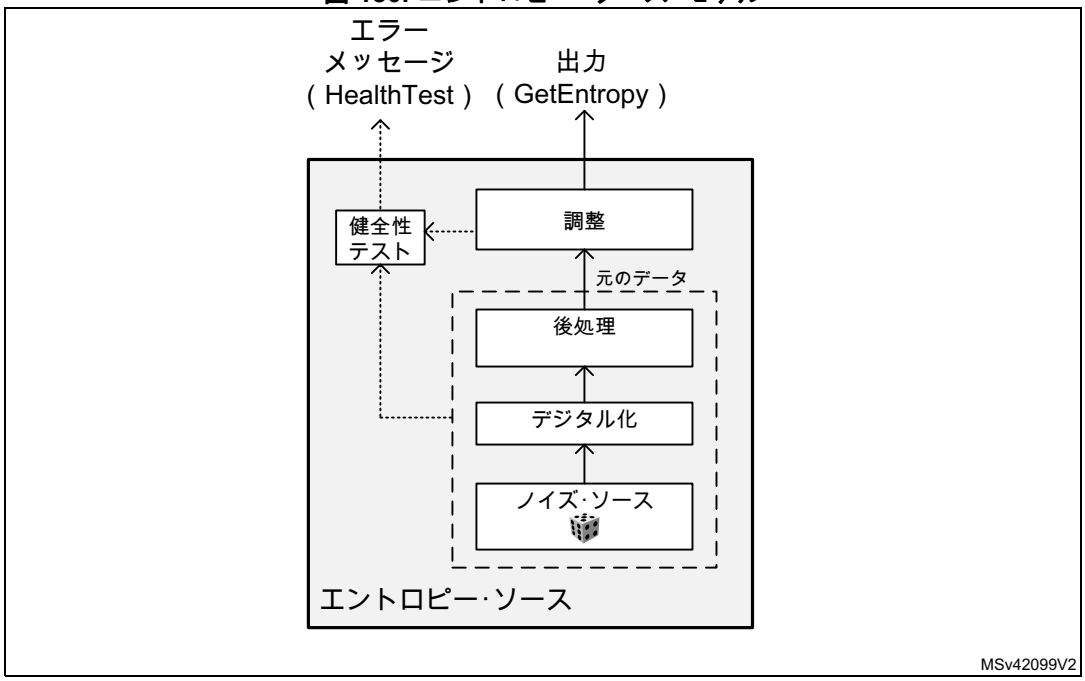
信号名	信号タイプ	説明
rng_it	デジタル出力	RNG グローバル割込みリクエスト
rng_hclk	デジタル入力	AHB クロック
rng_clk	デジタル入力	RNG 専用クロック、rng_hclk と非同期

26.3.3 乱数の生成

真性乱数発生器（RNG）は、決定論的な間隔で AHB インタフェースを介して真にランダムなデータを伝送します。RNG は、その境界内で図 180 に示すエントロピー·ソース·モデルを実装します。

このモデルは、アナログ·ノイズ·ソース、後処理を伴うデジタル化ステージ、調整アルゴリズム、健全性監視ブロック、およびエントロピー·ソースとの相互作用に使用する 2つのインタフェースである GetEntropy と HealthTest で構成されています。

図 180. エントロピー·ソース·モデル



上に図示されたコンポーネントの詳細を以下に示します。

ノイズ·ソース

ノイズソースは、非決定論的なエントロピーを生成する構成要素のノイズソースで、これは出力の不確実性に関連したものです（デジタル化されたノイズソースやソースデータ）。以下の要素で構成されます。

- それぞれXORされたフリーランのリングオシレータの出力 3 つに基づく 2 個のアナログノイズソースセクション 26.3.8 : RNG の低消費電力時の取り扱いに記載しているとおり、電力低減のためにこれらのアナログ·オシレータは無効化できます。
- 専用クロック入力（rng_clk）からクロック供給を受けるこれらの出力のサンプリング·ステージからは、2 ビットの元データが出力されます。

ノイズソースサンプリングは、AHB インタフェースクロック周波数（rng_hclk）に対して独立しています。

注： **セクション 26.6 : RNG エントロピー·ソース検証では、推奨される RNG クロック周波数が示されています。**

後処理

真性乱数ノイズソースから取得されるサンプル値は、2 ビットのビット列から成ります。このノイズソース出力はバイアスされるため、RNG が許容レベルまでそのバイアスを軽減する後処理コンポーネントを実行します。

より具体的に言えば、ノイズソースの 2 ビットそれぞれに対して、RNG は、サンプルノイズソースと反転したサンプルノイズソースからビットを半分ずつ取得します。こうして、ソースが“0”より“1”を多く生成する（またはその逆）場合、フィルタされます。

条件付け

RNG の条件付けコンポーネントは、結果として得られる固定長ビット列出力（128ビット）のエントロピー率を増加させる決定論的機能です。

32 ビット以上の元データを受信した場合、および出力 FIFO の再フィルが必要になった場合にも、後処理計算がトリガされます。したがって、64 RNG クロック･サイクル後にアプリケーションによって RNG の 128 ビット FIFO が空になると、RNG 出力のエントロピーは最大になります。

2 つの乱数生成の間に必要な時間､RNG を初期化してから最初のサンプルを利用できるまでに必要な時間については、セクション 26.5：RNG 処理時間に記載しています。

調整コンポーネントは、より高速な AHB クロックによりクロック制御されます。

出力バッファ

データ出力バッファは、調整コンポーネントから出力された 32 ビットワードを最大 4 つまで格納できます。RNG_DR レジスタを通じて出力 FIFO から 4 ワード読み出されると、128 ビット条件付け出力レジスタの内容は出力 FIFO にプッシュされ、新たな条件付けが自動的に始まります。213 AHB クロックサイクル後に、新たな 4 ワードが条件付け出力レジスタに追加されます。

RNG_DR レジスタから乱数が得られるようになると、DRDY フラグが“0”から“1”に遷移します。このフラグは､RNG_DR レジスタからの 4 ワードの読出しによって出力バッファが空になるまで“1”を保持します。

注： **割込みが有効の場合、このデータ･レディ･フラグが“0”から“1”に遷移するときに割込みが生成されます。その後、上記で説明したように割込みは RNG によって自動的にクリアされます。**

健全性チェック

このコンポーネントは、エントロピー･ソース全体（およびそのノイズ･ソース）が起動して想定どおりに動作していることを確認し、高い確率と信頼性で不具合を確実に検出できるようにします。

RNG は、以下のような健全性チェック機能を実装します。

1. ノイズ･ソースの出力で時間の制限がなく継続的に動作する健全性テスト
   - 以下の場合にエラー･フラグを生成する繰り返しカウント･テスト
   a) ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して提供しました。
   b) 両方のノイズソースが 32 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 16 ビット以上連続して提供しました。
2. ベンダ固有の連続した試験
   - リアルタイムの「遅すぎる」サンプリング･クロック検出回路。RNG クロックの 1 サイクルが 32 分周された AHB クロック･サイクルより短い場合にエラー･フラグを立てます。

エラー･コンディションが検出されると、RNG_SR レジスタの CECS および SECS ステータス･ビットにその状態が示されます。詳細は セクション 26.3.7：エラー管理を参照してください。

注： **エラーが検出された場合、割込みを生成することができます。**

26.3.4 RNG 初期化

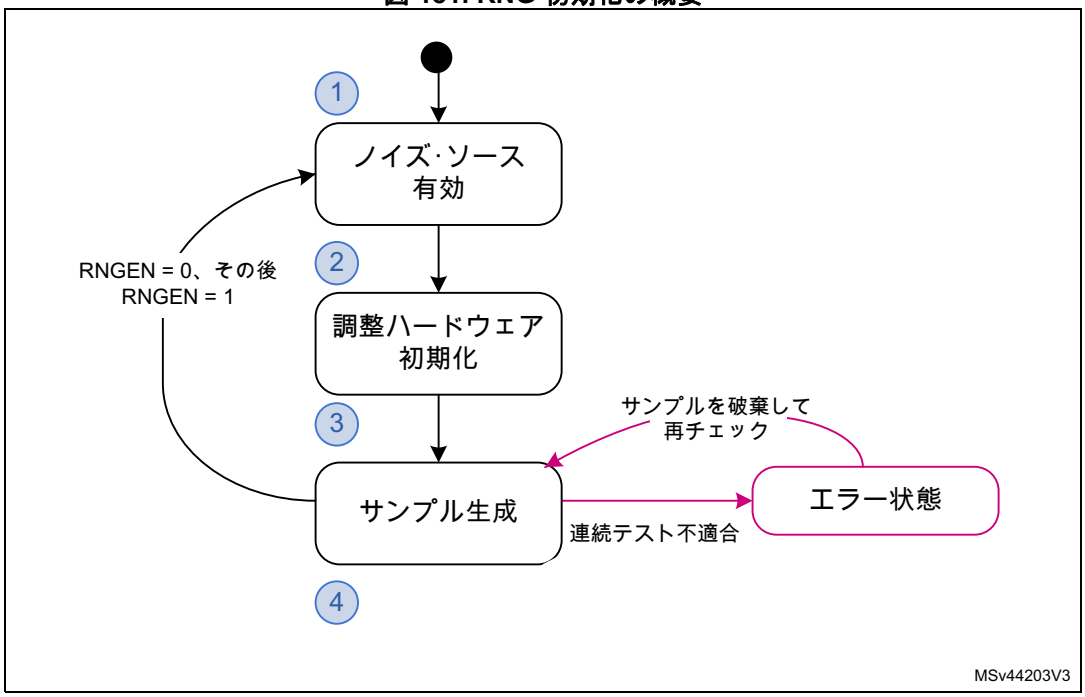
図 181 に RNG の簡易ステート･マシンを図示します。

RNG を有効化した後（RNG_CR の RNGEN = 1）、以下の一連のイベントが発生します。

1. アナログ･ノイズ･ソースが有効になり、その直後にロジックがアナログ出力のサンプリングを開始し、128 ビット調整シフト･レジスタに入力します。
2. 調整ロジックが有効になり、後処理のコンテキストが 2 つの 128 ノイズ･ソース･ビットを用いて初期化されます。
3. 調整ステージの内部入力データ･バッファに再度 128 ビットが入力され、調整処理の 1 ラウンドが実行されます。その後、出力バッファに後処理の結果が入力されます。
4. 出力バッファは RNG の使用方法に応じて自動的に再入力されます。

関連する初期化時間については、セクション 26.5：RNG 処理時間を参照してください。

図 181. RNG 初期化の概要



26.3.5 RNG 動作

通常動作

割込みを使用して RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの IE ビットをセットすることによって、割込みを有効にします。同時に、RNGEN=1 にセットすることにより、RNG を有効にします。
2. 乱数の準備ができたとき、またはエラーが発生したとき、割込みが生成されるようになります。そのため、割込みごとに次をチェックします。
   - 発生したエラーはありません。この場合は、RNG_SR レジスタの SEIS ビットと CEIS ビットが 0 に設定されている必要があります。
   - 乱数が準備できていること。この場合は、RNG_SR レジスタの DRDY ビットが 1 に設定されている必要があります。
   - 上記 2 つの条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容が読み出せるようになります。条件付け出力バッファに有効なデータがある場合には、アプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、セクション 26.3.7 : に記載されたエラー・リカバリ·シーケンスを使用する必要があります。

ポーリング·モードで RNG を作動させるには、次の手順を推奨します。

1. RNG_CR レジスタの RNGEN ビットを“1”にセットして、乱数発生を有効にします。
2. RNG_SR レジスタを読み出し、次を確認します。
   - エラーが発生していないこと（SEIS ビットと CEIS ビットが 0 に設定されていること）。
   - 乱数が準備できていること（DRDY ビットが 1 に設定されていること）。
3. 上記の条件に当てはまる場合、最大で連続 4 回まで RNG_DR レジスタの内容を読み出します。条件付け出力バッファに有効なデータがある場合にはアプリケーションによってさらに 4 ワード読み出すことができます（この場合には DRDY ビットはハイのままです）。上記の条件のどちらかまたは両方に当てはまらない場合、RNG_DR レジスタを読み出さないでください。エラーが発生した場合には、セクション 26.3.7 : に記載されたリカバリ·シーケンスを使用する必要があります。

注： **データが準備できていない場合（DRDY = 0）、RNG_DR は 0 を返します。**
RNG_DR がゼロではないことを必ず確認することをお勧めします。ゼロの場合、RNG_SR ポーリングと RND_DR 出力読出しの間にシードエラーが発生した（まれなイベント）ことを意味するからです。

低電力動作

消費電力がアプリケーションに対して懸念となる場合、セクション 26.3.8 : RNG の低消費電力時の取り扱いに示すように低電圧ストラテジーを使用できます。

ソフトウェア後処理

NIST が承認した 128 ビットのセキュリティ強度を持つ DRBG が必須である場合、RNG 真性乱数発生器を使用して認証済み乱数発生ソフトウェアを構築する必要があります。

内蔵されている健全性チェック機能の説明はセクション 26.3.3 : 乱数の生成を参照してください。

26.3.6 RNG クロック供給

RNG は、AHB バスクロックと専用 RNG クロックの 2 種類の異なるクロックで動作します。

AHB クロックは、AHB バンクレジスタおよび条件付けコンポーネントにクロック供給するために使用されます。RNG クロックは、ノイズソースサンプリングに使用されます。推奨されるクロック設定は、セクション 26.6 : RNG エントロピー・ソース検証で詳しく説明されています。

注： **RNG_CR レジスタの CED ビットが“0”に設定されている場合、RNG クロック周波数は 32 分周された AHB クロック周波数より高くする必要があります。そうしないと、クロック・チェッカーによって必ずクロック・エラー・フラグ（RNG_SR レジスタの CECS = 1）がセットされます。**

詳細（AHB および RNG クロック・ドメイン）については、セクション 26.3.1 : RNG のブロック図 を参照してください。

26.3.7 エラー管理

このセクションに詳細を示しているとおり、乱数生成と並行して、健全性チェックブロックにより、正しいノイズソース動作と RNG ソースクロックの周波数が検証されます。関連するエラー状態も記載されています。

クロックエラー検出

クロック・エラー検出が有効なとき（CED = 0）、RNG クロック周波数が低すぎる場合、RNG は CEIS と CECS の両方のビットを“1”に設定して、クロック・エラーが発生したことを示します。この場合、アプリケーションによって RNG クロックが正しく設定されていること（セクション 26.3.6 : RNG クロック供給を参照）をチェックしたうえで、CEIS ビットの割込みフラグをクリアする必要があります。CECS ビットは、クロック条件が正常なときに自動的にクリアされます。

注： **クロック・エラーは、生成済みの乱数には影響しないため、アプリケーションは引き続き RNG_DR レジスタを読み出すことができます。**

CEIS は、RNG によって CECS が 1 に設定されたときにのみセットされます。

ノイズ・ソース・エラー検出

ノイズ・ソース（つまりシード）エラーが発生すると、RNG は乱数生成を停止し、SEIS および SECS ビットを両方“1”に設定して、シード・エラーが発生したことを示します。このとき RNG_DR レジスタに値が用意されていても、エントロピーが十分ではない可能性があるので使用しないようにします。初期化フェーズ中にエラーが検出された場合、RNG によって初期化シーケンス全体が自動的に再スタートされます。

RNG 初期化後のシード・エラーからの完全なリカバリを行うには、次のシーケンスを使用します。

1. SEIS に“0”を書き込んでクリアします。
2. RNG_DR レジスタから 12 ワード読み出し、各ワードを破棄してパイプラインをきれいにします。
3. SEIS がクリアされたままであることを確認します。乱数発生は正常に戻りました。

26.3.8 RNG の低消費電力時の取り扱い

消費電力が懸念される場合、DRDY ビットが“1”に設定され次第、RNG_CR レジスタの RNGEN ビットを“0”に設定することで RNG を無効化できます。RNGEN = 0 であっても、後処理ロジックと出力バッファは動作を続けているので、ソフトウェアは次の機能を使用できます。

- 出力バッファに有効なワードが存在していれば、RNG_DR レジスタからさらに 4 個の乱数を読み出すことができます。
- 条件付け出力内部レジスタに有効なビットが存在する場合には、RNG_DR レジスタから乱数をもう 4 個読み出すことができます。有効なビットが存在しない場合は、アプリケーションから RNG を再度有効化し、ノイズ･ソースから新たに 32 ビット以上を取得して、調整ラウンドを完了させる必要があります。新たなビットのサンプリングには 16 RNG クロック･サイクル、調整ラウンド 1 回には 216 AHB クロック･サイクルを要します。

RNG を無効化する場合、すべてのアナログシード発生回路を無効にします。その消費電力は、データシートの電気的特性セクションに示されています。また、RNG クロックによってクロックの供給を受けるロジックはすべてゲートします。この方法では、RNG 初期化時間のために、RNG_DR レジスタでランダムサンプルが利用可能となるまでのレイテンシが増加することに注意してください。

初期化中に（DRDY ビットが初めて 1 に設定されるよりも十分前の時点）RNG ブロックが無効になった場合は、RNGEN ビットが“1”に設定されると、初期化シーケンスが、停止した時点の状態から再開されます。

26.4 RNG 割込み

RNG では、割込みは次のイベントによって生成できます。

- データレディフラグ
- シードエラー、セクション 26.3.7：エラー管理を参照
- クロックエラー、セクション 26.3.7：エラー管理を参照

表 206 に示すように、専用割込みイネーブル制御ビットが使用できます。

表 206. RNG 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法
RNG	データ･レディ･フラグ	DRDY	IE	なし（自動）
	シードエラーフラグ	SEIS	IE	SEIS に 0 を書き込みます。
	クロック･エラー･フラグ	CEIS	IE	CEIS に 0 を書き込みます。

RNG_CR レジスタのマスクビットまたは全体的な割込み制御ビット IE を変更することにより、上記の割込みソースを個別に有効にしたり無効にしたりすることができます。個別の割込みソースのステータスは RNG_SR レジスタから読み出すことができます。

注： **割込みは、RNG が有効化されている場合にのみ生成されます。**

26.5 RNG 処理時間

$16 \times \frac{f_{AHB}}{f_{RNG}}$ クロック･サイクルの値が 213 サイクルを超えている場合、調整ステージでは、このサイクルごとに 4 つの 32 ビット乱数を生成できます。この値が 213 サイクル以下の場合は、213 サイクルごとに生成できます。

デバイスがリセットを抜けた後初回の乱数にはそれ以上の時間が必要です（セクション 26.3.4 : RNG 初期化参照）。実際に初めて RNG を有効化した後、次のいずれかの時間の後に初めてランダムデータが利用可能となります。

- 128 RNG クロックサイクル + 426 AHB サイクル（$f_{AHB} < f_{threshold}$ の場合）
- 192 RNG クロック･サイクル + 213 AHB サイクル（$f_{AHB} \geq f_{threshold}$ の場合）

ここで、$f_{threshold} = (213 \times f_{RNG})/ 64$

26.6 RNG エントロピー･ソース検証

26.6.1 概要

RNG から使用できるエントロピーの量を評価するため、STMicroelectronics は ドイツ BSI AIS-31の統計的検定（T0～T8）に従って、ペリフェラルを試験しました。結果は要求に応じて提供でき、顧客側でテストを再現できます。

26.6.2 検証条件

STMicroelectronics は、以下の条件下で RNG 真性乱数発生器をテストしました。

- RNG クロック rng_clk= 48 MHz（RNG_CR レジスタの CED ビット = ’0’）および rng_clk = 400 kHz（RNG_CR レジスタの CED ビット = ‘1’）

26.6.3 データ収集

統計テストを行うためには、サンプルを元データレベルでのエントロピーソースからだけではなく、エントロピーソースの出力からも収集する必要があります。

製品で上記サンプルを取得する必要がある場合は、ST マイクロエレクトロニクスまでご連絡ください。

26.7 RNG レジスタ

RNG は、制御レジスタ、データレジスタ、ステータスレジスタと連動します。

26.7.1 RNG 制御レジスタ（RNG_CR）

アドレスオフセット：0x000

リセット値：0x0000 0000

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

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CED**：クロック･エラー検出

0：クロックエラー検出は有効です。
1：クロックエラー検出は無効です。
RNG を有効にすると、動作中はクロック･エラー検出の有効化も無効化もできません。CED を有効化または無効化するには、RNG を無効にする必要があります。

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **IE**：割込みイネーブル

0：RNG 割込みは無効です。
1：RNG 割込みは有効です。RNG_SR レジスタで DRDY = 1、SEIS = 1 または CEIS = 1 になると、割込みが直ちにペンディングになります。

ビット 2 **RNGEN**：真性乱数発生器イネーブル

0：真性乱数発生器は無効です。アナログノイズソースの電力がオフになり、RNG クロックによって供給されるロジックがゲートされます。
1：真性乱数発生器は有効です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

26.7.2 RNG ステータスレジスタ（RNG_SR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
									rc_w0	rc_w0			r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **SEIS**：シード･エラー割込みステータス

このビットは、SECS として同じ時間でセットされます。0 を書き込むことによってクリアされます。1 を書き込んでも、ビットの値は変化しません。

0：障害シーケンスは検出されませんでした。

1：1 つ以上の障害シーケンスが検出されました。詳細については、**SECS** ビットの説明を参照してください。

RNG_CR レジスタで IE = 1 である場合、割込みがペンディングになります。

ビット 5 **CEIS**：クロック･エラー割込みステータス

このビットは、CECS として同じ時間でセットされます。0 を書き込むことによってクリアされます。1 を書き込んでも、ビットの値は変化しません。

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。

1：遅すぎる RNG クロックが検出されました（fRNGCLK < fHCLK/32）。

RNG_CR レジスタで IE = 1 である場合、割込みがペンディングになります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **SECS**：シード･エラーの現在のステータス

0：障害のあるシーケンスは現在のところ検出されていません。SEIS ビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。

1：次の障害シーケンスのうち 1 つ以上が検出されました。

- ノイズソースの 1 つが 64 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 32 ビット以上連続して提供しました。
- 両方のノイズソースが 32 ビット以上連続して一定値（“0”または“1”）または、2 つのビットパターン（“01”や“10”）を 16 ビット以上連続して提供しました。

ビット 1 **CECS**：クロック･エラーの現在のステータス

0：正しい RNG クロックです（fRNGCLK > fHCLK/32）。CEIS ビットがセットされている場合、これは遅いクロックが検出されたが、回復していることを意味します。

1：遅すぎる RNG クロックです（fRNGCLK< fHCLK/32）。

注： CECS ビットは、RNG_CR レジスタの CED ビットが 0 にセットされたときにのみ有効です。

ビット 0 **DRDY**：データ･レディ

0：RNG_DR レジスタがまだ有効でなく、乱数データは利用できません。

1：RNG_DR レジスタに有効な乱数データが入っています。

出力バッファが空になると（RNG_DR レジスタの読出し後）、新しい乱数値が生成されるまで、このビットは 0 に戻ります。

注： ペリフェラルを無効にしている場合（RNG_CR レジスタの RNGEN = 0）、DRDY ビットが 1 になることがあります。

RNG_CR レジスタで IE = 1 の場合、DRDY = 1 になると割込みが生成されます。

26.7.3 RNG データ・レジスタ（RNG_DR）

アドレスオフセット：0x008

リセット値：0x0000 0000

RNG_DR レジスタは読出し専用レジスタであり、読み出すと 32 ビットの乱数値が返されます。このレジスタは、いったん読み出されると、出力 FIFO が空の場合、AHB クロック 216 周期の後、新しい乱数値を出力します。

このレジスタの内容は、RNGEN = 0 の場合でも、DRDY = 1 で、内容の値が 0x0 でないときは有効です。

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

ビット 31:0 **RNDATA[31:0]**：乱数データ

DRDY = 1 であれば有効な 32 ビット乱数データ。DRDY = 0 のとき、RNDATA の値は 0 です。

DRDY がセットされたら必ず RNG_DR がゼロ以外であることを確認することをお勧めします。ゼロの場合、RNG_SR のポーリングと RND_DR 出力読出しの間でシード・エラーが発生したことを意味するからです（その発生はまれです）。

26.7.4 RNG レジスタ・マップ

表 207. RNG レジスタ・マップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	RNG_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CED	Res.	IE	RNGEN	Res.	Res.
	リセット値																											0		0	0		
0x004	RNG_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SEIS	CEIS	Res.	Res.	SECS	CECS	DRDY
	リセット値																										0	0			0	0	0
0x008	RNG_DR	RNDATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2：を参照してください。

27 高分解能タイマ（HRTIM）

27.1 概要

高分解能タイマは高精度のタイミングで最大 12 個のデジタル信号を生成できます。主にスイッチモード電源や照明システムのような電力変換システムを駆動することを目的にしていますが、非常に細かいタイミングの分解能が期待される場合の汎用の用途にも使用できます。

モジュラー型のアーキテクチャにより、独立の波形または連動波形のどちらでも生成することができます。波形は（カウンタと比較ユニットを使用した）自己完結型のタイミングと、アナログまたはデジタルのフィードバックおよび同期信号などの広範囲な外部イベントによって定義されます。このため、多様な制御信号（PWM、位相シフト、コンスタント Ton、等）を生成したり、ほとんどの変換トポロジに対応することができます。

制御および監視の目的のために、タイマはタイミング測定の能力も持っており、内蔵の ADC および DAC コンバータにつながっています。最後に、タイマは軽負荷のマネージメントモードを備えており、安全に停止する目的のためにさまざまな障害を取り扱うことができます。

27.2 主な特徴

- 高分解能のタイミングユニット
  - 電圧変動と温度変動に対して補正可能で、184 ps の分解能
  - 高分解能はすべての出力で利用可能で、デューティサイクル、周波数、およびトリガされたワンパルスモードでのパルス幅を調整可能
  - 6 個の 16 ビットタイミングユニット（それぞれ独立のカウンタと 4 つの比較ユニットを装備）
  - どのタイミングユニットからも制御できる 12 個の出力、チャネルあたり最大 32 のセット／リセットソース
  - 1 または 2 スイッチを備える複数の独立コンバータか、少数の大型マルチスイッチ接続形態のどちらかに対応できるモジュラー型アーキテクチャ
- どのタイミングユニットでも利用可能な最大 10 個の外部イベント
  - プログラム可能な極性およびエッジ感度
  - 高速非同期モードでの 5 つのイベント
  - プログラム可能なデジタルフィルタを備えた 5 つのイベント
  - ブランキングおよびウィンドウイングモードによる誤ったイベントのフィルタリング
- 内蔵アナログペリフェラルへの複数リンク
  - ADC コンバータへの 4 つのトリガ
  - DAC コンバータへの 3 つのトリガ
  - アナログ信号調節用の 3 つのコンパレータ
- 多目的の保護スキーム
  - 6 つのフォールト入力をどのタイミングユニットにも結合し、関連付け可能
  - プログラム可能な極性、エッジ感度、およびプログラム可能なデジタルフィルタ
  - 共振コンバータ専用の遅延保護
- 複数の HRTIM インスタンスは外部同期入力／出力によって同期できます。
- 多様な出力ステージ
  - 高分解能のデッドタイム挿入（最小 735 ps まで）
  - プログラム可能な出力極性
  - チョッパモード
- 複数のコンバータで同期した軽負荷の動作を行わせるバーストモードコントローラ
- 8 つの割込みベクトル、それぞれ最大 14 ソース
- 複数レジスタの更新のバーストモードで、最大 14 個のソースによる 7 つの DMA リクエスト

27.3 機能詳細

27.3.1 概要

HRTIM は次の下位構成要素に分けることができます。

- マスタタイマ
- タイミングユニット（タイマ A～タイマ F）
- 出力ステージ
- バーストモードコントローラ
- すべてのタイマに共用される、外部イベントおよびフォールト信号の調整ロジック
- システムインタフェース

マスタタイマは 16 ビットのアップカウンタに基づいています。4 つの比較ユニット経由の 12 の出力のどれでもセット／リセットすることができ、また同期信号を 6 つのタイマユニットに供給します。主な目的は、それらタイマユニットを唯一のソースによって制御することです。マスタタイマが複数のユニット間の位相シフトを管理する典型的な適用例としては、インターリーブ型降圧コンバータがあります。

タイマユニットは独立に動作することも、マスタタイマも含めて他のタイマと連動して動作することもあります。各タイマには 2 つの出力用のコントロールがあります。出力のセット／リセットイベントは、タイミングユニットの比較レジスタによって、またはマスタタイマ、他のタイマ、あるいは外部イベントから到着するイベントによってトリガされます。

出力ステージにはいくつかの役目があります。

- 2 つの出力が相補 PWM モードに設定されているとき、デッドタイムを追加
- 変調信号の先頭にキャリア周波数を付加
- 出力を事前設定した安全なレベルに非同期にアサートすることによってフォールトイベントを管理

軽負荷の動作の場合、バーストモードコントローラが 1 つまたは複数のタイマを制御することができます。出力のアイドル状態だけでなく、バースト長および周期をプログラムすることが可能です。

外部イベントおよびフォールト信号の調整ロジックには次のものが含まれます。

- 入力選択 MUX（たとえば、デジタル入力か、所定の外部イベントチャネル向けのオンチップソースかの選択）
- 極性およびエッジ感度のプログラミング
- デジタルフィルタリング（12 チャネルのうち 5 チャネル）

システムインタフェースによって、HRTIM は以下の MCU の他の部分とのやり取りができます。

- CPU への割込みリクエスト
- HRTIM の特定バーストモードなど、メモリへの／メモリからの自動アクセスのための DMA コントローラ
- ADC および DAC コンバータへのトリガ

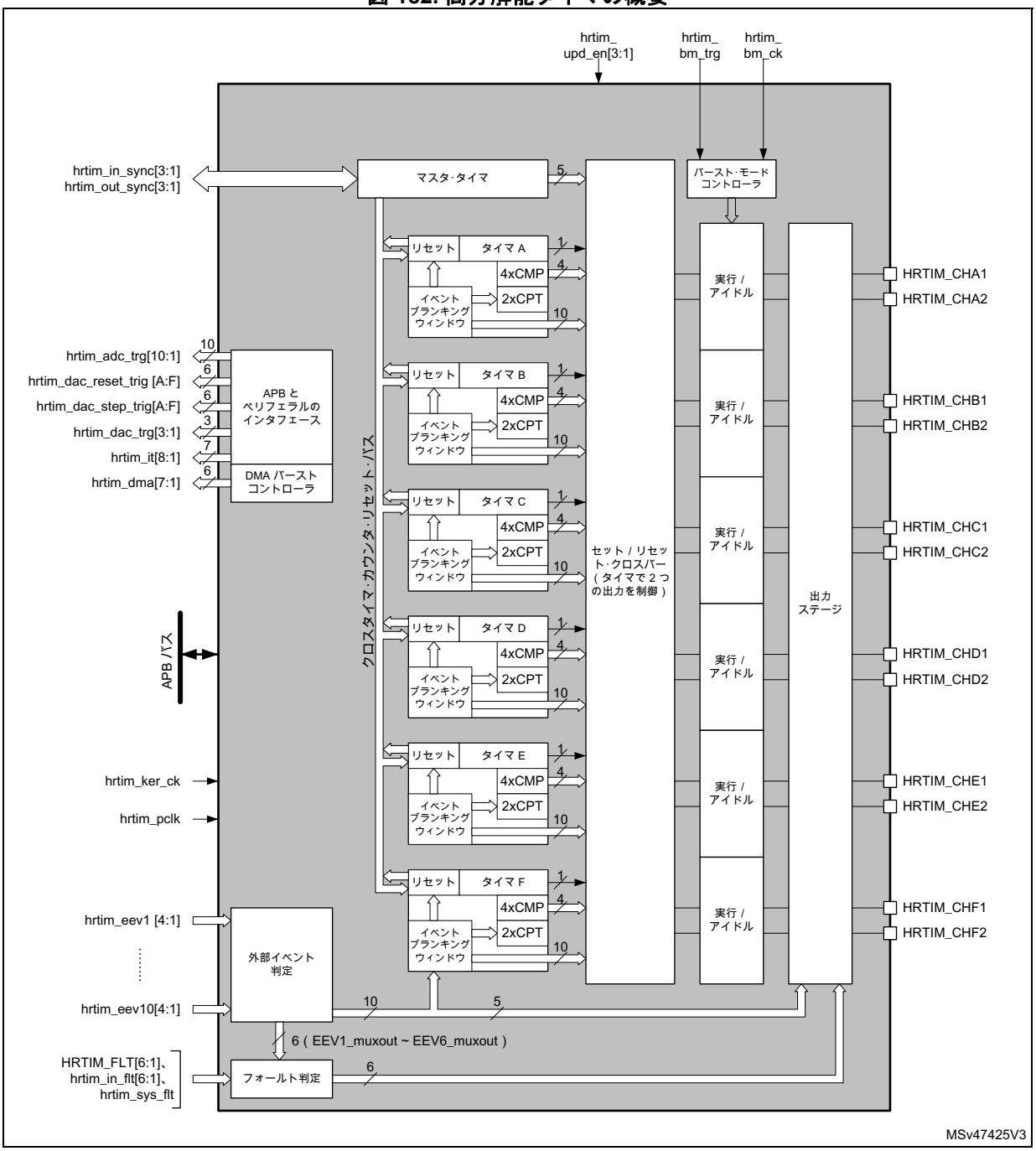
HRTIM レジスタは次の 8 つのグループに分かれます。

- マスタタイマのレジスタ
- タイマ A～タイマ F のレジスタ
- すべてのタイマユニットに共有される機能向けの共通レジスタ

注： **表記法として、本文中およびレジスタ内で 6 つのタイミングユニットを参照する場合、「x」の文字を使用して一般化します。ここで、x は A～F のどの値も取ることができます。**

タイマのブロック図を図 182 に示します。

図 182. 高分解能タイマの概要



27.3.2 HRTIM ピンおよび内部信号

このセクションの以下の表は、HRTIM の入力および出力（オンチップとオフチップの両方）の一覧です。

表 208. HRTIM の入出力の概要

信号名	信号タイプ	説明
HRTIM_CHA1 HRTIM_CHA2、 HRTIM_CHB1、 HRTIM_CHB2、 HRTIM_CHC1、 HRTIM_CHC2、 HRTIM_CHD1、 HRTIM_CHD2、 HRTIM_CHE1、 HRTIM_CHE2、 HRTIM_CHF1、 HRTIM_CHF2	出力	HRTIM タイマのメイン出力これらはデッドタイムを挿入してペア（HRTIM_CHx1 と HRTIM_CHx2）で連動させることも、独立に動作させることもできます。
hrtim_in_flt1[4:1] hrtim_in_flt2[4:1] hrtim_in_flt3[4:1] hrtim_in_flt4[4:1] hrtim_in_flt5[4:1] hrtim_in_flt6[4:1]	デジタル入力	フォールト入力：アサートされた場合、直ちに HRTIM の出力を無効化します（12 個のオンチップ入力と 6 つのオフチップ HRTIM_FLTx 入力）。
hrtim_sys_flt	デジタル入力	MCU の内部フォールトを収集するシステム障害イベント（クロックセキュリティシステム、SRAM パリティエラー、FPU 搭載 Cortex®-M4 ロックアップ（ハードフォールト）、PVD 出力）
hrtim_in_sync[3:1]	デジタル入力	HRTIM 全体を下記の他の内部または外部タイマリソースと同期するための同期入力： hrtim_in_sync1：予約済みです。 hrtim_in_sync2：ソースは TIM1_TRGO 出力です。 hrtim_in_sync3：ソースは HRTIM_SCIN 入力ピンです。
hrtim_out_sync[2:1]	デジタル出力	この出力の目的は、いくつかの HRTIM インスタンスをカスケードにする、または同期することです（オンチップまたはオフチップ）。 hrtim_out_sync1： 予約済みです。 hrtim_out_sync2： 出力先はオフチップの HRTIM または ペリフェラル（HRTIM_SCOUT 出力ピン経由）。

表 208. HRTIM の入出力の概要 （続き）

信号名	信号タイプ	説明
hrtim_eev1[4:1]	デジタル入力	外部イベント。10 個のイベントのそれぞれは、オンチップ（他の内蔵ペリフェラル : コンパレータ、ADC アナログウォッチドッグ、TIMx タイマ、トリガ出力）またはオフチップ（HRTIM_EEVx 入力ピン）の 4 つのソースから選択することが可能。
hrtim_eev2[4:1]		
hrtim_eev3[4:1]		
hrtim_eev4[4:1]		
hrtim_eev5[4:1]		
hrtim_eev6[4:1]		
hrtim_eev7[4:1]		
hrtim_eev8[4:1]		
hrtim_eev9[4:1]		
hrtim_eev10[4:1]		
hrtim_upd_en[3:1]	デジタル入力	更新イネーブル入力 hrtim_upd_end[3:1]（オンチップ相互接続）のパルスはシャドウレジスタからアクティブレジスタへの転送をトリガします。
hrtim_bm_trg	デジタル入力	この入力のパルスがバーストモードへの移行をトリガします。この入力は TIM7_TRGO 出力に接続されています。
hrtim_bm_ck[4:1]	デジタル入力	バーストモードクロック（オンチップ相互接続）
hrtim_adc_trg[10:1]	デジタル出力	ADC 変換開始トリガ。hrtim_adc_trg1～hrtim_adc_trg10 出力は、これ以降、ADC トリガ 1～ADC トリガ 10 と呼びます。
hrtim_dac_trg[3:1]	デジタル出力	DAC 変換更新トリガ
hrtim_dac_reset_trg[6:1] hrtim_dac_step_trg[6:1]	デジタル出力	デュアルチャネル DAC トリガ
hrtim_it[8:1]	デジタル出力	割込みリクエスト
hrtim_dma[7:1]	デジタル出力	DMA リクエスト
hrtim_pclk	-	APB クロック
hrtim_ker_ck	-	HRTIM カーネルクロック（APB2 クロック）

表 209. 外部イベントのマッピングおよび関連機能

外部イベントチャネル	高速モード	デジタルフィルタ	バランスフォールトタイマ A、B、C	バランスフォールトタイマ D、E、F	ソース（EExSRC[1:0]）				パッケージごとの利用可能なコンパレータと入力ソース	
					Src1 (00)	Src 2 (01)	Src3 (10)	Src4 (11)	48 ピン	64 ピンおよび 100 ピン
hrtim_eev1[4:1]	可能	-	-	-	PC12	COMP2	TIM1_TRGO	ADC1_AWD1	Comp	Comp & 入力
hrtim_eev2[4:1]	可能	-	-	-	PC11	COMP4	TIM2_TRGO	ADC1_AWD2	Comp	Comp & 入力
hrtim_eev3[4:1]	可能	-	-	-	PB7	COMP6	TIM3_TRGO	ADC1_AWD3	Comp & 入力	Comp & 入力
hrtim_eev4[4:1]	可能	-	-	-	PB6	COMP1	COMP5	ADC2_AWD1	Comp & 入力	Comp & 入力
hrtim_eev5[4:1]	可能	-	-	-	PB9	COMP3	COMP7	ADC2_AWD2	Comp & 入力	Comp & 入力
hrtim_eev6[4:1]	-	可能	可能	-	PB5	COMP2	COMP1	ADC2_AWD3	Comp & 入力	Comp & 入力
hrtim_eev7[4:1]	-	可能	可能	-	PB4	COMP4	TIM7_TRGO	ADC3_AWD1	Comp & 入力	Comp & 入力
hrtim_eev8[4:1]	-	可能	-	可能	PB8	COMP6	COMP3	ADC4_AWD1	Comp & 入力	Comp & 入力
hrtim_eev9[4:1]	-	可能	-	可能	PB3	COMP5	TIM15_TRGO	COMP4	Comp & 入力	Comp & 入力
hrtim_eev10[4:1]	-	可能	-	-	PC5/PC6	COMP7	TIM6_TRGO	ADC5_AWD1	-	Comp & 入力

表 210. 更新イネーブル入力とソース

hrtim_upd_en[3:1]	更新ソース
hrtim_upd_en1	TIM16_OC
hrtim_upd_en2	TIM17_OC
hrtim_upd_en3	TIM6_TRGO

表 211. 汎用タイマからのバースト･モードのクロック･ソース

hrtim_bm_ck[4:1]	BMCLK[3:0]	クロック･ソース
hrtim_bm_ck1	0110	TIM16 OC
hrtim_bm_ck2	0111	TIM17 OC
hrtim_bm_ck3	1000	TIM7 TRGO
hrtim_bm_ck4	1001	予約済みです。

表 212. フォールト入力

フォールト チャネル	外部入力 FLTxSRC[1:0] = 00	オンチップソース FLTxSRC[1:0] = 01	外部入力 FLTxSRC[1:0] = 10	オンチップソース FLTxSRC[1:0] = 11
hrtim_in_flt1[4:1]	PA12	COMP2	EEV1	N/A
hrtim_in_flt2[4:1]	PA15	COMP4	EEV2	N/A
hrtim_in_flt3[4:1]	PB10	COMP6	EEV3	N/A
hrtim_in_flt4[4:1]	PB11	COMP1	EEV4	N/A
hrtim_in_flt5[4:1]	PB0/PC7	COMP3	EEV5	N/A
hrtim_in_flt6[4:1]	PC10	COMP5	EEV6	N/A

表 213. HRTIM DAC トリガ接続

HRTIM DAC トリガ	DAC1_CH1 DAC_CH2	DAC2_CH1	DAC3_CH1 DAC3_CH2	DAC4_CH1 DAC4_CH2
hrtim_dac_trg1	可能	-	-	可能
hrtim_dac_trig2	-	可能	-	-
hrtim_dac_trig3	-	-	可能	-
hrtim_dac_reset_trg1 hrtim_dac_step_trg1	可能	可能	可能	可能
hrtim_dac_reset_trg2 hrtim_dac_step_trg2	可能	可能	可能	可能
hrtim_dac_reset_trg3 hrtim_dac_step_trg3	可能	可能	可能	可能
hrtim_dac_reset_trg4 hrtim_dac_step_trg4	可能	可能	可能	可能
hrtim_dac_reset_trg5 hrtim_dac_step_trg5	可能	可能	可能	可能
hrtim_dac_reset_trg6 hrtim_dac_step_trg6	可能	可能	可能	可能

27.3.3 クロック

HRTIM は、フル分解能を提供するために、t_{HRTIM} APB2 クロックによって供給される必要があります。t_{HRTIM} クロック周期は、エッジ位置決めロジックを使用して最大 32 個の中間ステップに均等に分割されます。HRTIM にあるすべてのクロックはこのリファレンスクロックから導き出されます。

用語の定義

f_{HRTIM} : HRTIM メインクロック（hrtim_ker_ck）。以下に続くすべてのクロックはこのソースから導き出され、それに同期します。

f_{HRCK} : 高分解能等価クロック。f_{HRTIM} クロック周期を 32 で分周すると、170 x 32 = 5.44 GHz の周波数に相当します。

f_{DTG} : デッドタイムジェネレータクロック。便宜上、本書では t_{DTG} 周期（$t_{DTG} = 1/f_{DTG}$）のみを使用します。

f_{CHPFRQ}： チョッパステージのクロックソース。

f_{1STPW}： チョッパモードでの最初のパルスの長さを定義するクロックソース。便宜上、本書では t_{1STPW} 周期（$t_{1STPW} = 1/f_{1STPW}$）のみを使用します。

f_{BRST}： バーストモードコントローラカウンタクロック。

$f_{SAMPLING}$： フォールトまたは外部イベント入力をサンプリングするのに必要なクロック。

f_{FLTS}： f_{HRTIM} から導き出されるクロックで、フォールトイベントをフィルタリングするために $f_{SAMPLING}$ のソースとして使用されます。

f_{EEVS}： f_{HRTIM} から導き出されるクロックで、外部イベントをフィルタリングするために $f_{SAMPLING}$ のソースとして使用されます。

タイマクロックおよびプリスケーラ

HRTIM の各タイマはそれぞれ独自の個別クロックプリスケーラを備えており、タイマの分解能を調整することができます（表 214 参照）。

表 214. タイマ分解能および最小 PWM 周波数（f_{HRTIM} = 170 MHz の場合）

CKPSC[2:0]	プリスケーリング比	f_{HRCK} 等価周波数	分解能	最小 PWM 周波数
000	1	170 x 32 MHz = 5.44 GHz	184 ps	83.0 kHz
001	2	170 x 16 MHz = 2.72 GHz	368 ps	41.5 kHz
010	4	170 x 8 MHz = 1.36 GHz	735 ps	20.8 kHz
011	8	170 x 4 MHz = 680 MHz	1.47 ns	10.4 kHz
100	16	170 x 2 MHz = 340 MHz	2.94 ns	5.19 kHz
101	32	170 MHz	5.88 ns	2.59 kHz
110	64	170/2 MHz = 85 MHz	11.76 ns	1.30 kHz
111	128	170/4 MHz = 42.5 MHz	23.53 ns	0.65 kHz

高分解能は、エッジ位置決め、PWM 周期調整、および外部からトリガされるパルス幅に対して利用可能です。

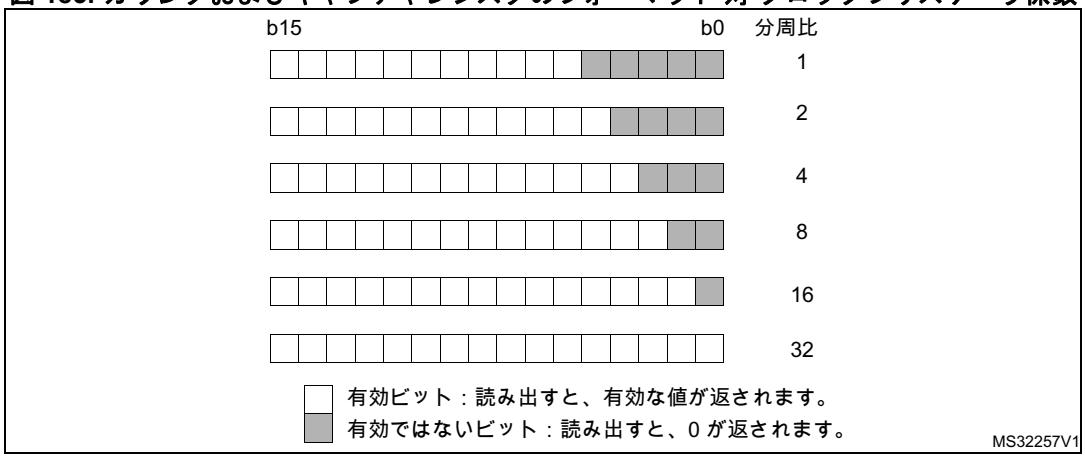
高分解能は次の機能では使用できません。

- タイマカウンタの読出し／書込みアクセス
- キャプチャユニット

32 を下回る（CKPSC[2:0] < 5）クロックプリスケーラ比に対しては、カウンタおよびキャプチャレジスタの最下位ビットは意味がありません。最下位ビットは書込みできず（カウンタレジスタのみ）、また、読み出すと 0 を返します。

たとえば、CKPSC[2:0] = 2（4 でプリスケーリング）の場合、カウンタ･レジスタに 0xFFFF を書き込むと、有効値として 0xFFF8 が生成されます。逆に、0xFFFF と 0xFFF8 の間のカウンタ値は 0xFFF8 として読み出されます。

図 183. カウンタおよびキャプチャレジスタのフォーマット 対 クロックプリスケーラ係数



初期化

起動時に、比較および周期レジスタを書き込む前に、最初にプリスケーラのビットフィールドを初期化することが必須です。タイマが有効化されると（HRTIM_MCR レジスタの MCEN または TxCEN のビット設定）、プリスケーラは変更できなくなります。

複数のタイマを有効化するときは、各プリスケーラは最初に起動したタイマのプリスケーラに同期します。

警告： **カウンタおよび出力動作が他のタイマの情報および信号に依存しない場合のみ、マスタタイマおよび TIMA～E タイマに異なるプリスケーリング比を与えることができます。次のイベントのうちの 1 つが 1 つのタイミングユニット（またはマスタタイマ）から別のタイミングユニットに伝播されるときは、これらのタイマでは同一のプリスケーリング比に設定することが必須です。そのイベントは、出力セット／リセットイベント、カウンタリセットイベント、更新イベント、外部イベントフィルタ、またはキャプチャトリガです。プリスケーラ分周比が異なると、予測できない結果になります。**

デッドタイムジェネレータクロック

デッドタイムプリスケーラは（f_{HRTIM} x 8）/ $2^{(DTPRSC[2:0])}$で与えられ、HRTIM_DTxR レジスタの DTPRSC[2:0] ビットでプログラムされます。

t_{DTG} は、f_{HRTIM} = 170 MHz に対して 735 ps～94.1 ns の範囲の値を取ります。

チョッパステージクロック

チョッパステージクロックのソース f_{CHPFRQ} は、16～256 の範囲の分周比を使って f_{HRTIM} から導き出されます。したがって、f_{HRTIM} = 170 MHz に対して、664.06 kHz <= f_{CHPFRQ} <= 10.625 MHz となります。

t_{1STPW} はチョッパモードでの最初のパルスの長さですが、HRTIM_CHPxR レジスタの STRPW[3:0] ビットで次のようにプログラムされます。

t_{1STPW} = (STRPW[3:0]+1) x 16 x t_{HRTIM}

クロックソースとして f_{HRTIM} / 16 を使用しています（f_{HRTIM}= 170 MHz に対して 10.625 MHz）。

バーストモードプリスケーラ

バーストモードコントローラのカウンタクロック f_{BRST} はいくつかのソースから供給されます。その中の 1 つは f_{HRTIM} から導き出されます。

この場合、f_{BRST} の範囲は f_{HRTIM}～f_{HRTIM} / 32768（f_{HRTIM} = 170 MHz に対して 5.188 kHz）となります。

フォールト入力サンプリングクロック

フォールト入力ノイズ低減フィルタは $f_{SAMPLING}$ （f_{HRTIM} または f_{FLTS} のどちらかになります）で定義される時定数を持ちます。

f_{FLTS} は f_{HRTIM} から導き出され、その範囲は f_{HRTIM} = 170 MHz に対して 170 MHz～21.25 MHz となります。

外部イベント入力サンプリングクロック

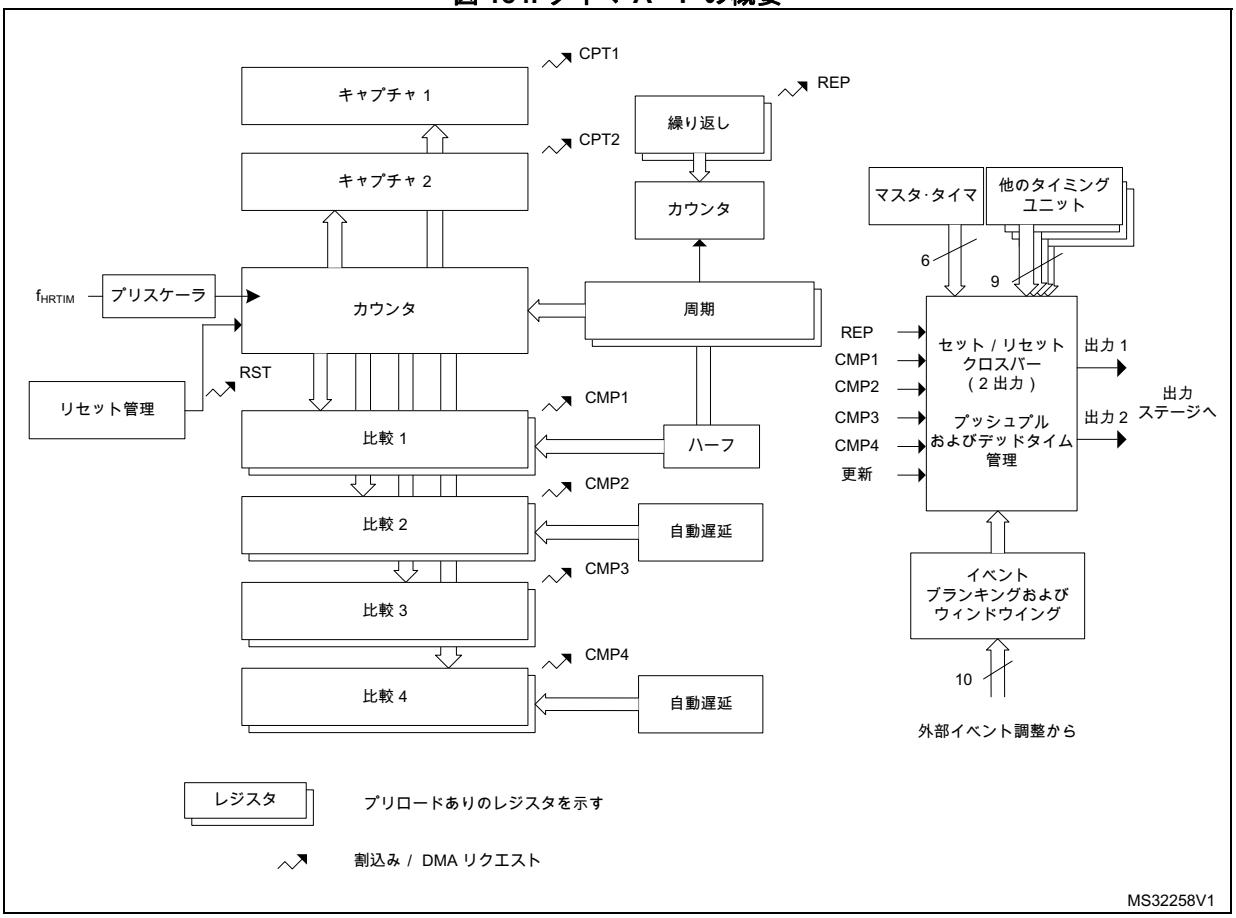
フォールト入力ノイズ低減フィルタは $f_{SAMPLING}$ （f_{HRTIM} または f_{EEVS} のどちらかになります）で定義される時定数を持ちます。

f_{EEVS} は f_{HRTIM} から導き出され、その範囲は f_{HRTIM} = 170 MHz に対して 170 MHz～21.25 MHz となります。

27.3.4 タイマ A～F のタイミングユニット

HRTIM は、図 184 のように、カウント周期を定義するための自動再ロードメカニズムを持つ 16 ビットのアップカウンタと、4 つの比較ユニット、2 つのキャプチャユニットから成る 6 つの同じタイミングユニットを内蔵しています。各ユニットは、独立したタイマとして動作するように、2 つの出力のためのすべての制御機能を装備しています。

図 184. タイマ A～F の概要



周期および比較値は高分解能の実装に関連する下限値および上限値以内でなければならず、表 215 に示すとおりです。

- 最小値は、f_{HRTIM} クロックの 3 周期以上である必要があります。PWM パルスをスキップするために、0x0000 の値は CMP1 と CMP3 のレジスタのみに書込み可能です。詳細は、セクション : ヌルデューティサイクルの例外ケースを参照してください。
- 最大値は、f_{HRTIM} クロックの 0xFFFF - 1 周期以下である必要があります。

表 215. 周期および比較レジスタの最小値および最大値

CKPSC[2:0] の値	最小値(1)	最大値
0	0x0060	0xFFDF
1	0x0030	0xFFEF
2	0x0018	0xFFF7
3	0x000C	0xFFFB
4	0x0006	0xFFFD
≥ 5	0x0003	0xFFFD

1. PWM パルスをスキップするために、0x0000 の値は CMP1 と CMP3 のレジスタのみに書込み可能です。詳細については、セクション ：ヌルデューティサイクルの例外ケースを参照してください。

注： **比較値が周期のレジスタ値より大きいと、比較一致イベントは発生しません。**

カウンタ動作モード

タイマ A～F は連続モード（フリーランニングモード）またはシングルショット方式で動作することができ、HRTIM_TIMxCR 制御レジスタの CONT ビットを使用して、リセットイベントによりカウントが始まります。追加の RETRIG ビットによって、シングルショット動作を再トリガ可能にするか再トリガ不可能にするかを選択できます。動作の詳細を表 216 および、図 185 と図 186 にまとめます。

表 216. タイマ動作モード

CONT	RETRIG	動作モード	開始／停止条件 クロック供給およびイベント生成
0	0	シングルショット 再トリガ不可能	TxEN ビットをセットすると、タイマが有効になりますが、カウンタは開始しません。 最初のリセットイベントによってカウントが始まり、カウンタが PER 値に達するまで後続のリセットは無視されます。 その後、PER イベントが生成され、カウンタは停止します。 リセットイベントがカウント動作を 0x0000 から再開します。
0	1	シングルショット 再トリガ可能	TxEN ビットをセットすると、タイマが有効になりますが、カウンタは開始しません。 カウンタが停止していれば、リセットイベントはカウントを開始し、そうでなければカウンタをクリアします。カウンタが PER の値に達すると、PER イベントが生成され、カウンタは停止します。 リセットイベントがカウント動作を 0x0000 から再開します。
1	X	連続モード	TxEN ビットをセットすると、タイマが有効になり、同時にカウンタが開始されます。 カウンタが PER の値に達すると、カウンタは 0x0000 に戻され、再開されます。 カウンタはいつでもリセットすることができます。

TxEN ビットはいつでもクリアでき、タイマを無効にしてカウントを停止させることができます。

図 185. 連続タイマ動作

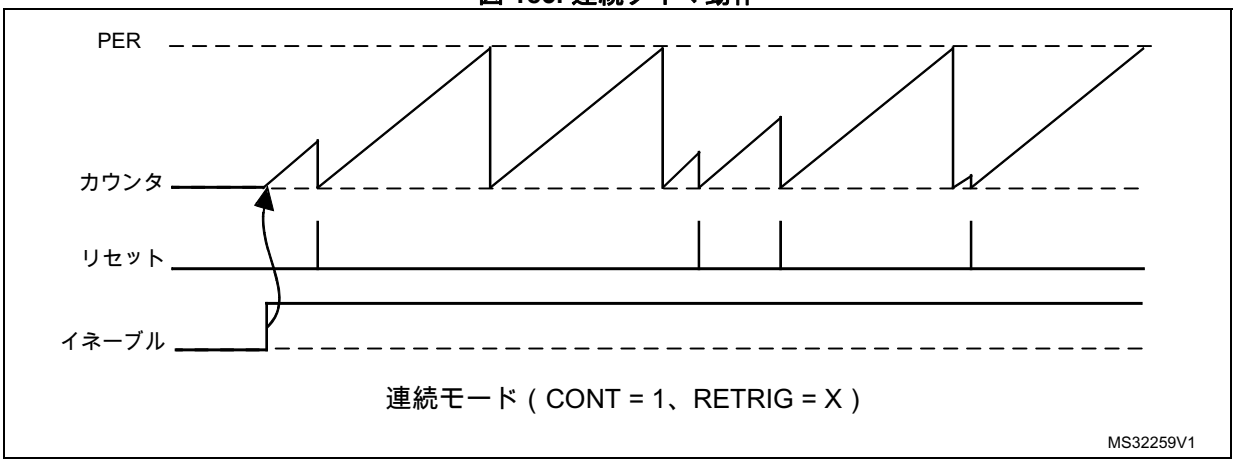
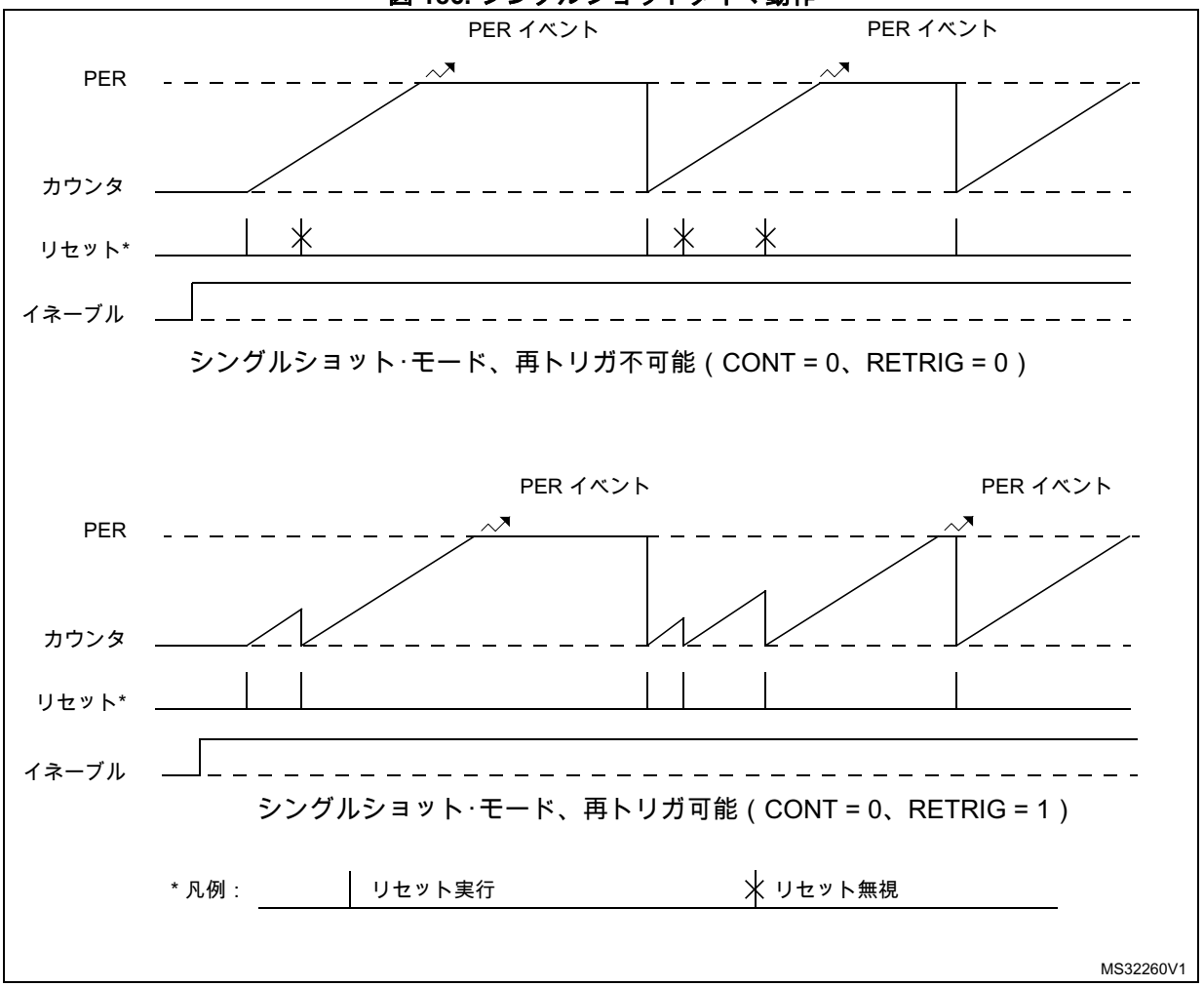


図 186. シングルショットタイマ動作



ロールオーバーイベント

連続モードにおいて、カウンタが HRTIM_PERxR レジスタで設定された周期値に達した後、0 に戻る時、カウンタのロールオーバーイベントが生成されます。

このイベントは、HRTIM において下記の複数の目的で使用されます。

- 出力をセット／リセットする
- レジスタの内容を更新するトリガとして（プリロードからアクティブへ転送）
- IRQ または DMA リクエストをトリガする
- バーストモードのクロックソースまたはバースト開始トリガとして働く
- ADC トリガとして
- 繰り返しカウンタをデクリメントする

タイマを開始する時、カウンタの初期値が周期の値より大きい場合、あるいは新しい周期が設定される場合にカウンタがすでにこの値より大きいときに、カウンタはリセットされず、周期の最大値でオーバーフローし、繰り返しカウンタはデクリメントしません。

タイマのリセット

タイミングユニットのカウンタのリセットは最大 30 個のイベントによってトリガできます。それらは以下のソースから HRTIM_RSTxR レジスタで同時に選択できます。

- タイミングユニット：比較 2、比較 4 および 更新（3 つのイベント）
- マスタタイマ：リセットおよび比較 1～4（5 つのイベント）
- 外部イベント EXTEVNT1～10（10 個のイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～F）：比較 1、2 および 4（12 個のイベント）

いくつかのイベントを同時に選択して複数のリセットソースを扱うことができます。この場合、複数のリセットリクエストは論理和がとられます。同じ f_{HRTIM} クロックサイクル内に 2 つのカウンタリセットイベントが発生した場合、最後のカウンタリセットが考慮されます。

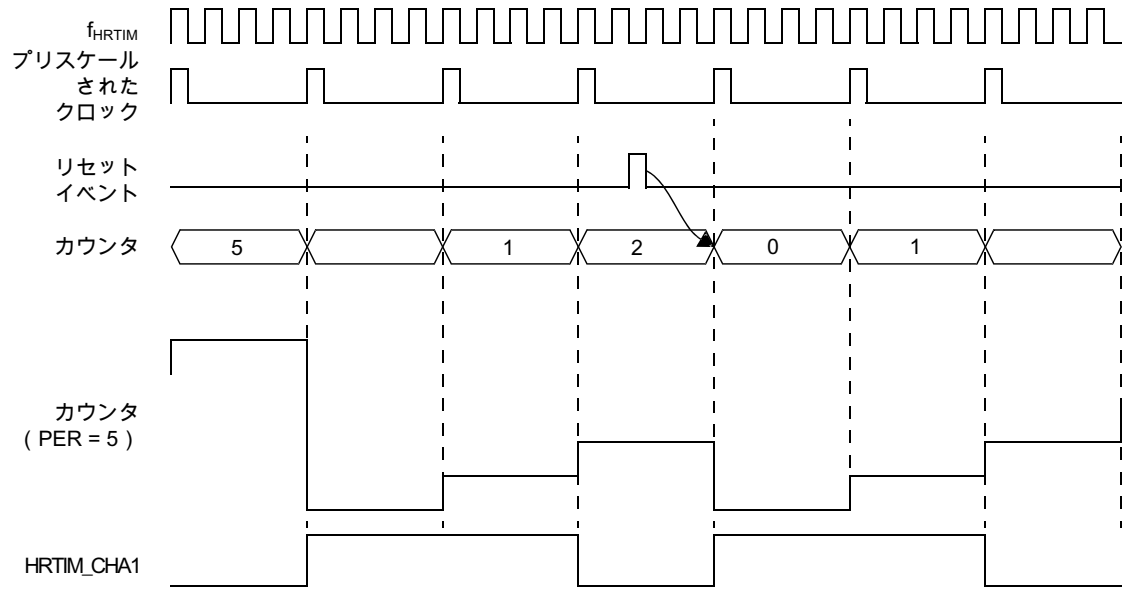
さらに、HRTIM_CR2 レジスタの TxRST ビットを使用して、カウンタのソフトウェアリセットをすることが可能です。これらの制御ビットは 1 つのレジスタ内にグループ化されているので、複数のカウンタを同時にリセットすることができます。

リセットリクエストは、関連カウンタが有効化（TxCEN ビットをセット）されてからのみ、考慮されます。

f_{HRTIM} クロックプリスケーリング比が 32 より大きい（カウント周期が f_{HRTIM} を上回る）とき、カウンタリセットイベントは、プリスケールされたクロックの次のアクティブエッジまで遅延されます。これによって、出力の遷移がリセットイベントに同期されているとき（コンスタント Ton 時間コンバータに典型的）、ジッタレス波形生成を維持することができます。

図 187 に、クロックのプリスケーリング比が 128 の（f_{HRTIM} を 4 で割った）場合に、リセットがどのように処理されるかを示します。

図 187. タイマリセットの再同期（32 を超えるプリスケーリング比）



HRTIM_CHA1：タイマ A のリセット·イベントでセット、比較 1 = 2 でリセット

MS32261V2

繰り返しカウンタ

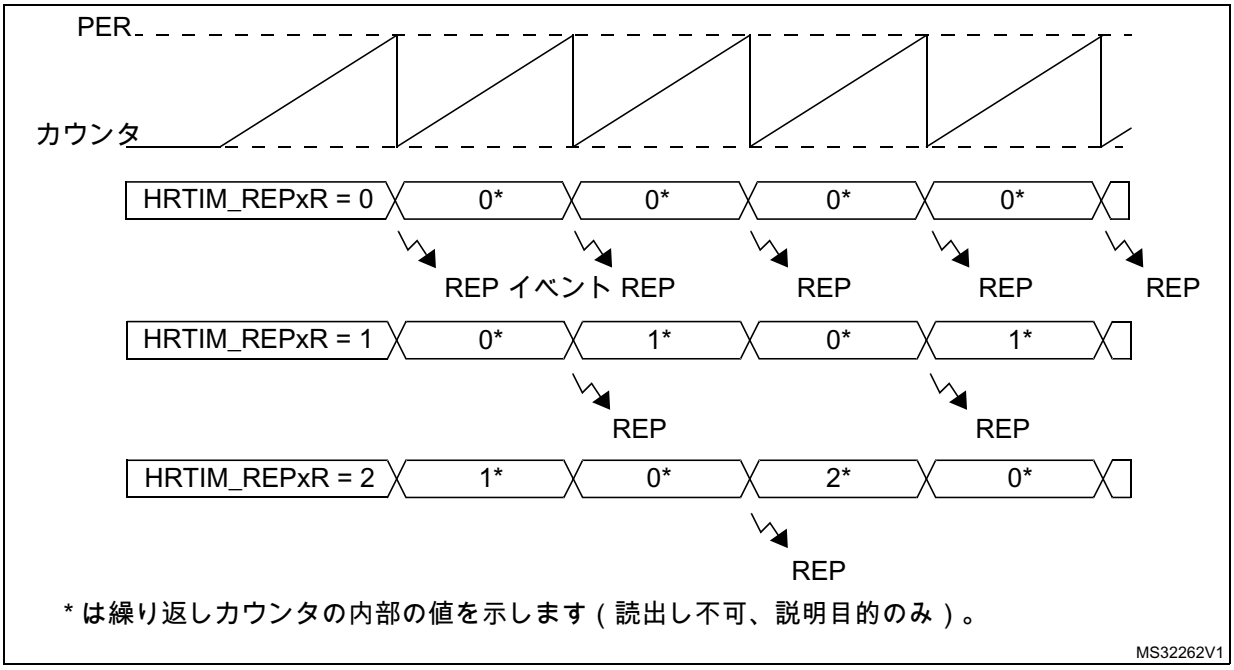
一般的なソフトウェアによる手法では、次の周期が始まる前に、処理時間の最大量が残るように、周期値に達したら割込みを発生させます。繰り返しカウンタの主な目的は、周期割込みレートを調整し、スイッチング周波数と割込み周波数を分離することによって CPU の負荷を削減することです。

タイミングユニットは繰り返しカウンタを備えています。このカウンタは読出しはできず、HRTIM_REPxR レジスタの自動再ロードの値でプログラムすることのみができます。

繰り返しカウンタは、タイマを有効にするとき（TXCEN ビットをセット）、HRTIM_REPxR レジスタの内容で初期化されます。いったんタイマを有効にすると、リセットイベントまたはカウンタのロールオーバーによってカウンタがクリアされる時に、繰り返しカウンタはデクリメントされます。カウンタがゼロに達すると、有効化されている場合（HRTIM_DIER レジスタの REPIE および REPDE ビット）、REP 割込みまたは DMA リクエストが発行されます。

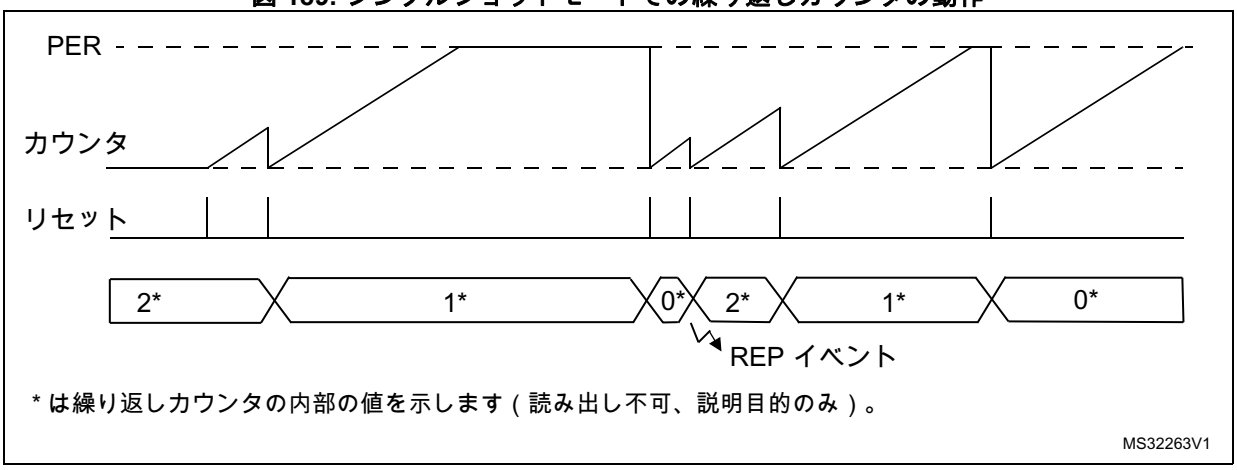
HRTIM_REPxR レジスタが 0 に設定されている場合、すべての周期ごとに割込みが発生します。0 より大きい値では、（HRTIM_REPxR + 1）周期 の後に REP 割込みが発生します。図 188 に、連続モードでの種々の値に対する繰り返しカウンタの動作を示します。

図 188. 連続モードでの繰り返しレート 対 HRTIM_REPxR の内容



連続モードまたはシングルショットモード（以下の図 189）のどちらにおいても、カウンタが周期値に達する前にリセットされるとき（可変周波数動作）にも繰り返しカウンタが使用できます。リセットすると、繰り返しカウンタはデクリメントされます。ただし、カウンタ有効化（TxCEN ビットのセット）の直後の、最初の開始は除きます。

図 189. シングルショットモードでの繰り返しカウンタの動作



HRTIM_SCIN[3:1] ソースからのリセットまたは開始イベントは、他のリセットと同じく、繰り返しカウンタのデクリメントを引き起こします。しかし、SYNCIN で開始されたシングルショットモード（HRTIM_TIMxCR レジスタの SYNCSTRTx ビットのセット）では、繰り返しカウンタは、周期の後の最初のリセットイベント時のみデクリメントされます。カウンタが HRTIM_SCIN[3:1] の入力による新しいリクエストによって再開されるまで、後続のリセットが来ても繰り返しカウンタは変更されません。

セット／リセットクロスバー

「セット」イベントは出力のアクティブ状態への遷移に対応し、「リセット」イベントは出力の非アクティブ状態への遷移に対応します。

波形の極性は、正または負論理の外部コンポーネントに適応させるために出力ステージで定義されます。アクティブレベルは、正の極性（POLx = 0）に対して論理レベル 1 に対応し、負の極性（POLx = 1）に対しては論理レベル 0 に対応します。

各タイミングユニットは 2 つの出力のためにセット／リセットクロスバーを操作します。これらの 2 つの出力は、以下のソースから選択できる最大 32 個のイベントによってセット、リセット、またはトグル反転ができます。

- タイミングユニット：周期、比較 1～4、レジスタ更新（6 つのイベント）
- マスタタイマ：周期、比較 1～4、HRTIM 同期（6 つのイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～F）：TIMEVNT1～9（表 217 に記載された 9 つのイベント）
- 外部イベント EXTEVNT1～10（10 個のイベント）
- ソフトウェアの強制（1 つのイベント）

注： **アップ／ダウンモード（UDM ビットを 1 に設定）では、カウンタ周期イベントは OUTROM [1:0] ビットの設定に従って定義されます。**

イベントソースは論理和がとられ、複数のイベントが同時に選択できます。

各出力は 2 つの 32 ビット･レジスタによって制御されます。1 つはセット用（HRTIM_SETxyR）で、もう 1 つはリセット用（HRTIM_RSTxyR）です。ここで、x はタイミング･ユニット：A ~ F を表し、y は出力 1 または 2 を表します（たとえば、HRTIM_SETA1R、HRTIM_RSTC2R、など）。

同じイベントがセット用とリセット用の両方に選択された場合は、出力をトグル反転させます。出力状態をトグル反転することは t_{HRTIM} 周期あたり 1 回しかできません。同じサイクル内で 2 つの連続するトグル反転イベントが起きた場合、最初のイベントのみ考慮されます。

セットおよびリセットリクエストは、カウンタが有効にされて（TxCEN ビットをセット）からのみ考慮されます。ただし、ソフトウェアがタイマ起動時に出力の事前設定を許すリクエストを強制している場合を除きます。

表 217 に、出力をセット・リセットするために使用できる、他のタイミングユニットからのイベントをまとめます。数字はレジスタに挙がっているタイマイベント（TIMEVNTx など）に対応し、空の場所は利用できないイベントを示します。

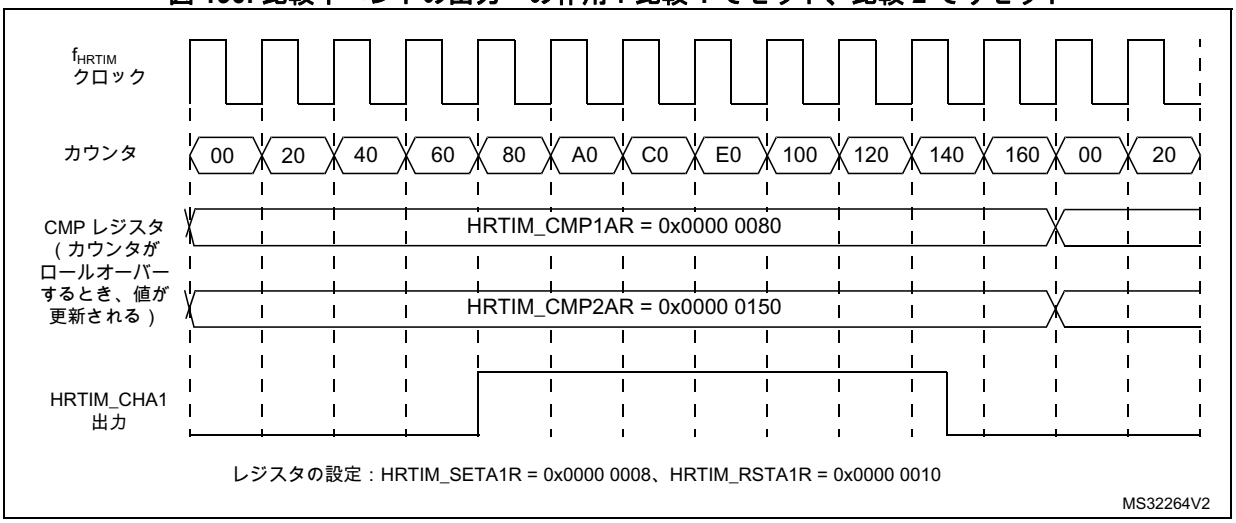
たとえば、タイマ A の出力は次のイベントによってセットまたはリセットできます。タイマ B の比較 1 および 2、タイマ C の比較 2 および 3、・・・なお、タイマ E の比較 3 は HRTIM_SETA1R で TIMEVNT7 としてリストされています。

表 217. タイマ A～F におけるイベントマッピング

転送元		タイマ A				タイマ B				タイマ C				タイマ D				タイマ E				タイマ F			
		CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4	CMP1	CMP2	CMP3	CMP4
転送先	TA	-	-	-	-	1	2	-	-	-	3	4	-	5	6	-	-	-	-	7	8	-	-	-	9
	TB	1	2	-	-	-	-	-	-	-	-	3	4	-	-	5	6	7	8	-	-	-	-	9	-
	TC	-	1	2	-	-	3	4	-	-	-	-	-	-	5	-	6	-	-	7	8	-	9	-	-
	TD	1	-	-	2	-	3	-	4	-	-	-	5	-	-	-	-	6	-	-	7	8	-	9	-
	TE	-	-	-	1	-	-	2	3	4	5	-	-	6	7	-	-	-	-	-	-	-	-	8	9
	TF	-	-	1	-	2	-	-	3	4	-	-	5	-	-	6	7	-	8	9	-	-	-	-	-

図 190 は 2 つの比較イベントを使用して PWM を生成する方法を表します。

図 190. 比較イベントの出力への作用：比較 1 でセット、比較 2 でリセット



更新イベントによるセット／リセット

更新によるセットまたはリセットイベントは低分解能で行われます。CKPSC[2 : 0] < 5 の場合、高分解能の遅延は最大値に設定されるため、更新時のセット/リセットイベントは、他の比較セット/リセットイベントと比較して常に遅れ、ジッターは f_{HRTIM} クロックの 0～31/32 周期の間で変化します。

ハーフモード

このモードは、50% 固定デューティサイクルで可変周波数の矩形波信号（共振接続形態を使用したコンバータに典型的）の生成を目的にしています。新しい周期がプログラムされるとき、デューティサイクルを自動的に周期値の半分にすることができます。

このモードは、HRTIM_TIMxCR レジスタの HALF ビット に 1 を書き込むことによって有効になります。HRTIM_PERxR レジスタに値が書き込まれると、比較 1 の値が自動的に HRTIM_PERxR/2 の値に更新されます。

矩形波が生成される出力は、以下のように、CMP1 イベントで 1 回の遷移が起き、周期イベントで 1 回の遷移が起きるようにプログラムする必要があります。

- HRTIM_SETxyR = 0x0000 0008、HRTIM_RSTxyR = 0x0000 0004、または
- HRTIM_SETxyR = 0x0000 0004、HRTIM_RSTxyR = 0x0000 0008

ハーフモードでは、HRTIM_CMP1xR レジスタの内容が無効にされます。HRTIM_PERxR レジスタへのアクセスによってのみ、比較 1 の内部レジスタが更新されます。ユーザアクセス可能な HRTIM_CMP1xR レジスタは HRTIM_PERxR / 2 の値で更新はされません。

プリロードが有効になっている場合（PREEN = 1、MUDIS、TxUDIS）、比較 1 アクティブレジスタは更新イベントによりリフレッシュされます。プリロードが無効になっている場合（PREEN = 0）、比較 1 アクティブレジスタは HRTIM_PERxR が書き込まれるとすぐに更新されます。

HALF モードが有効になっているとき、周期は f_{HRTIM} クロックの 6 周期（CKPSC[2:0] = 0 の場合は 0xC0、CKPSC[2:0] = 1 の場合は 0x60、CKPSC[2:0] = 2 の場合は 0x30、・・・）以上でなければなりません。

インタリーブモード

このモードは、ハーフモードを補完し、インタリーブ接続形態の実装に役立ちます。

これにより、HRTIM_PERxR の値が更新されると、比較レジスタの内容を自動的に再計算できます。

その選択は、以下の表 218 に示すように、HRTIM_MCR および HRTIM_TIMxCR レジスタの HALF ビットと INTLVD[1:0] ビットを使用して行います。

表 218. インタリーブモードの選択

HALF ビット	INTLVD [1:0] ビット	モード
0	00	無効
0	01	トリプルインタリーブ（120°）
0	10	クワッドインタリーブ（90°）
0	11	予約済みです。
1	xx	デュアルインタリーブ（180°）

表 219 に、3 つの使用可能なモードの比較値を示します。比較レジスタの内容は上書きされます。対応する比較イベントを使用して、出力セット/リセットをトリガすることや、スレーブタイマをリセットすることができます。

表 219. インタリーブモードでの比較 1～3 の値

モード	デュアルインタリーブ 180°	トリプルインタリーブ 120°	クワッドインタリーブ 90°
CMP1 値	PERxR/2	PERxR/3	PERxR/4
CMP2 値	影響されません	2x（PERxR/3）	PERxR/2
CMP3 値	影響されません	影響されません	3x（PERxR/4）

注： **ハーフおよびインタリーブモードでは、比較レジスタはハードウェアによって制御され、それらへ書き込んでも効果はありません。ただし、書き込まれた値はプリロードレジスタに保存され、これらのモードの終了後の更新イベントでアクティブになります。**

注： **トリプルおよびクワッドインタリーブモードは、CMP2 を使用する他のモード（デュアルチャネル DAC トリガおよびトリガハーフモード）と同時に使用しないでください。**

ヌルデューティサイクルの例外ケース

高分解能の動作は、3 t_{HRTIM} 周期より狭いパルスではサポートされておらず（セクション 27.3.7：セット／リセットイベントの優先順位と狭いパルスの管理を参照）、HRTIM_TIMxCMPy レジスタでは f_{HRTIM} クロックの 3 周期（すなわち、CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）より厳密に小さい値は禁止されています（27.5.19：HRTIM タイマ x 比較 1 レジスタ（HRTIM_CMP1xR）（x = A～F）を参照）。

ただし、次の条件が満たされている場合にのみ、HRTIM_TIMxCMP1 および HRTIM_TIMxCMP3 の 2 つのレジスタにヌル値を書き込むだけで、出力パルスをスキップしてデューティ･サイクルをゼロにすることができます。

- 出力 SET イベントは、PERIOD イベントによって生成されます。
- 出力 RESET は比較 1（あるいは比較 3）イベントによって生成されます。
- 比較 1（比較 3）イベントは、自身のタイマユニット内でアクティブであり、他のタイミングユニットには使用されません。

その他の使用事例の場合、これは、f_{HRTIM} クロックの 3 周期を超える、まったく同じ比較値を使用して SET イベントと RESET イベントをプログラミングすることで実行できます。この場合、出力は強制的にリセットされます（セクション 27.3.7：セット／リセットイベントの優先順位と狭いパルスの管理に定義したセット／リセット優先度スキームに従って）。

スワップモード

このモードでは、単一のビットアクセスで 2 つの出力を交換できます。出力 1 信号は出力 2 ピンに接続され、出力 2 信号は出力 1 ピンに接続されます。出力スワップは、HRTIM_CR2 レジスタの SWPx ビットでトリガされ、次の更新イベントで有効になります。

出力は、次のように、セット/リセットクロスバーユニットの前で交換されます。

- SWPx = 0 の場合、HRTIM_SETx1R と HRTIM_RSTx1R は出力 1 用であり、HRTIM_SETx2R と HRTIM_RSTx2R は出力 2 用です。
- SWPx = 1 の場合、HRTIM_SETx1R と HRTIM_RSTx1R は出力 2 用であり、HRTIM_SETx2R と HRTIM_RSTx2R は出力 1 用です。

スワップモードはプリロードレジスタにのみ影響し、アクティブレジスタには影響しません。

注： **スワップモードを使用する場合は、プリロードモードを有効にする必要があります。**

したがって、出力ステージに行く通常の出力と並行した補助出力は変更されません（詳細については、セクション 27.3.18 : を参照）。それらは以下の内部ステータス、イベント、および信号を提供します。

- O1CPY、O2CPY、SETxy および RSTxy ステータスフラグ、ならびに、対応する割込みおよび DMA リクエスト
- 出力セット／リセット時のキャプチャトリガ（TA2、TB2、TC2、TD2、TE2、TF2）
- Tx2 出力コピーによって生成される外部イベントフィルタ

たとえば、SETx1 フラグは、SWP = 0 の場合は出力 1 に関連し、SWPx = 1 の場合は出力 2 に関連します。

同様に、スワップモードは、HRTIM_OUTxR レジスタの制御ビット（DIDLx、CHPx、FAULTx[1:0]、IDLESx、POLx ビット）の属性を変更しません。たとえば、POL1 ビットは、SWP ビットの値に関係なく、出力 1 の極性を制御します。

注 : **SWPx ビットは、プッシュプルモード（HRTIM_TIMxCR レジスタの PSHPLL = 1）では無視されます。**

キャプチャ

タイミングユニットには、内部および外部イベントをトリガにしてカウンタ値を取得する機能があります。用途は、次のとおりです。

- イベント到着のタイミングや発生間隔を測定する。
- 自動遅延モードにおいて比較 2 と比較 4 の値を更新する（セクション : 自動遅延モードを参照）。

キャプチャは f_{HRTIM} 分解能で行われます。32 を下回る（CKPSC[2:0] < 5）クロックプリスケーラ比に対しては、レジスタの最下位ビットは意味がありません（0 として読み出されます）。

タイマには次の 2 つのキャプチャレジスタがあります。HRTIM_CPT1xR および HRTIM_CPT2xR。キャプチャトリガは HRTIM_CPT1xCR および HRTIM_CPT2xCR レジスタでプログラムされます。

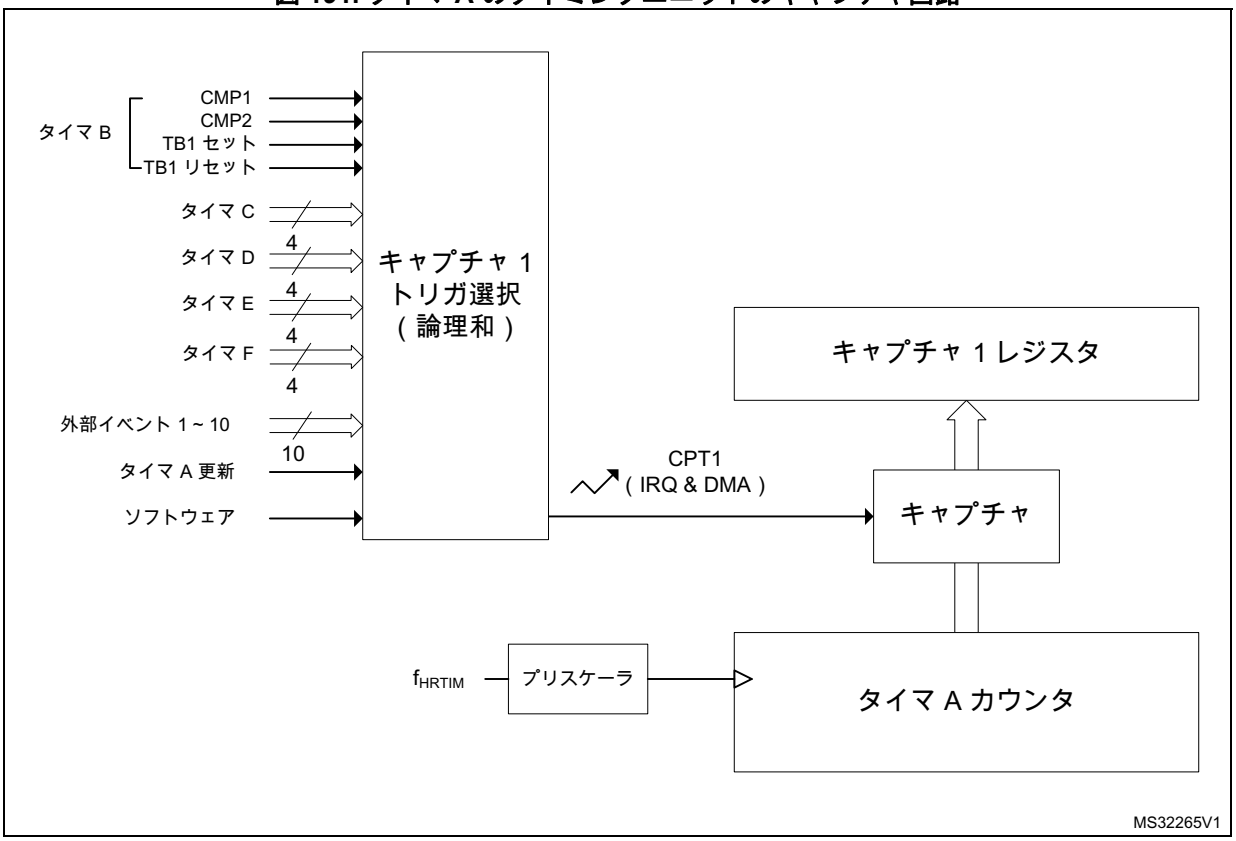
タイミングユニットのカウンタのキャプチャは最大 28 個のイベントによってトリガできます。それらは以下のソースから HRTIM_CPT1xCR および HRTIM_CPT2xCR レジスタで同時に選択できます。

- 外部イベント、EXTEVNT1～10（10 個のイベント）
- その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～F）: 比較 1、2 および出力 1 セット／リセットイベント（16 個のイベント）
- タイミングユニット : 更新（1 つのイベント）
- ソフトウェアキャプチャ（1 つのイベント）

いくつかのイベントを同時に選択して複数のキャプチャトリガを処理することができます。この場合、同時のトリガリクエストは論理和がとられます。HRTIM_TIMxDIER レジスタに CPTxIE および CPTxDE ビットがセットされているとき、キャプチャによって割込みまたは DMA リクエストを生成することができます。

オーバーキャプチャは回路では防げません。前の値が読み出されていなくても、あるいはキャプチャフラグがクリアされていなくても、新しいキャプチャはトリガされます。

図 191. タイマ A のタイミングユニットのキャプチャ回路



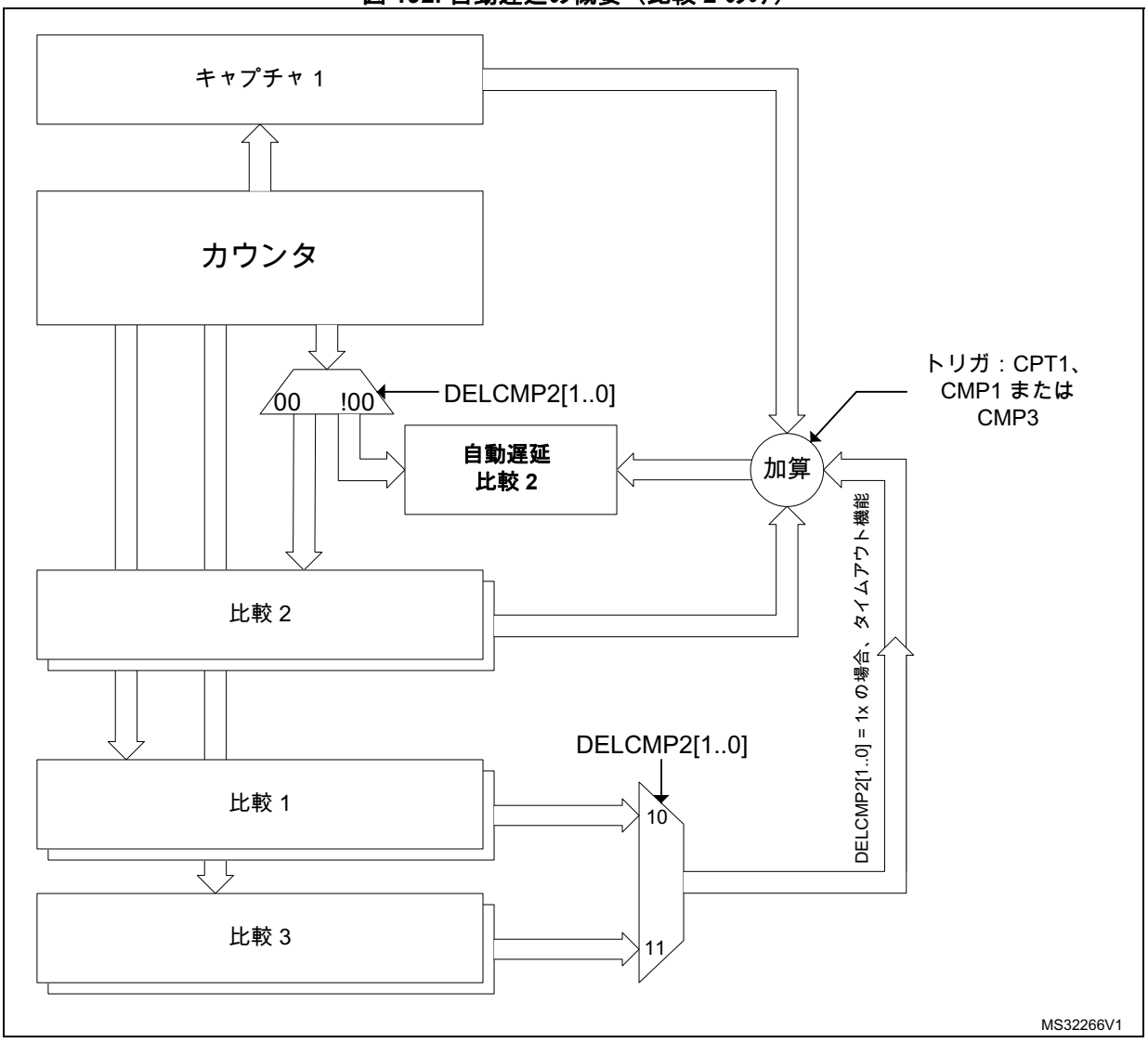
自動遅延モード

このモードではキャプチャイベントに相対的に比較イベントを生成させることができるので、たとえば、キャプチャの後にプログラムされたタイミングで出力を変化させることができます。この場合、比較一致はタイマのカウンタ値とは独立に起きます。ソフトウェアでの計算や割込み処理を必要とせずに、外部イベントに同期したタイミングで波形を生成することが可能になります。

キャプチャがトリガされない限り、HRTIM_CMPxR レジスタの内容は無視されます（カウンタ値が比較値に一致しても、比較イベントは生成されません）。キャプチャがトリガされされると、HRTIM_CMPxR にプログラムされた比較値が HRTIM_CPTxyR のキャプチャされたカウンタ値と合計され、図 192 に見られるように、内部自動遅延比較レジスタを更新します。自動遅延比較レジスタはタイミングユニットの内部にあり、読出しはできません。HRTIM_CMPxR プリロード･レジスタは、計算の後、変更されません。

この機能は、比較 2 および比較 4 レジスタでのみ使用できます。比較 2 はキャプチャ 1 に関連付けられ、比較 4 はキャプチャ 2 に関連付けられています。通常モードと同様に、HRTIM_CMP2xR および HRTIM_CMP4xR の比較は、f_{HRTIM} クロックの 3 周期を下回る値でプログラムすることはできません。

図 192. 自動遅延の概要（比較 2 のみ）



自動遅延比較は、キャプチャから周期イベントまでの間のみ有効です。カウンタが周期値に達したら、キャプチャが発生するまで、システムは比較を無効にして再び動作可能になります。

以下のように、HRTIM_TIMxCR レジスタの DELCMP2[1:0] および DELCMP4[1:0] ビットによって、自動遅延モードを設定できます。

- 00
  通常比較モード：HRTIM_CMP2xR および HRTIM_CMP4xR レジスタの内容はカウンタ値と直接比較されます。
- 01
  自動遅延モード：比較 2 および比較 4 の値は、キャプチャ 1/2 イベントの後、再計算されてカウンタとの比較に使用されます。
- 10 または 11
  タイムアウト付き自動遅延モード：比較 2 および比較 4 の値は、キャプチャ 1/2 イベントの後、またはキャプチャ 1/2 イベントが失われた場合にタイムアウト機能が働くように、比較 1 が一致（DELCMPx[1:0]= 10）、または比較 3 が一致（DELCMPx[1:0]= 11）した後に、再計算されてカウンタとの比較に使用されます。

キャプチャが起きると、比較は （HRTIM_CMP2/4xR + HRTIM_CPT1/2xR）の値で行われます。周期内でキャプチャがトリガされない場合、動作は以下のように DELCMPx[1:0] の値に依存します。

- DELCMPx[1:0] = 01：比較イベントは生成されません。
- DELCMPx[1:0] = 10 または 11 : 2 つの比較の合計（たとえば、HRTIM_CMP2xR + HRTIM_CMP1xR）で比較が行われます。キャプチャが CMPx + CMP1（または CMPx + CMP3）の後にトリガされた場合、そのキャプチャは考慮されません。

キャプチャは、次の PWM 周期の始めに再び有効化されます。

自動遅延の合計の結果が 0xFFFF を超えた場合（オーバーフロー）、その値は無視され、新しい周期が始まるまで比較イベントは生成されません。

注： **DELCMPx[1:0] ビットフィールドをある値から別の値に再プログラムするときは、自動遅延メカニズムを適切に再初期化するために、そのビットフィールドをリセットする必要があります。たとえば以下のとおりです。**

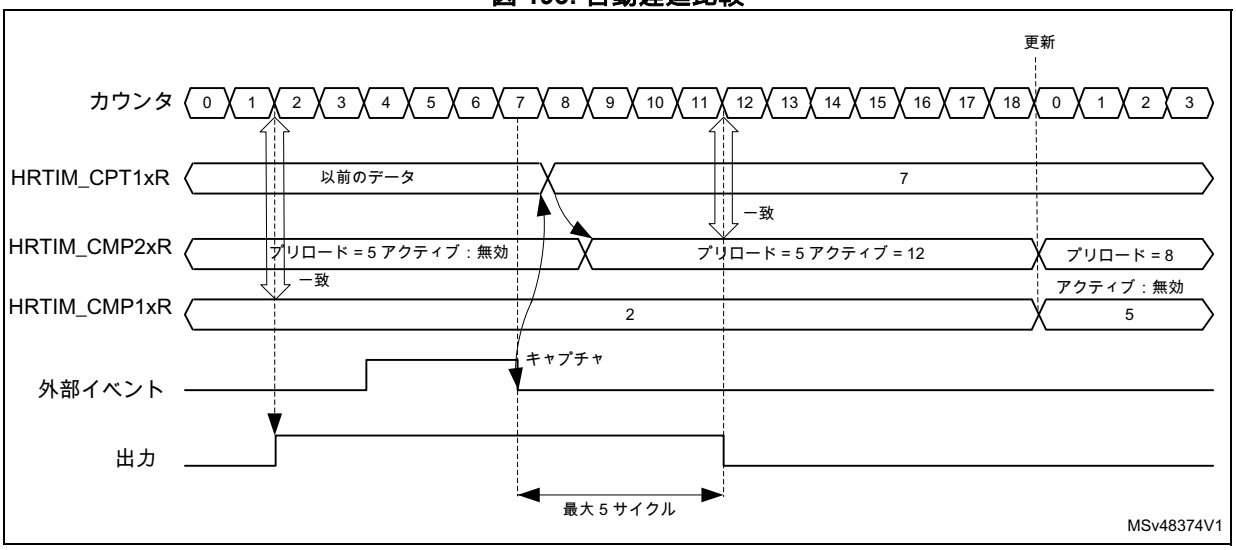
- DELCMPx[1:0] = 10
- DELCMPx[1:0] = 00
- DELCMPx[1:0] = 11

例として、次の信号がどのようにして生成されるかを図 193 に示します。

- カウンタが比較 1 の値に等しい時に出力をセット
- 与えられた外部イベントの立ち下がりエッジの 5 サイクル後に出力をリセット

注： **図を単純化するために、この例では高分解能は使用されていません（CKPSC [2:0] = 101）。したがって、カウンタは f_{HRTIM} レートでインクリメントされます。同様に、外部イベント信号は再同期遅延なしで表示されています。実際は、外部入力信号を処理するのに必要な内部再同期ステージのため、立ち下がりエッジとキャプチャイベントの間に f_{HRTIM} クロックの 1～2 周期の遅れがあります。**

図 193. 自動遅延比較



通常の比較チャネル（たとえば、比較 1）は出力のセットに使用されます。カウンタが比較レジスタの内容と一致すると、すぐに出力はアクティブ状態に移行します。

遅延比較は出力のリセットに使用されます。キャプチャイベントが発生した場合のみ、比較イベントが生成されます。カウンタが遅延比較値に一致するとき（カウンタ = 4）、イベントは生成されません。キャプチャイベントが外部イベントによってトリガされると、キャプチャレジスタの内容が遅延比較値に合計され、新しい比較値になります。例では、自動遅延値 4 が、7 に等しいキャプチャ値に

合計され、自動遅延比較レジスタに 12 の値を与えます。これ以降、比較イベントは生成可能となり、カウンタが 12 に等しくなる時に生成され、出力をリセットします。

自動遅延モードにおけるオーバーキャプチャ管理

自動遅延モードが有効になっているとき（DELCMPx[1:0] = 01、10、11）、オーバーキャプチャは防止されます。

複数のキャプチャリクエストが同じカウント周期内で発生したとき、最初のキャプチャのみが考慮されて、自動遅延比較値が計算されます。新しいキャプチャは以下のときのみ、可能となります。

- 自動遅延比較がカウンタ値と一致した（比較イベント）後
- カウンタがロールオーバーした（周期）後
- タイマがリセットされた後

自動遅延比較値の変更

自動遅延比較値がプリロードされると（PREEN ビットのセット）、新しい比較値は、次に来る更新イベント時（たとえば、周期イベント時）に考慮されます。これは、比較レジスタがいつ書き込まれたか、およびキャプチャが発生したか否かに関わりません（図 193 を参照ください。そこでは、カウンタがロールオーバーする時に、遅延が変更されています）。

プリロードが無効になっているときは（PREEN ビットのリセット）、たとえキャプチャイベントが発生した後に変更されても、新しい比較値は直ちに考慮されます。その例を以下に示します。

1. t1 で、DELCMP2 = 1
2. t2 で、CMP2_act = 0x40 => 比較は無効化
3. t3 で、キャプチャイベントが発生し、CPTR1 = 0x20 の値を取得=> 比較は有効化、比較値 = 0x60
4. t4 で、CMP2_act = 0x100（カウンタが CPTR1 + 0x40 の値に達する前に）=> 比較はまだ有効、新しい比較値 = 0x120
5. t5 で、カウンタが周期値に達する => 比較は無効化、cmp2_act = 0x100

同様に、DELCMPx = 10 または 11 で、プリロードが無効のとき、CMP1（CMP3）の値が変化した場合は、以下のとおり。

1. t1 で、DELCMP2 = 2
2. t2 で、CMP2_act = 0x40 => 比較は無効化
3. t3 で、キャプチャ 1 イベントが発生する前に、CMP3 イベントが発生 - CMP3_act = 0x50 => 比較は有効化、比較値 = 0x90
4. t4 で、CMP3_act = 0x100（カウンタが 0x90 の値に達する前に）=> 比較はまだ有効、比較 2 のイベントが = 0x140 で発生

トリガハーフモード

このモードの目的は、可変周波数動作で 180°の位相シフトを必要とする 2 つのインタリーブコンバータの同期を可能にすることです。基本的な原理は、マスタ／スレーブシステムを使用することです。スレーブコンバータの同期は、マスタコンバータの以前のスイッチング周期に基づいて継続的に調整されます。

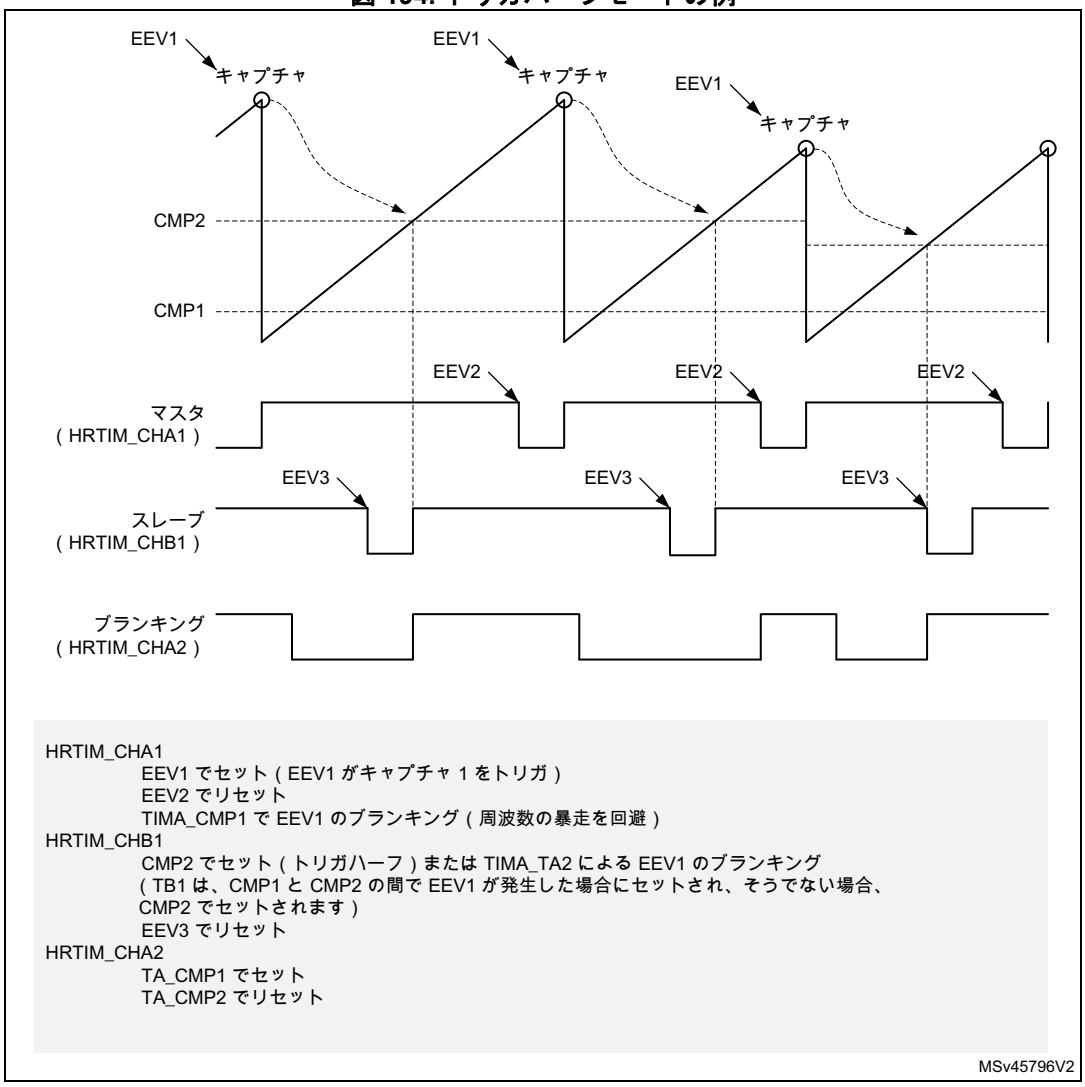
これはキャプチャユニットを使用して行います。マスタコンバータのスイッチング周期がキャプチャされ、2 で除算され、ハードウェアによって比較 2 レジスタに格納されます。比較 2 レジスタには、キャプチャされた周期（これは、マスタコンバータのスイッチング周期です）の半分に相当する値が含まれます。その後、比較 2 イベントを使用して、スレーブコンバータを管理する 2 番目のタイミングユニットをトリガできます。

このモードを有効にするには、HRTIM_TIMxCR2 レジスタの TRGHLF ビットをセットします。このビットは、タイマが動作を開始（TxEN ビットのセット）した後は変更できません。

トリガハーフモードは、CMP2 を使用する他のモード（デュアルチャネル DAC トリガ、インタリーブおよびバランスアイドルモード）と同時に使用しないでください。

CMP2 の初期値はユーザが書き込むことができますが、最初のキャプチャがトリガされると無視されます。TRGHLF ビットがリセットされているとき、CMP2 ではプリロード･メカニズムは無効になります。

図 194. トリガハーフモードの例



注： **トリガハーフモードでは、比較 2 レジスタはハードウェアによって制御され、それへ書き込んでも効果はありません。ただし、書き込まれた値はプリロードレジスタに保存され、このモードの終了後の更新イベントでアクティブになります。**

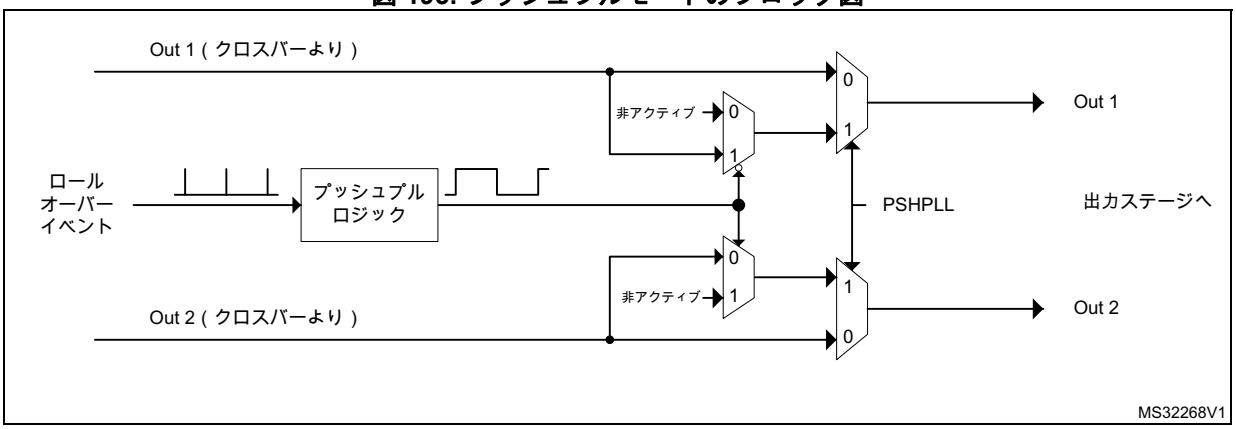
プッシュプルモード

このモードは主にプッシュプル接続形態を使用してコンバータを駆動することを目的にしています。共振コンバータが典型的ですが、遅延アイドル保護が必要なとき、これを有効にする必要があります（セクション 27.3.10：遅延保護を参照）。

プッシュプルモードは、HRTIM_TIMxCR レジスタの PSHPLL ビットをセットすることによって有効になります。

クロスバーで生成された信号を、周期ベースで、出力 1 と出力 2 に交互に適用し、適用してない他方の出力は非アクティブに保ちます。リダイレクトレート（プッシュプル周波数）は、図 195 に示すように、タイマの周期イベントによって定義されます。プッシュプル周期はタイマのカウント周期の 2 倍です。

図 195. プッシュプルモードのブロック図



プッシュプルモードは、タイマが連続モードおよびシングルショットモードで動作しているときに、利用できます。プッシュプル動作を停止するためには、タイマを無効にする必要があり、それを再び有効にする前にはカウンタをリセットする必要があります。

正しい動作を得るには、カウンタリセットのソースとして選択したイベントは、出力のセット（またはリセット）のためにも選択する必要があります。出力をオン期間に設定する場合、イベントによって出力をセットする必要があり、そうでなければリセットする必要があります。そうしないと、非アクティブ期間からアクティブ期間への出力切り替えが不正になることがあります（予期せずに立ち上がるか、予期せずに低いままになる可能性があります）。

信号波形は、両出力に対して HRTIM_SETxyR および HRTIM_RSTxyR を使用して定義します。HRTIM_SETx1R = HRTIM_SETx2R および HRTIM_RSTx1R = HRTIM_RSTx2R として、両出力を同じ波形とし、バランスのとれた動作にする必要があります。それでも、他の用途では、両出力に異なるプログラミングをすることができます。

HRTIM_TIMxISR の CPPSAT ステータスビットは、信号が現在、どちらの出力にアクティブなのかを示します。プッシュプルモードが無効になると、CPPSTAT はリセットされます。

図 196 の例では、タイマの内部波形は次のように定義されています。

- 周期イベントで出力セット
- 比較 1 の一致イベントで出力リセット

図 196. プッシュプルモードの例

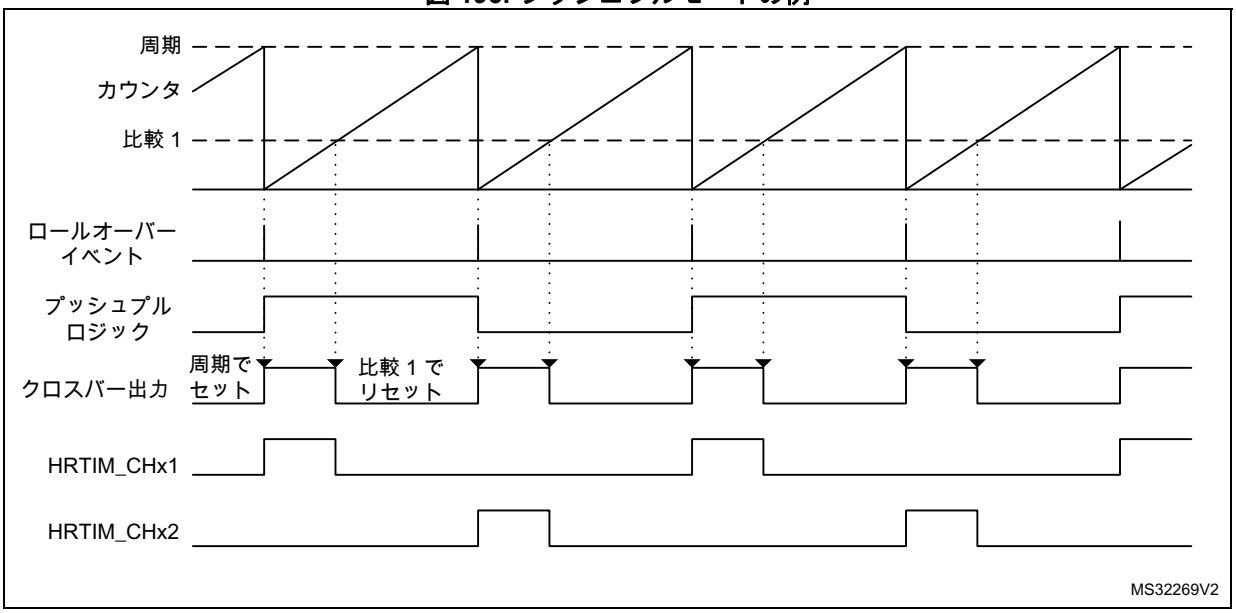
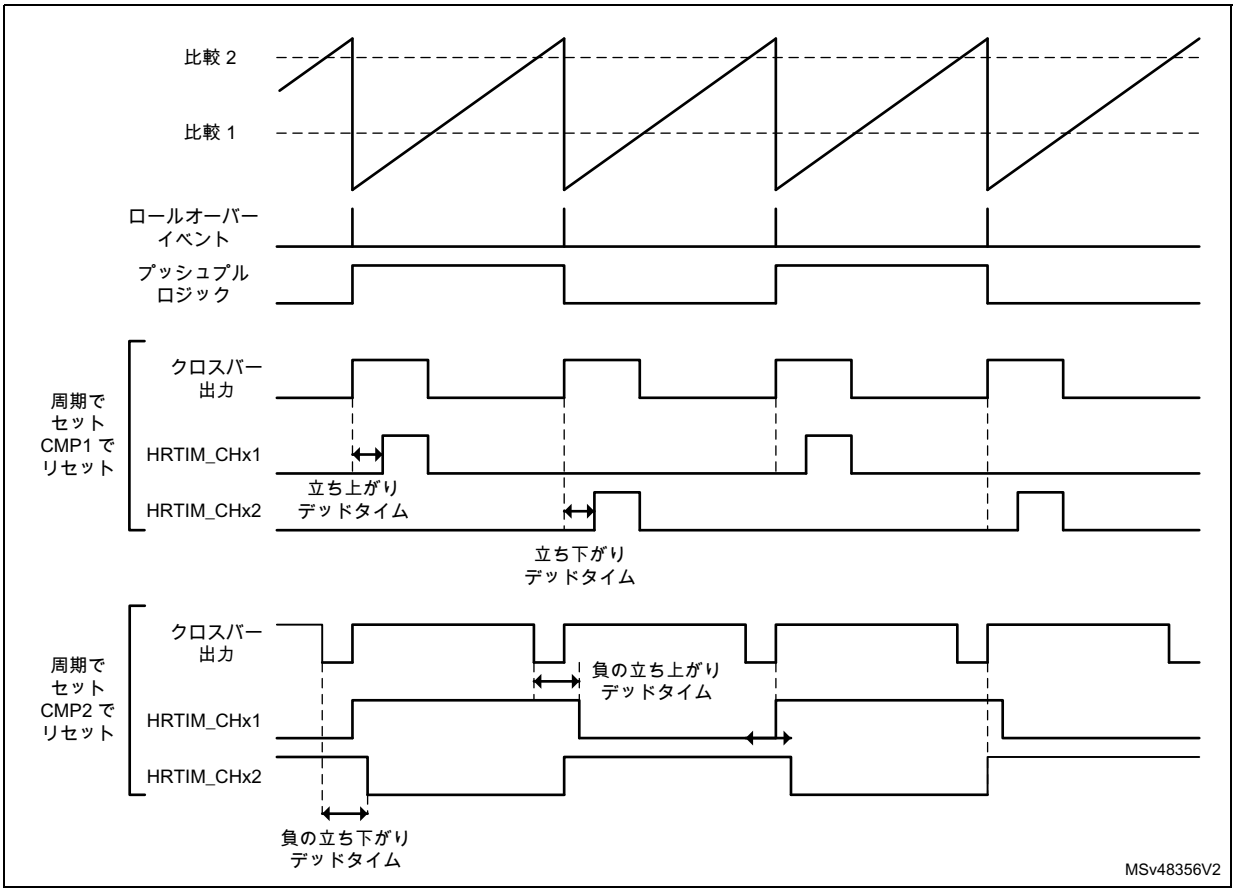


図 197 に、デッドタイムがプッシュプルモードでどのように挿入されるかを、正と負の両方のデッドタイムについて示します。この場合、出力は以降相補されず、デッドタイムは各出力に個別に適用されます（クロスバーの出力 1 と出力 2 の両方が使用されます）。

図 197. デッドタイム付きのプッシュプル



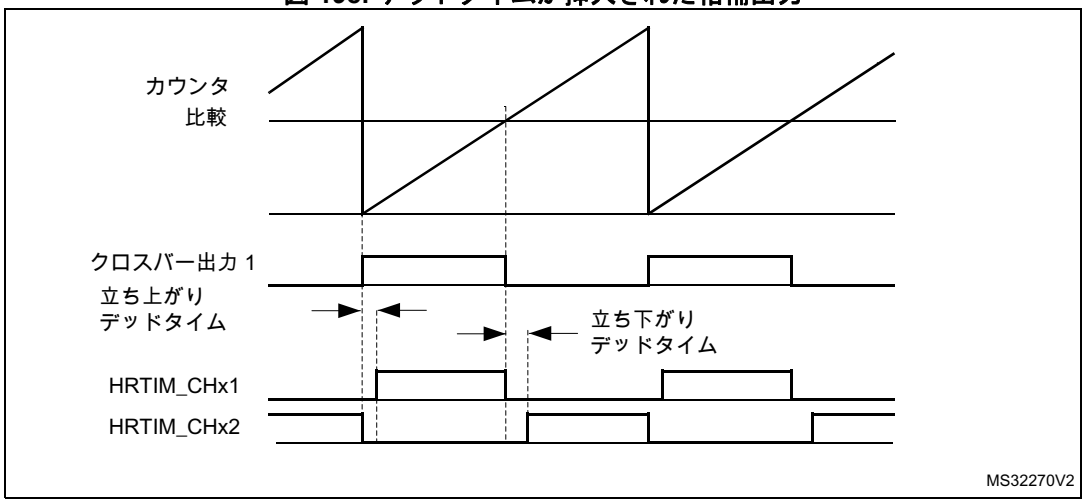
デッドタイム

デッドタイム挿入ユニットによって、単一の基準波形から、アクティブ状態の遷移間にプログラム可能な遅れを設定した 1 対の相補信号を生成することができます。これは一般にハーフブリッジまたはフルブリッジを使用する接続形態に使用されます。これによってソフトウェアが簡単になります。2 つの出力を駆動するのに、1 つの波形のみをプログラムして制御するだけで済みます。

デッドタイム挿入を有効にするには、HRTIM_OUTxR レジスタの DTEN ビットをセットします。HRTIM_SETx1R および HRTIM_RSTx1R レジスタを使用して、出力 1 に対して定義された基準波形に基づいて相補信号が作成されます。DTEN ビットをセットしているとき、HRTIM_SETx2R および HRTIM_RSTx2R レジスタは意味がありません。

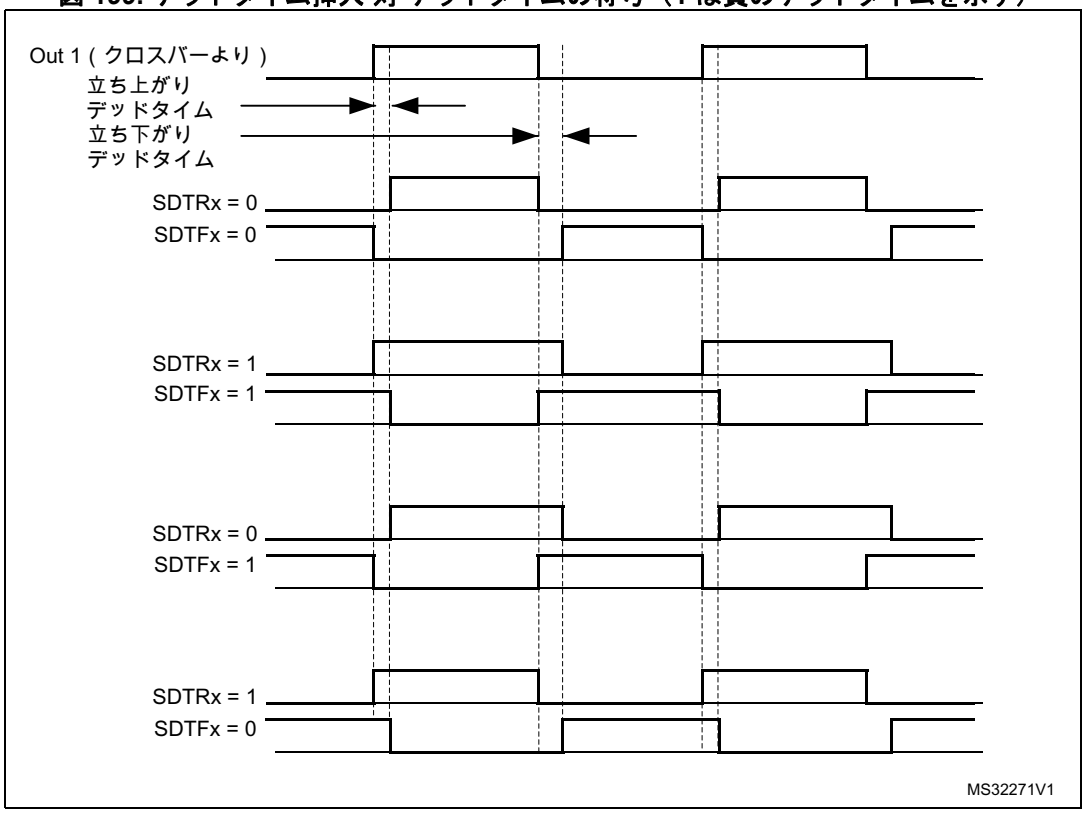
図 198 のように、基準波形の立ち上がりエッジと立ち下がりエッジに関連して 2 つのデッドタイムを定義できます。

図 198. デッドタイムが挿入された相補出力



制御のオーバーラップが必要なとき、負のデッドタイム値を定義できます。これは、デッドタイムの符号ビット（HRTIM_DTxR レジスタの SDTFx および SDTRx ビット）を使用することでできます。図 199 に、各々の符号に応じた相補信号波形を示します。

図 199. デッドタイム挿入 対 デッドタイムの符号（1 は負のデッドタイムを示す）



デッドタイムの値は、以下のとおり、DTFx[8:0] および DTRx[8:0] ビットフィールドによって、および DTPRSC[2:0] ビットに従ってプリスケールされた特定のクロックに基づいて定義されます。

t_{DTx} = +/- DTx[8:0] x t_{DTG}

ここで、x は R または F、t_{DTG} = ($2^{(DTPRSC[2:0])}$) x (t_{HRTIM} /8)。

表 220 は、プリスケーラの値に応じた分解能と最大の絶対値を示します。

表 220. デッドタイム分解能と最大の絶対値

DTPRSC[2:0]	t_{DTG}	t_{DTx} max	f_{HRTIM}= 170 MHz	
			t_{DTG}（ns）	$\|t_{DTx}\|$ max（µs）
000	t_{HRTIM} / 8	511 * t_{DTG}	0.73	0.38
001	t_{HRTIM} / 4		1.47	0.75
010	t_{HRTIM} / 2		2.94	1.50
011	t_{HRTIM}		5.89	3.01
100	2 * t_{HRTIM}		11.76	6.01
101	4 * t_{HRTIM}		23.53	12.03
110	8 * t_{HRTIM}		47.06	24.04
111	16 * t_{HRTIM}		94.12	48.10

図 200～図 203 に、すべてのデッドタイムの設定について、パルス幅がデッドタイムの値より小さい基準波形に対してデッドタイムジェネレータがどのように動作するかを示します。

図 200. 小パルス幅に対する相補出力（SDTRx = SDTFx = 0）

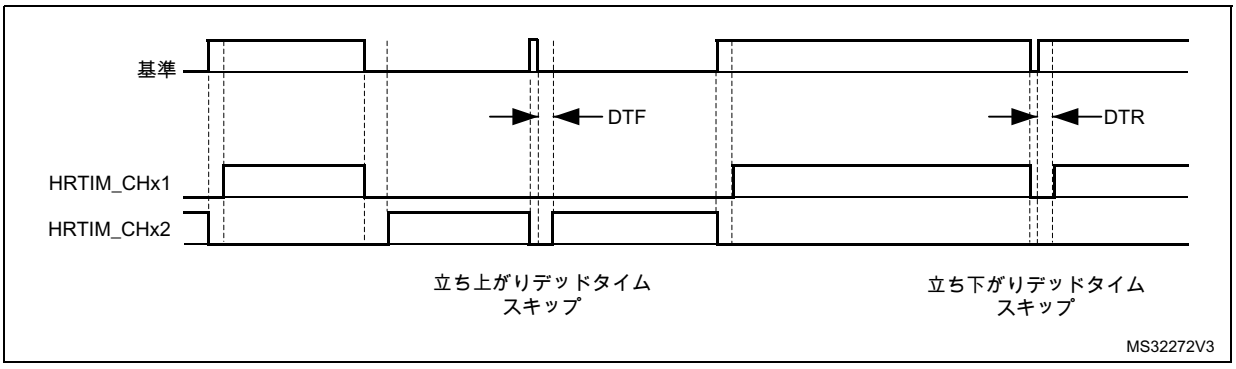


図 201. 小パルス幅に対する相補出力（SDTRx = SDTFx = 1）

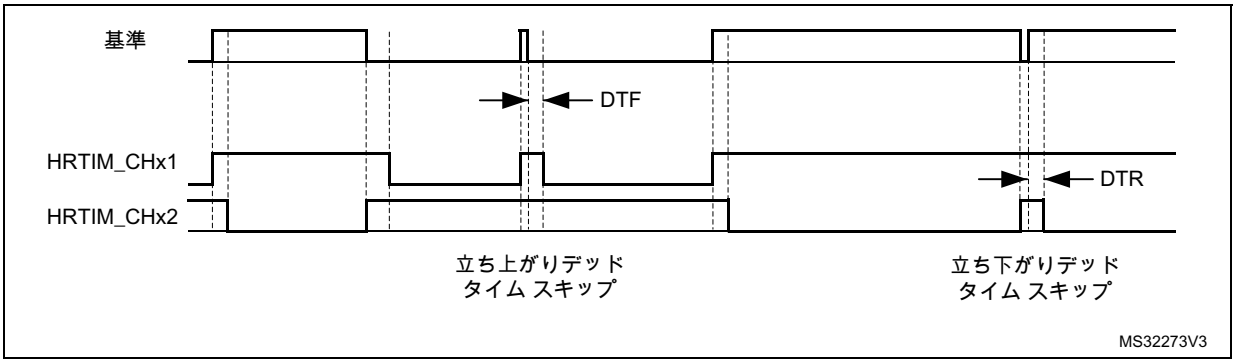


図 202. 小パルス幅に対する相補出力（SDTRx = 0、SDTFx = 1）

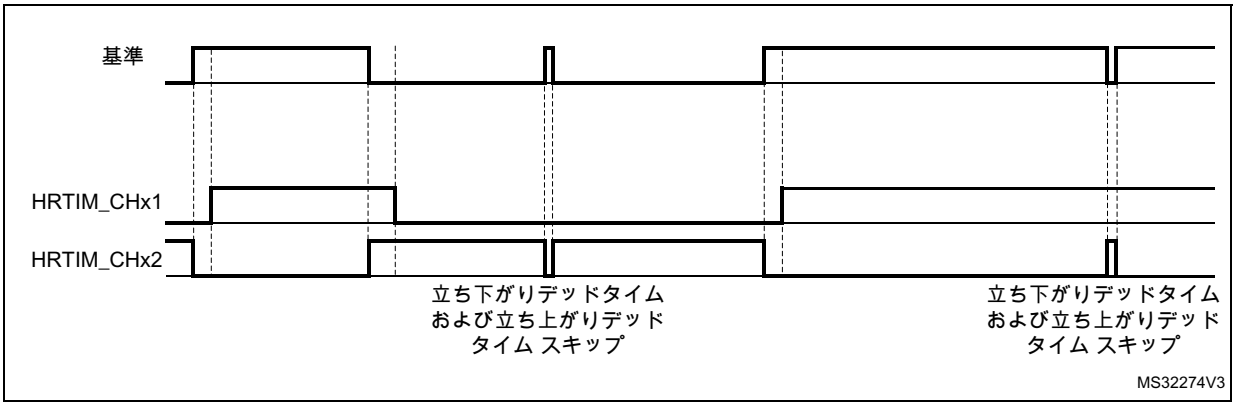
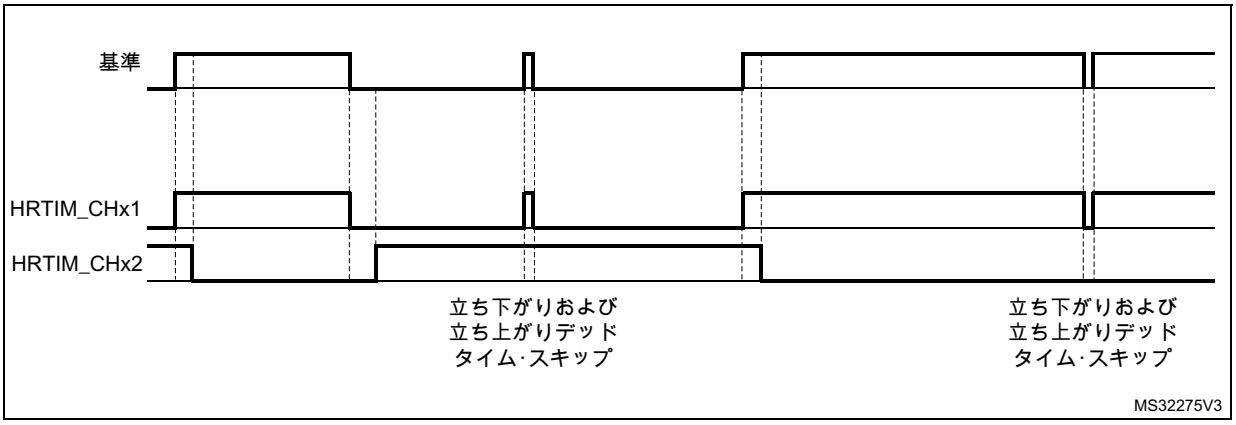


図 203. 小パルス幅に対する相補出力（SDTRx = 1、SDTFx = 0）



安全のため、DTFLKx、DTRLKx、DTFSLKx および DTRSLKx を使用して、デッドタイムの符号および／または値をロックすることによって、デッドタイムレジスタへの誤った書込みを防ぐことができます。これらのビットがセットされると、次のシステムリセットまで関連ビットおよびビットフィールドが読出し専用になります。

注意： **以下の場合、DTEN ビットは変更してはなりません。**

- **タイマが有効になっているとき（TxEN ビットをセット）**
- **タイマ出力が別のタイマによってセット／リセットされているとき（TxEN はリセットされている）**

そうしないと、予測できない動作になることがあります。
したがって、タイマを無効化して（TxCEN ビットをリセット）、対応する出力を無効化する必要があります。

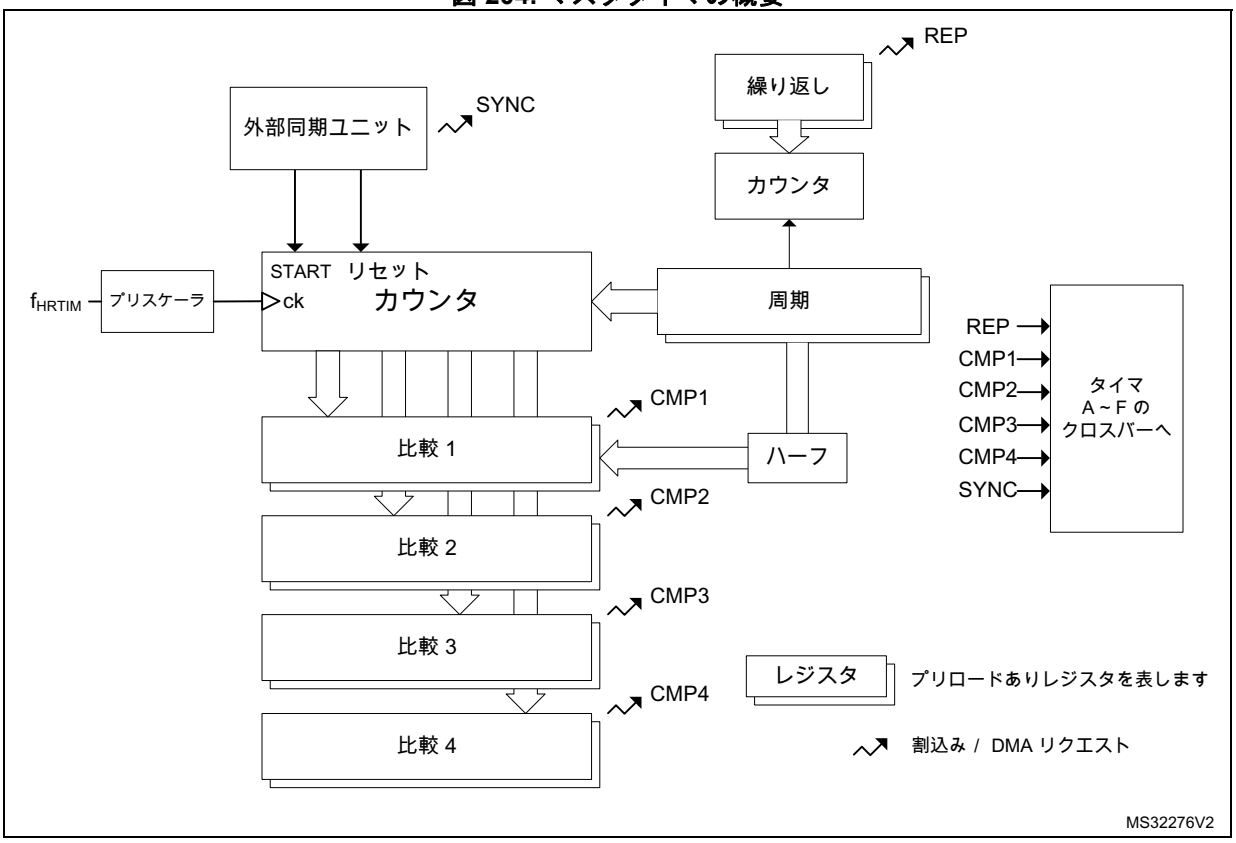
バーストモードが有効になっているときに、モード移行時にデッドタイムと共に DTEN をセットしなければならない特定の場合には（BME = 1、DIDL = 1、IDLEM = 1）、DTEN ビットをセットする前にソフトウェアコマンドで 2 つの出力を IDLES 状態に強制する（SST、RST ビット）必要があります。これは、バーストモードへの移行がデッドタイムが有効になる前に直ちに起きることから生じる副作用を避けるためです。

27.3.5 マスタタイマ

マスタタイマの主な目的は、同期目的のために、あるいは出力をセット／リセットするために、6 つのタイミングユニットに共通の信号を提供することです。マスタタイマは出力を直接制御することはしませんが、セット／リセットクロスバーによって間接的に行うことができます。

図 204 にマスタタイマの概要を示します。

図 204. マスタタイマの概要



マスタタイマはタイミングユニットとまったく同じアーキテクチャに基づいていますが、下記のような違いがあります。

- 関連する出力を持たず、出力に関連するコントロールも持ちません。
- 自身のクロスバーユニットも、プッシュプルまたはデッドタイムモードもありません。
- 外部の同期回路によってのみ、リセットできます。
- キャプチャユニットも、自動遅延モードもありません。
- 外部イベントのブランキングおよびウィンドウイング回路は含まれません。
- 割込み／ DMA リクエストは次のように限定されています。比較 1～4、繰り返し、レジスタ更新、および外部同期イベント。

マスタタイマ制御レジスタには、マスタおよびタイマ A～F のタイミングユニット用のすべてのタイマイネーブルビットが含まれています。これによって、1 回の書込みアクセスで、すべてのタイマを同期して開始することができます。

また、MCU の内部および外部（入出力）リソースの両方によって HRTIM タイマ全体の外部同期も行います（セクション 27.3.19：他のタイマまたは HRTIM インスタンスとの HRTIM の同期を参照）。

マスタタイマ制御レジスタは、タイミングユニットのレジスタと同じオフセットで配置されています。

27.3.6 アップダウンカウントモード

HRTIM は元々アップカウンタとして設計されています。しかし、アップダウンカウンタの動作モードが提供されています。センターアラインモードとも呼ばれます。

このモードを有効にするには、HRTIM_TIMxCR2 レジスタの UDM ビットを使用します。このビットは、タイマが動作（TxEN ビットのセット）された後は、変更できません。これは TIMA～F でのみ使用できます。マスタタイマはアップカウントモードでのみ動作します。

アップダウンカウントでは、HRTIM の一部の機能がサポートされていません。このセクションでは、アップカウントモードに対する機能の違いについて詳しく説明します。

アップダウンモードでは、HRTIM_PERxR の周期をプリロードする（または静的である）必要があります。それは周期イベント時またはカウンタリセット時にのみ更新できます。

セット／リセットクロスバーのプログラミングは、次のように異なっています。

タイミングユニットからのイベントによって､カウンタのアップ／ダウン方向に応じて出力がセット/リセットされます。

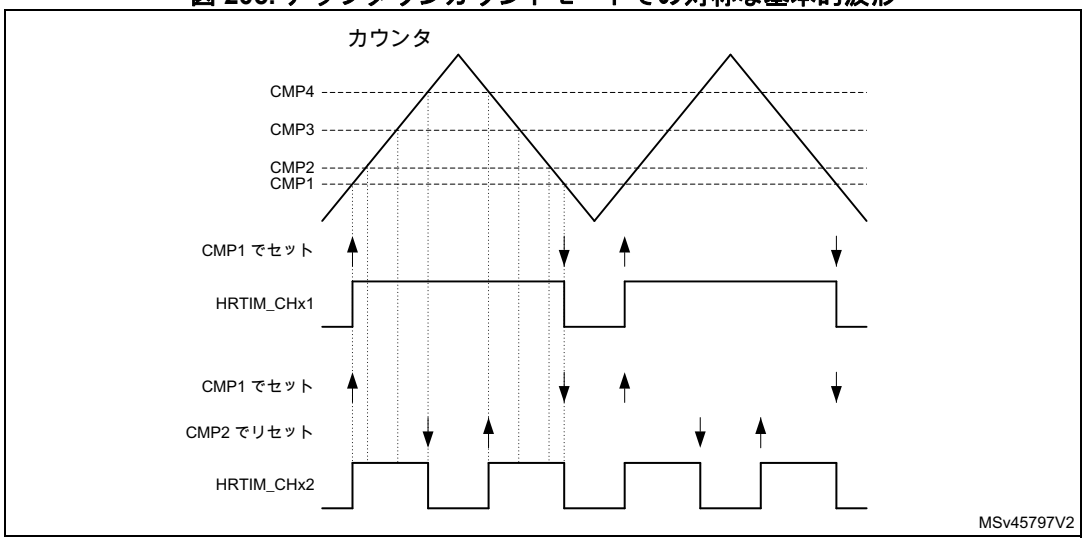
- HRTIM_SETxyR レジスタでイベントが有効になっている場合、アップカウント中に出力をセットし、ダウンカウント中に出力をリセットします。
- HRTIM_RSTxyR レジスタでイベントが有効になっている場合、アップカウント中に出力をリセットし、ダウンカウント中に出力をセットします。
- HRTIM_SETxyR および HRTIM_RSTxyR レジスタの両方でイベントが有効にされている場合は、出力は反転します。

このことは以下に適用されます。

  – タイミングユニット：周期、比較 1～4、レジスタ更新（6 つのイベント）
  – マスタタイマ：周期、比較 1～4、HRTIM 同期（6 つのイベント）
  – その他のすべてのタイミングユニット（たとえば、タイマ A に対してタイマ B～F）：TIMEVNT1～9（表 217 に記載された 9 つのイベント）

以下の図 205 に、基本的波形を生成する方法を示します。

図 205. アップダウンカウントモードでの対称な基本的波形



以下の図 206 は、4 つの使用可能な比較ユニットとトグルモードを使用して、より複雑な波形を生成する方法を示しています。

図 206. アップダウンカウントモードでの複雑な対称波形

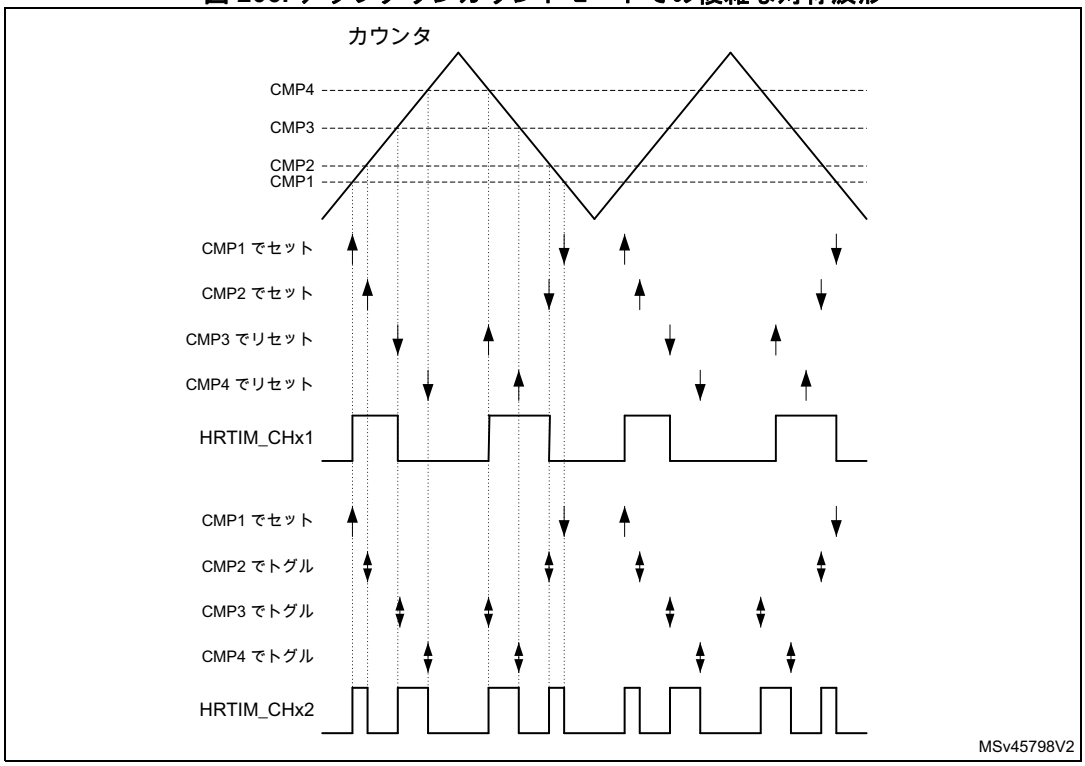
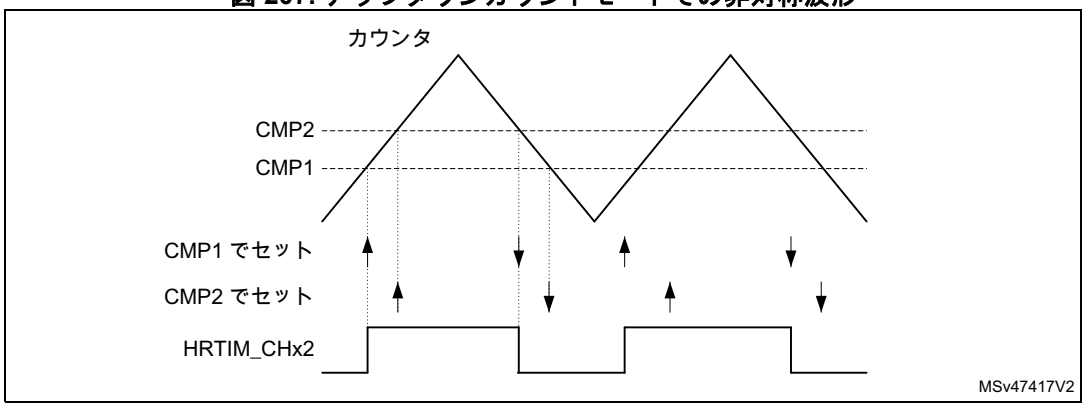


図 207 に、非対称な波形を生成する方法を示します。この場合、波形を非対称にするためには、比較 2 の値を比較 1 の値よりも大きくする必要があることに注意してください。

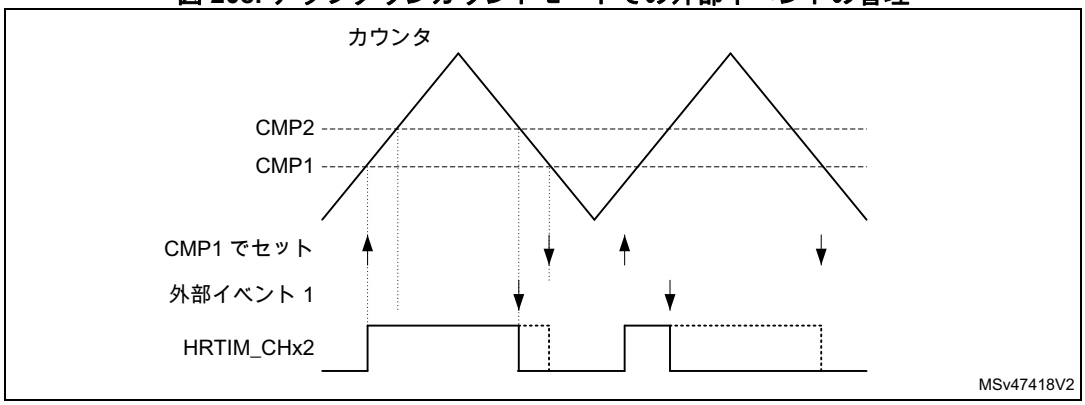
図 207. アップダウンカウントモードでの非対称波形



注：　非対称動作のためには、CMP2 > CMP1 とする必要があります。

ソフトウェア強制ビットと外部イベント EXTEVNT1～10 の動作は、アップのみのカウントモードとアップダウンカウントモードで同じです。以下の図 208 は、外部イベントに反応してパルスを短くする方法を示します。

図 208. アップダウンカウントモードでの外部イベントの管理



アップダウンカウントモードは、連続およびシングルショット（再トリガ可能および再トリガ不可）の両方の動作モードで使用できます。リセットすると、カウンタは 0 から再開します。以下の図 209 は、TimB のシングルショット再トリガ可能モードでのカウンタの動作を示しています。

図 209. インタリーブされたアップダウンカウンタ動作

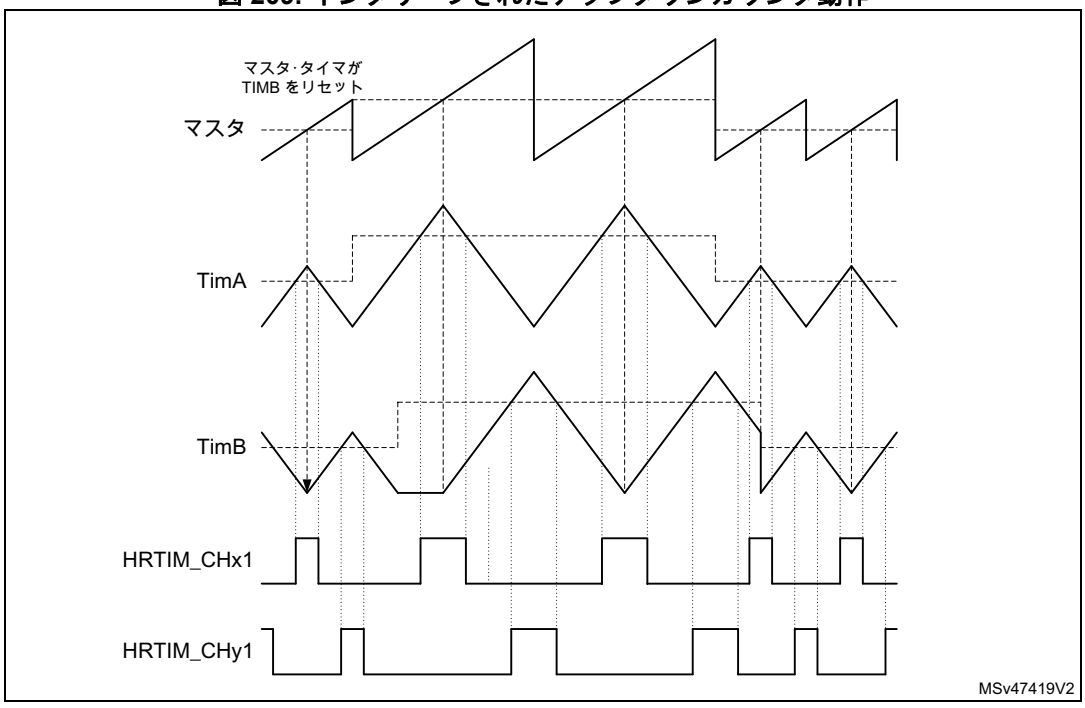
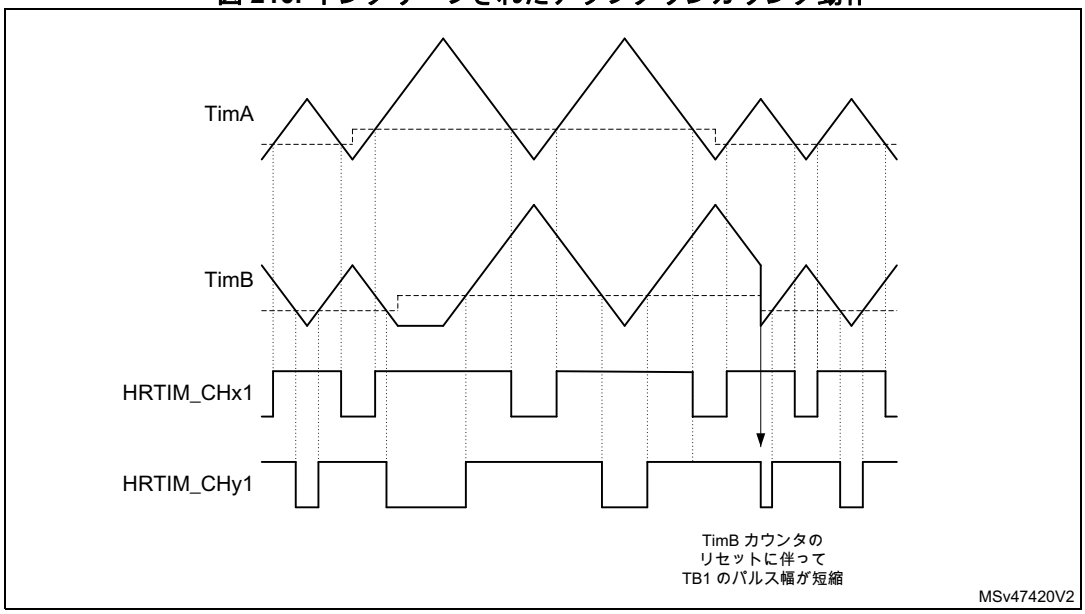


図 210. インタリーブされたアップダウンカウンタ動作



注： **アップダウンカウントモードでは、比較値は、周期値より fHRTIM クロックの 3 周期分下回る値（CKPSC[2:0] = 0 の場合は TIMx_PER - 0xC0、CKPSC[2:0] = 1 の場合は TIMx_PER - 0x60、CKPSC[2:0] = 2 の場合は TIMx_PER - 0x30、・・・）でなければなりません。これは、タイミングユニット内部で生成された比較イベントに適用されます。他のタイミングユニットで生成された比較イベントについては、カウンタ方向の変更（カウンタ 0 の時、周期イベント時、またはカウンタリセット時）の fHRTIM クロックの 1 周期未満で発生するイベントを回避する必要があります。**

アップダウンカウントモードでは、以下の機能がサポートされています。

- ハーフモード
- デッドタイムの挿入
- プッシュプルモード、カウンタ = 0 の時に実行される交互プッシュプル（図 211 参照）
- 遅延アイドル
- バーストモード
- 「より大きい」比較を備えた PWM モード（図 212 参照）

図 211. プッシュプル・アップダウンモードの例

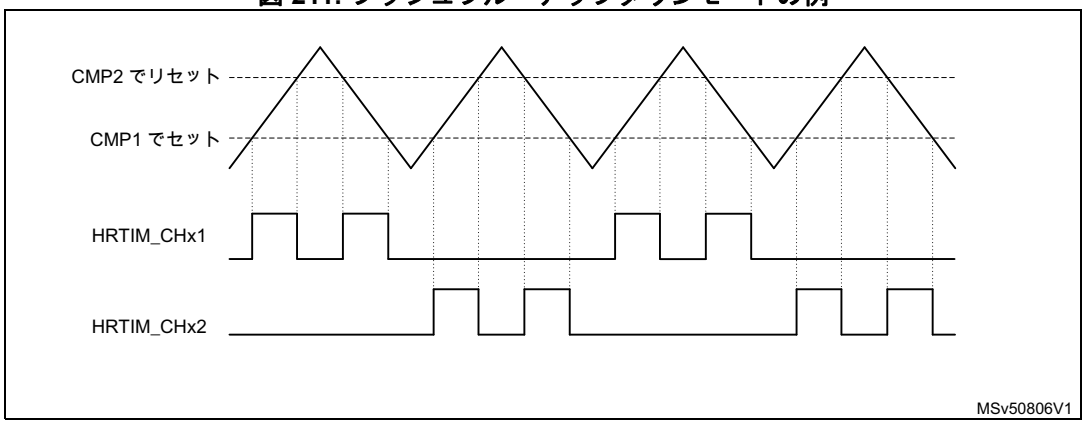
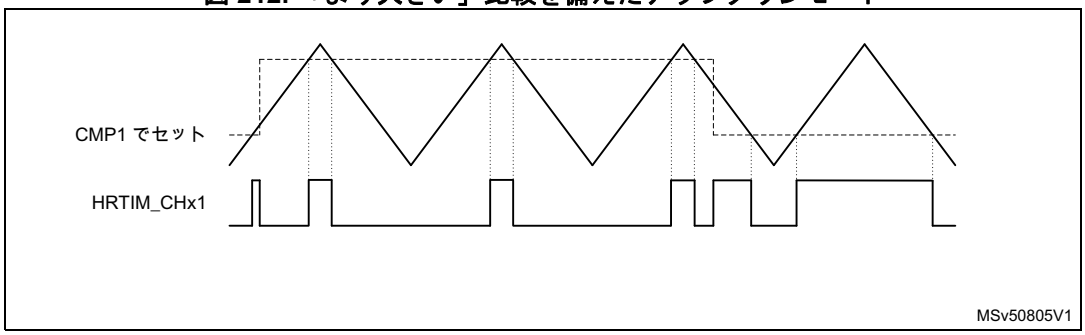


図 212. 「より大きい」比較を備えたアップダウンモード



注意： **アップダウンカウントモードでは、以下の機能はサポートされていません。**

- 自動遅延モード
- バランス·アイドル
- トリガハーフ·モード

キャプチャ機能はサポートされていますが、以下の違いがあります。

- アップカウントの場合、キャプチャ値はカウント開始に参照されます。
- ダウンカウントの場合、キャプチャ値は PER イベントに参照されます。
- キャプチャレジスタのビット 16 には、カウント方向ステータスが保持されます。

カウンタのロールオーバーイベントは、さまざまな動作条件をサポートするために、アップダウンカウントモードで異なる方法で定義されます。同イベントは下記のいずれかで生成されます。

- カウンタが 0 に等しい時（「谷」モード）
- カウンタが、HRTIM_PERxR でセットされた周期値に達した時（「山頂」モード）
- 両方の条件が満たされた時（0 または HRTIM_PERxR 値）

このイベントは、HRTIM において複数の目的で使用されます。生成モード（谷、山頂、または両方）は、転送先に応じて個別にプログラムすることができます。表 221 は、使用例と、HRTIM_TIMxCR2 レジスタの関連するロールオーバーモード（xxROM[1:0]）プログラミングビットをまとめたものです。

表 221. ロールオーバーイベントの転送先とモードのプログラミング

ロールオーバーイベントの用途	プログラミングビット
出力のセット／リセット	OUTROM[1:0]
レジスタの内容の更新トリガ（プリロードからアクティブへの変更）	ROM[1:0]
IRQ または DMA リクエスト（あるいはその両方）のトリガ	ROM[1:0]
バースト·モードのクロック·ソースまたはバースト開始（あるいはその両方）のトリガ	BMROM[1:0]
ADC トリガ（詳細はセクション ：ADC ポストスケーラを参照）	ADROM[1:0]
外部イベントのフィルタリング	ROM[1:0]
繰り返しカウンタのデクリメント	ROM[1:0]
フォールトおよびイベントのカウンタ	FEROM[1:0]

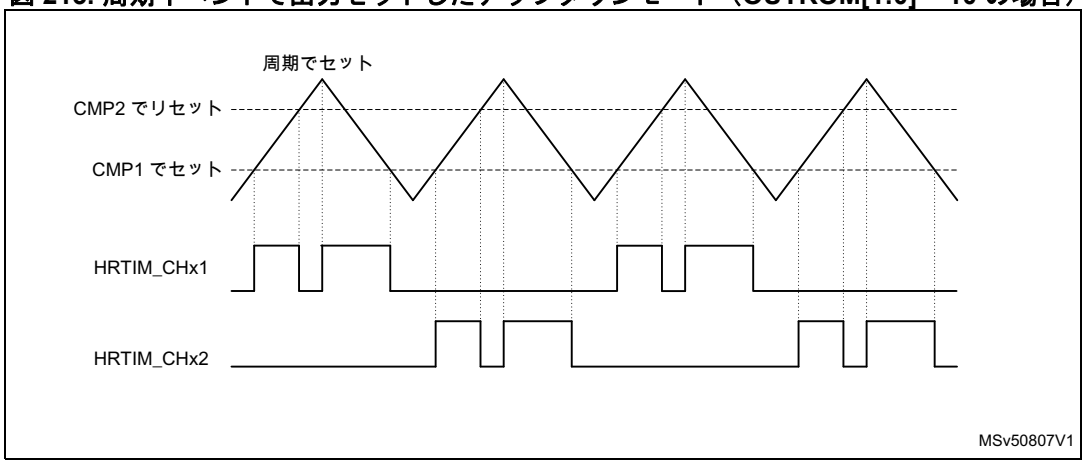
注： **リセットとロールオーバーの両方が考慮されるイベント（IRQ/DMA および TxRSTU）の場合、ROM[1:0] はロールオーバー生成のみに影響します。リセットイベントは、ROM[1:0] の値に関係なく常に考慮されます。**

ロールオーバーイベントの生成は、次の xxROM[1:0] ビットフィールドの設定に従って定義されます。

- xxROM[1:0] = 00 : 両方の条件が満たされた時（0 または HRTIM_PERxR 値）にイベントが生成されます。
- xxROM[1:0] = 01 : カウンタが 0 に等しい時（「谷」モード）またはカウンタがリセットされた時に、イベントが生成されます。
- xxROM[1:0] = 10 : カウンタが、HRTIM_PERxR でセットされた周期値に達した時（「山頂」モード）にイベントが生成されます。

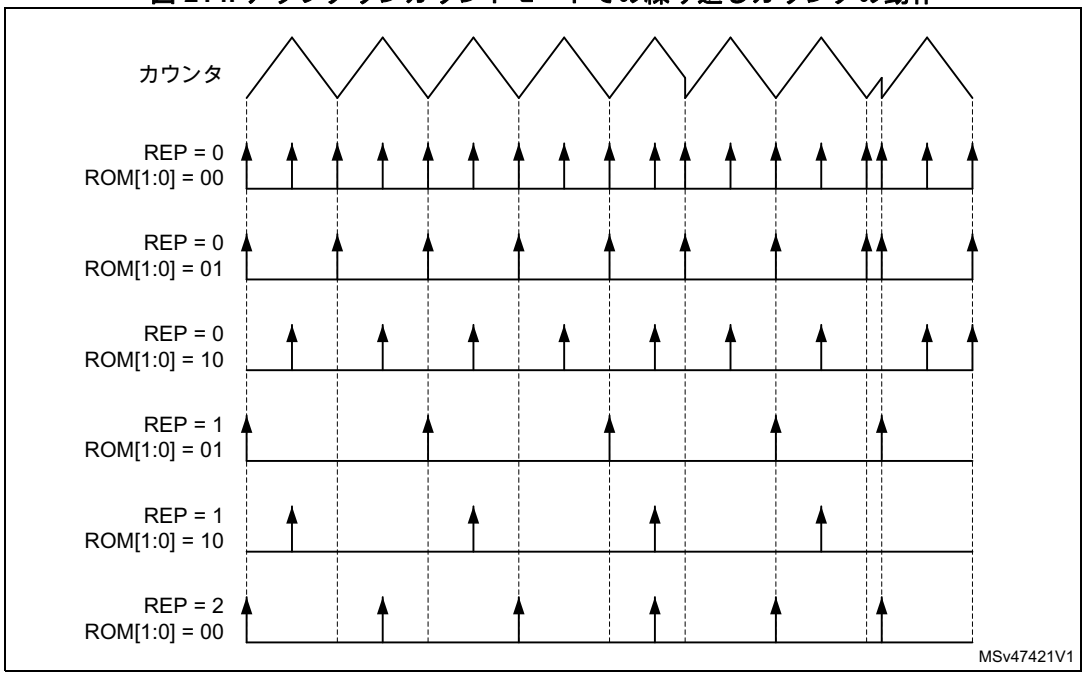
下記の図 213 は、周期イベントおよび OUTROM[1:0] =10 で設定されたプッシュプル・アップダウンモードを示します。

図 213. 周期イベントで出力セットしたアップダウンモード（OUTROM[1:0] = 10 の場合）



下記の図 214 は、アップダウンカウントモードでどのように繰り返しカウンタがデクリメントされるかを示しています。

図 214. アップダウンカウントモードでの繰り返しカウンタの動作



デュアルチャネル DAC トリガは、アップカウントモードと同様に機能します。

イベントブランキングとウィンドウイングは異なり、プログラム可能な時間に出力パルス内でブランキングまたはウィンドウイングが行われます。EExFLTR[3:0] コードは、下記表 222 のように UDM ビットの設定によって意味が変わります。ロールオーバーイベントがブランキングまたはウィンドウイングに使用される場合は常に、それがいつ生成されるかを定義するために ROM[1:0] プログラミングが適用されます。

表 222. UDM ビットの設定によって変わる EExFLTR[3:0] コード

EE1FLTR[3:0]	アップカウント･モード （UDM = 0）	アップダウンカウントモード （UDM = 1）
0010	カウンタのリセット／ロールオーバーから 比較 2 までブランキング	比較 1 から比較 2 までブランキング （アップカウントフェーズ中のみ）
0100	カウンタのリセット／ロールオーバーから 比較 4 までブランキング	比較 3 から比較 4 までブランキング （アップカウントフェーズ中のみ）
1101	カウンタのリセット／ロールオーバーから 比較 2 までウィンドウイング	比較 2 から比較 3 までウィンドウイング （アップカウントフェーズ中のみ）
1110	カウンタのリセット／ロールオーバーから 比較 3 までウィンドウイング	比較 2 から比較 3 までウィンドウイング （ダウンカウントフェーズ中のみ）
1111	別のタイミングユニットから ウィンドウイング：TIMWIN ソース （詳細は 表 226 を参照）	アップカウント･フェーズ中の比較 2 から ダウンカウント･フェーズ中の比較 3 まで ウィンドウイング

27.3.7 セット／リセットイベントの優先順位と狭いパルスの管理

このセクションでは、複数のセットおよび／またはリセットリクエストが連続する t_{HRTIM} の 3 周期内に発生した場合に出力波形がどのように生成されるかを説明します。

ケース 1：クロックプリスケーラ CKPSC[2:0] < 5

各 t_{HRTIM} 周期内で以下の 3 ステップで調整が行われます。

1. 各アクティブイベントに対して、望まれる出力遷移を判断します（セット、リセット、または反転）。
2. アクティブイベントの中で、事前に定義された調整が行われます（最高から最低の優先順位に、CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1 ⟶ PER。セクション ：同時のセットリクエスト／同時のリセットリクエストを参照）。
3. 高分解能の遅延ベースの調整は、低分解能のイベントと、事前に定義された調整に勝ち残ったイベントの中で、最も高い優先度を持つリセットによって実行されます。

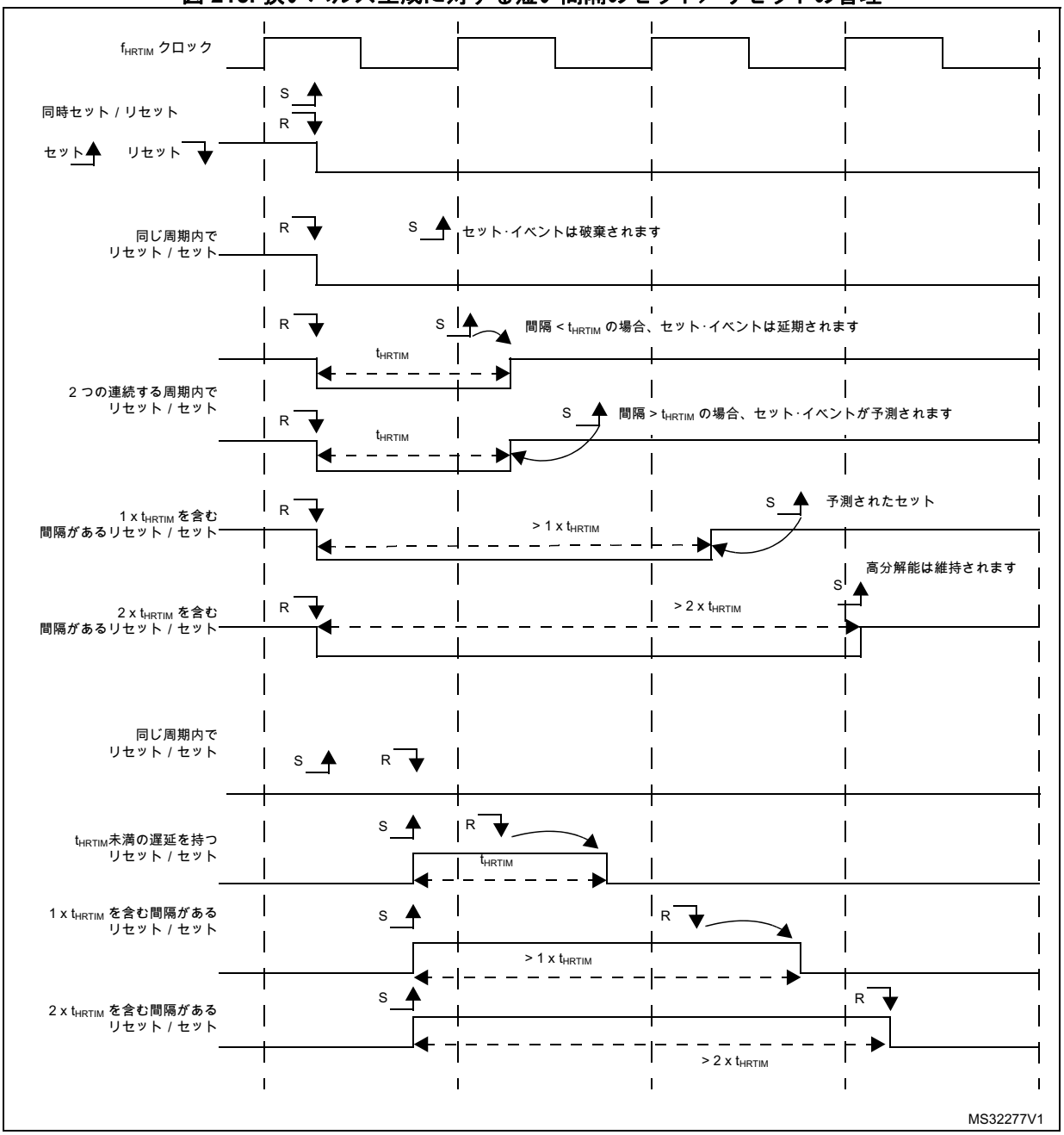
2 つの異なるソースからセットとリセットのリクエストが同時に来たとき、リセット動作が最高の優先順位になります。セット要求とリセット要求の間隔が 2 f_{HRTIM} 周期より短い場合、動作は、図 215 に示すように、時間間隔と、f_{HRTIM} クロックとのアライメントに依存します。

注： **セットとリセットのリクエストが同時に発生し、同じタイミングユニットから送信される場合、上記の手順 2 に示すように、CMPx の優先度が適用されます。たとえば、CMP2 = CMP4 の場合：**

- CMP2 がセットを、CMP4 がリセットをリクエストした場合、出力はリセットされます。

- CMP2 がリセットを、CMP4 がセットをリクエストした場合、出力はセットされます。

図 215. 狭いパルス生成に対する短い間隔のセット／リセットの管理



セットとリセットイベントが同じ t_{HRTIM} 周期内に生成された場合、リセットイベントが最も高い優先度を持ち、セットイベントは無視されます。

セットとリセットイベントが 2 周期にわたって t_{HRTIM} 周期より短い間隔で生成された場合、1 t_{HRTIM} 周期のパルスが生成されます。

セットとリセットイベントが 2 t_{HRTIM} 周期より短い間隔で生成された場合、2 t_{HRTIM} 周期のパルスが生成されます。

セットとリセットイベントが 2～3 t_{HRTIM} 周期の間の間隔で生成された場合、その間隔が完全な 2 t_{HRTIM} 周期を超えていた場合は、高分解能が利用できます。

セットとリセットイベントが 3 t_{HRTIM} 周期を超える間隔で生成された場合、常に高分解能が利用できます。

同時のセットリクエスト／同時のリセットリクエスト

セットイベントに複数のソースが選択された場合、同じ f_{HRTIM} クロック周期内で複数のリクエストが発生したとき、調整が行われます。

隣接するタイマからの複数のリクエスト（TIMEVNT1～9）の場合、最初に発生したリクエストが考慮されます。調整は、以下の内容に応じて、2 ステップで行われます。

1. ソース（CMP4 ⟶ CMP3 ⟶ CMP2 ⟶ CMP1）
2. 遅延

マスタタイマから複数のリクエストが同じ f_{HRTIM} クロック周期内に発生した場合、有効な高分解能設定に関係なく、事前に定義された調整が適用され、単一のリクエストが考慮されます（最高から最低の優先順位で）。

すなわち、MSTCMP4 → MSTCMP3 → MSTCMP2 → MSTCMP1 → MSTCMPER。

注： **高分解能を維持するために、3x t_{HRTIM} 未満の間隔で、マスタタイマから特定のタイマへ複数のセット（リセット）リクエストを生成することは避けることをお勧めします。**

内部からタイマへの複数のリクエストが同じ f_{HRTIM} クロック周期内に発生した場合、事前に定義された調整が適用され、リクエストは、有効なタイミングであれば何であれ、次の優先順位で取り込まれます（最高から最低の順で）。

すなわち、CMP4 → CMP3 → CMP2 → CMP1 → PER。

注： **実際には、複数の比較イベントが同時に生成される可能性があるとき、または自動遅延比較 2 と比較 4 を同時に使用するとき（すなわち、外部イベントに関係しているので、有効なセット／リセットを事前に決められないとき）、これは最も重要です。この場合、最も高い優先順位の信号を CMP4 イベントに影響させる必要があります。**

最後に、最も高い優先順位は、次の低分解能イベントに与えられます。EXTEVNT1～10、RESYNC（SYNCRSTx または SYNCSTRTx がセットされている場合の SYNC イベントから、またはソフトウェアリセットから来るもの）、更新、およびソフトウェアセット（SST）。更新イベントは最も大きな遅延があると見なされています（PSC = 0 の場合、0x1F）。

まとめると、近接している（同じ f_{HRTIM} クロック周期内に発生するイベントの）場合、有効なセット（リセット）イベントは以下の間で調整されます。

- 任意の TIMEVNT1～9 イベント
- マスタからの単一のソース（上記で与えられた固定の調整による）
- タイマからの単一のソース
- 「低分解能イベント」

同時のリセットリクエストにも同じ調整原則が適用されます。この場合、リセットリクエストは最も高い優先順位になります。

ケース 2：クロックプリスケーラ CKPSC[2:0] ≥ 5

高分解能が有効でない場合、狭いパルスの管理は単純化されます。

プリスケーラクロックサイクル内で起きるセットまたはリセットイベントは、たとえ調整が t_{HRTIM} サイクルごとに行われていても、プリスケーラクロックの次のアクティブエッジまで遅延されます（カウンタリセットに関して）。

同じプリスケーラクロックサイクル内で、リセットイベントの後にセットイベントが発生した場合、最後のイベントが考慮されます。

27.3.8 外部イベントのグローバル調整

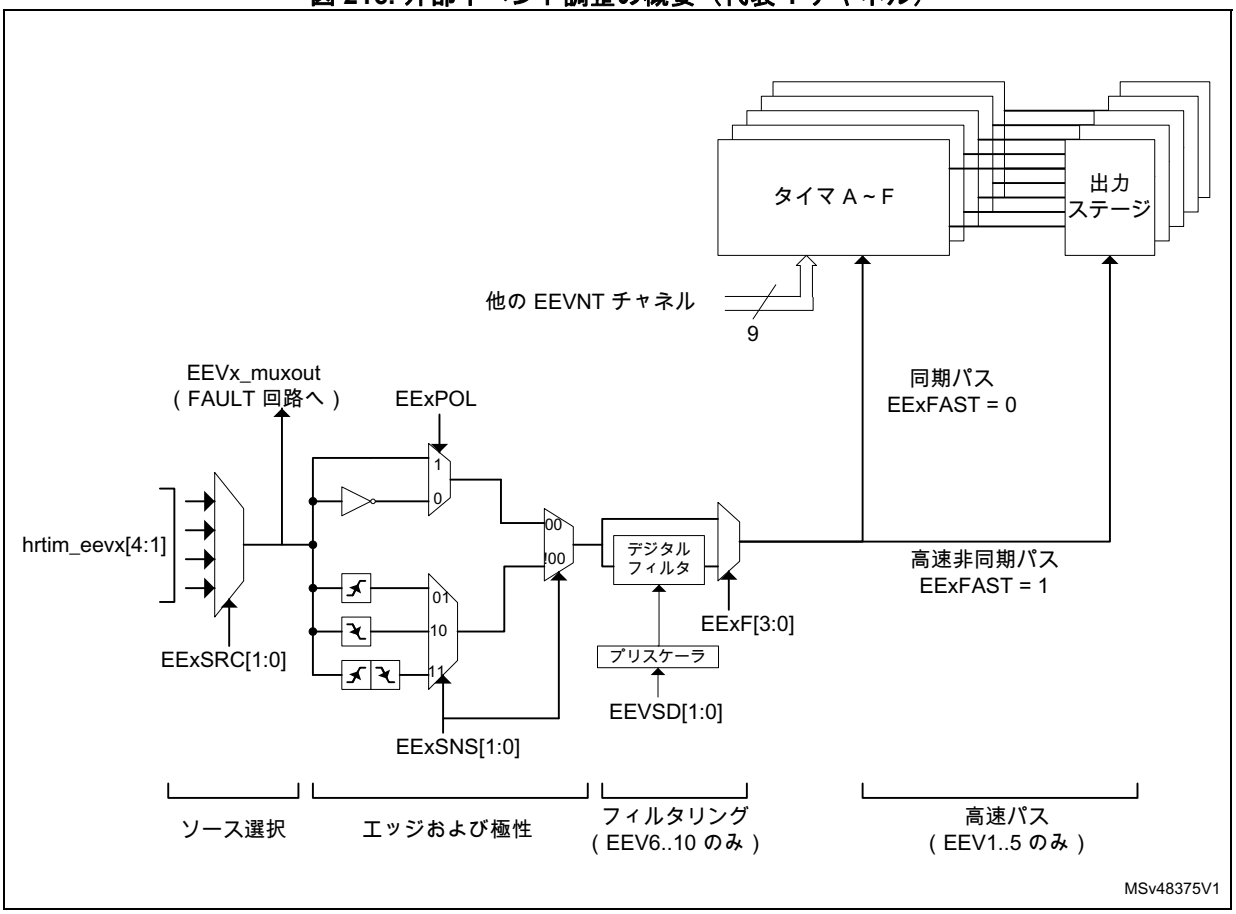
HRTIM タイマは、タイマ内で生成されないイベント（「外部イベント」と呼びます）を処理できます。これらの外部イベントは、以下のオンチップやオフチップの複数のソースから来ます。

- 内蔵コンパレータ、
- デジタル入力ピン（典型として、オフチップコンパレータおよびゼロ交差検出回路に接続されたピン）、
- 他のペリフェラル用のオンチップイベント（ADC のアナログウォッチドッグおよび汎用タイマトリガ出力）。

外部イベント調整回路によって、(4:1 マルチプレクスサ付きの）所定チャネルに対して信号ソースを選択すること、そして、それをクロスバーユニットで処理できる情報に変換すること（たとえば、外部イベントチャネルの立ち下がりエッジ検出をトリガとして出力をリセットさせるなど）ができます。

最大 10 個の外部イベントチャネルを調整でき、6 つのどのタイマにも同時に使用できます。この調整は、通常外部コンポーネント（ゼロ交差検出回路など）および環境条件によって要求されるので（一般にフィルタのセットアップはアプリケーションのノイズレベルや特性に関連します）、すべてのタイマに共通です。図 216 に、単一のチャネルに対する調整ロジックの概要を示します。

図 216. 外部イベント調整の概要（代表 1 チャネル）



HRTIM_EECR1 および HRTIM EECR2 レジスタを使用して 10 個の外部イベントが以下のように初期化されます。

- EExSRC[1:0] ビットによって最大 4 つのソースを選択します。
- EExSNS[1:0] ビットによって感度を選択し、レベル対応にするか、またはエッジ対応（立ち上がりか、立ち下がりか、両方）にします。
- レベル対応の場合、EExPOL ビットによって極性を選択します。
- 外部イベント 1～5 に対して、EExFAST ビット（セクション ：外部イベントへの遅延時間を参照）によって低遅延時間モードにします。

注： **リセット、キャプチャ、バーストモード、ADC トリガ、および遅延保護のトリガとして使用される外部イベントは、EESNS ビットがリセット（レベル対応の選択）されていても、エッジ対応となります。POL = 0 の場合、外部イベントの立ち上がりエッジでトリガがアクティブになり、POL = 1 の場合、外部イベントの立ち下がりエッジでトリガがアクティブになります。**

出力状態の変化およびカウンタのリセットを防ぐためにカウンタが無効にされている限り（TxCEN ビットのリセット）、ADC トリガとして使用される場合を除き、外部イベントは破棄されます。

さらに、HRTIM_EECR3 レジスタの EExF[3:0] ビットを使用して、外部イベント 6～10 に対してデジタルノイズフィルタを有効にすることが可能です。

デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。カウンタが N の値に達する前に入力値が変化した場合、カウンタはリセットされ、変化は破棄されます（誤ったイベントとみなされます）。カウンタが N に達した場合、変化は有効とみなされ、正しい外部イベントとして転送されます。したがって、デジタルフィルタは、サンプリングクロックとフィルタ長（予想される有効なサンプルの数）に応じて、フィルタリングされる外部イベントに遅延時間を付加します。

サンプリングクロックは f_{HRTIM} クロックか、または f_{HRTIM} から導き出される特定のプリスケールされたクロック f_{EEVS}（HRTIM_EECR3 レジスタの EEVSD[1:0] ビットで定義されます）になります。

表 223 に、10 個の外部イベントチャネルの各々に関連して利用可能な機能をまとめます。機能とソースは表 209 にまとめています。

表 223. 外部イベントの機能

外部イベントチャネル	高速モード	デジタルフィルタ	バランスフォールトタイマ A、B、C	バランスフォールトタイマ D、E、F
hrtim_eev1[4:1]	可能	-	-	-
hrtim_eev2[4:1]	可能	-	-	-
hrtim_eev3[4:1]	可能	-	-	-
hrtim_eev4[4:1]	可能	-	-	-
hrtim_eev5[4:1]	可能	-	-	-
hrtim_eev6[4:1]	-	可能	可能	-
hrtim_eev7[4:1]	-	可能	可能	-
hrtim_eev8[4:1]	-	可能	-	可能
hrtim_eev9[4:1]	-	可能	-	可能
hrtim_eev10[4:1]	-	可能	-	-

外部イベントへの遅延時間

外部イベント調整によって、性能の期待に応じて外部イベント処理時間（および関連する遅延時間）を以下のように調節することができます。

- 出力クロスバーに作用する前に外部イベントがクロックで再サンプリングされる、通常の動作モード。これによって、いくらかの遅延時間が付加されますが、すべてのクロスバー機能にアクセスできます。外部からトリガされる高分解能パルスを生成できます。
- 外部イベントと出力への作用の間の遅延時間が最小になる高速動作モード。このモードは、たとえば、超高速の過電流保護に便利です。

HRTIM_EECR1 レジスタの EExFAST ビットによって、チャネル 1～5 に対する動作を定義することができます。これによって、下記表に要約されるように、出力パルス上の遅延時間とジッタが影響を受けます。

表 224. 出力のセット／リセット遅延時間とジッタ 対 外部イベント動作モード

EExFAST	応答時間の遅延	応答時間のジッタ	出力パルス上のジッタ（外部イベントによるカウンタリセット）
0	f_{HRTIM} クロックの 5～6 サイクル	f_{HRTIM} クロックの 1 サイクル	ジッタなし、パルス幅は高分解能で維持
1	最小遅延時間（コンパレータが使用されるかデジタル入力が使用されるかによる）	最小ジッタ	f_{HRTIM} クロックの 1 サイクルのジッタ、パルス幅分解能は t_{HRTIM} より下がる

EExFAST はレベル対応のプログラミング（EExSNS[1:0] = 00）でのみ使用可能であり、エッジ対応はプログラムできません。

外部イベントにイベントフィルタを適用できます（EExFLTR[3:0] != 0000 によりブランキングとウィンドウイングの両方。セクション 27.3.9：参照）。この場合、EExLTCHx ビットはリセットする必要があります。延期モードはサポートされておらず、ウィンドウイングのタイムアウト機能も同じです。

注： **関連する EExFAST ビットをセットすると、外部イベントの設定（ソースおよび極性）は変更してはなりません。**

高速外部イベントは出力のトグルには使用できません。HRTIM_SETxyR または HRTIM_RSTxyR のどちらかで（両方ではなく）有効にする必要があります。

独立の 2 つの高速外部イベントからセットとリセットのイベントが同時に発生した場合は、クロスバーではリセットが最大の優先順位を持ち、出力は非アクティブになります。

EExFAST ビットがセットされているとき、外部イベントの後、f_{HRTIM} の 11 クロック周期の間、出力は変更できません。

図 217 と 図 218 に、出力セット／リセットおよびカウンタリセットの外部イベントへの反応時間の実用例を示します。

図 217. 外部イベントの立ち下がりエッジの遅延時間（カウンタリセットおよび出力セット）

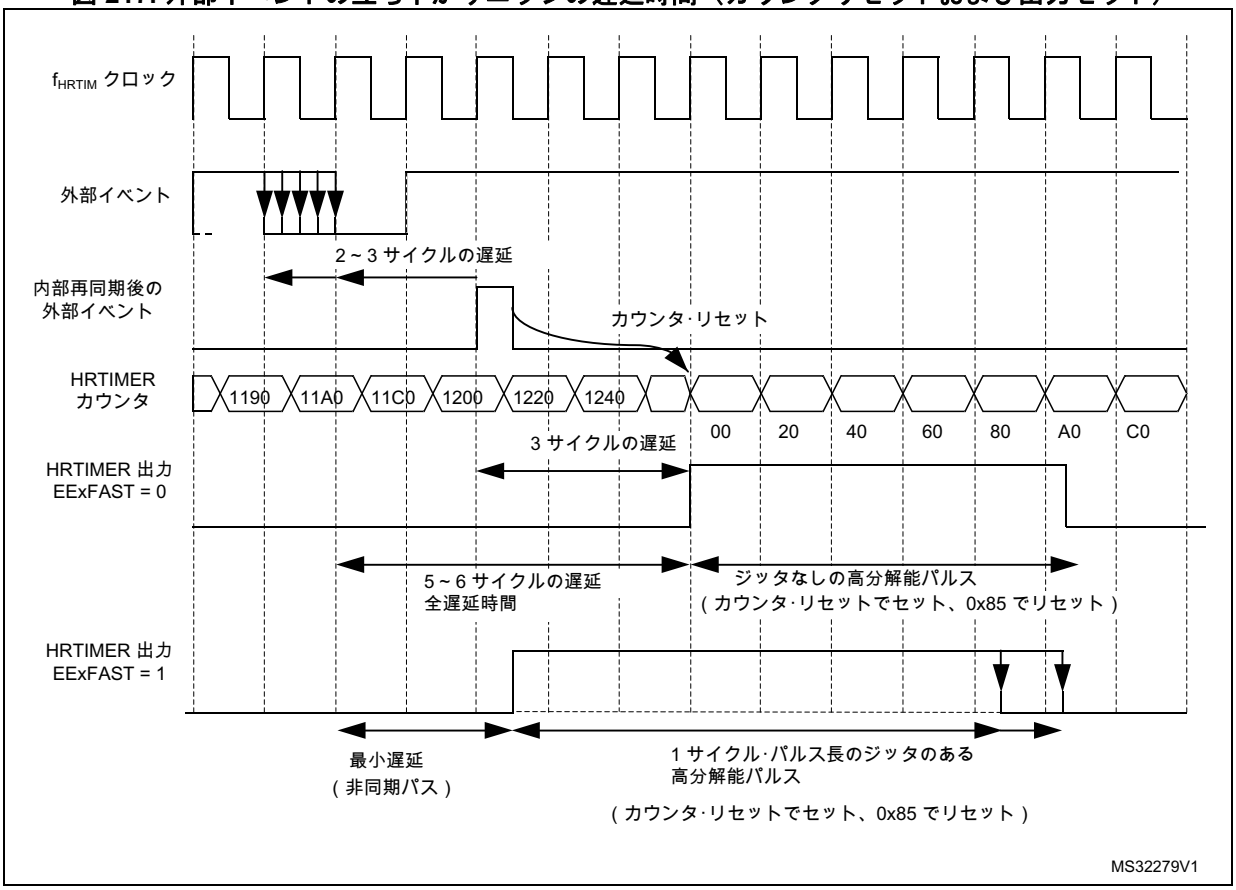
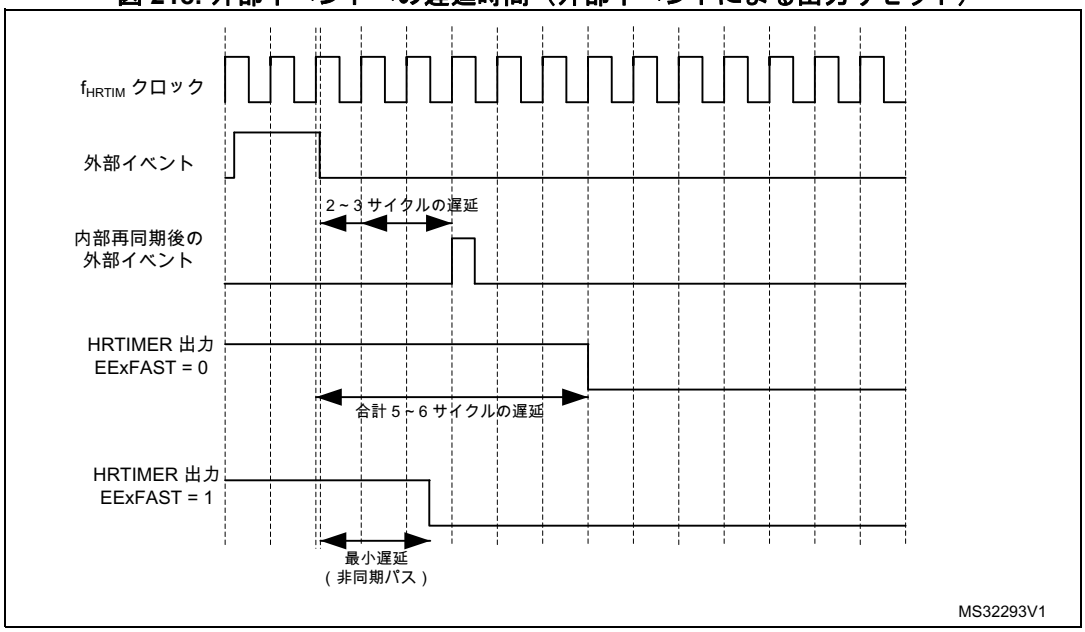


図 218. 外部イベントへの遅延時間（外部イベントによる出力リセット）



27.3.9 タイミングユニットでの外部イベントのフィルタリング

調整が終わると、すべてのタイミングユニットで 10 個の外部イベントが利用可能になります。

これらは直接使用可能で、タイミングユニットのカウンタが有効になると（TxCEN ビットのセット）すぐにアクティブになります。

また、通常、カウント周期に関連して、動作に時間的制限を与えるようにフィルタリングすることもできます。次の 2 つの動作を行わせることができます。

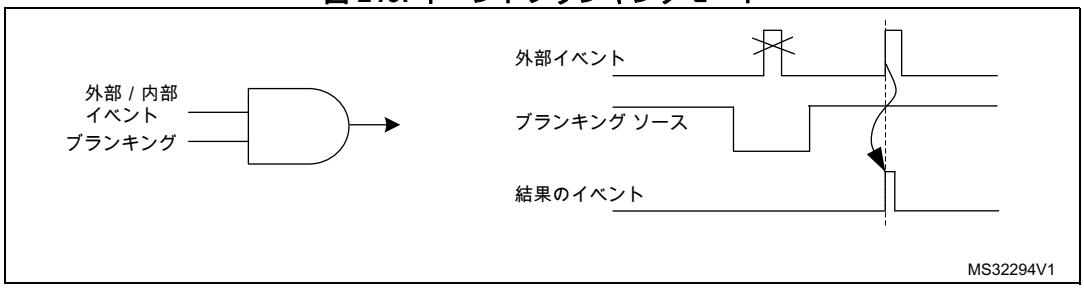
- ブランキング。定義された時間の間、外部イベントをマスクします。
- ウィンドウイング。定義された時間の間のみ、外部イベントを有効にします。

これらのモードは、HRTIM_EEFxR1 および HRTIM_EEFxR2 レジスタの HRTIM_EExFLTR[3:0] ビットを使って有効化されます。6 つのタイマ A～F のタイミングユニットの各々には、10 個の外部イベントに対して独自のプログラム可能なフィルタ設定が用意されています。

ブランキングモード

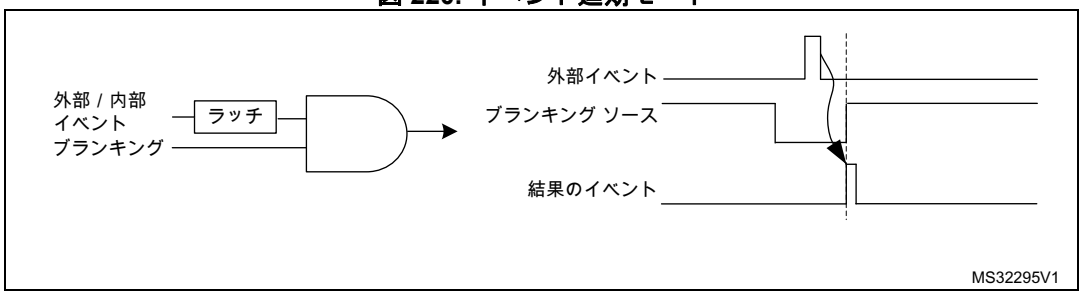
イベントブランキングモードでは（図 219 参照）、決められたブランキング期間中に発生した外部イベントは無視されます。これは、たとえば、PWM 周期の始めのスイッチングノイズによるトリップの電流限度を避けるのに便利です。このモードは、0001～1100 の範囲の EExFLTR[3:0] ビットフィールドの値に対してアクティブになります。

図 219. イベントブランキングモード



イベント延期モードでは、図 220 に示すように、外部イベントはすぐには考慮されませんが、記憶され（ラッチされ）て、ブランキング期間が終わると直ちに生成されます。このモードを有効にするには、HRTIM_EEFxR1 および HRTIM_EEFxR2 レジスタの EExLTCH ビットをセットします。

図 220. イベント延期モード



ブランキング信号は以下のいくつかのソースから受信します。

- タイマ自身：ブランキングはカウンタリセットから比較一致まで続きます（比較 1 ～比較 4 に対して、EExFLTR[3:0] = 0001～0100）。アップ／ダウンモード（UDM ビットを 1 に設定）では、カウンタリセットイベントは ROM [1:0] ビットの設定に従って定義されます。
- 他のタイミングユニットから（EExFLTR[3:0] = 0101～1100）：ブランキングは、選択したタイミングユニットのカウンタリセットから比較の 1 つの一致まで続き、あるいは Tx2 出力の波形

として完全にプログラム可能です。この場合、Tx2 信号が非アクティブである限り、イベントはマスクされます（出力を有効にする必要はなく、信号は出力ステージの前で取られます）。

EEXFLTR[3:0] の 0101～1100 の設定は、ビットの説明では TIMFLTR1～TIMFLTR8 と呼ばれ、タイミングユニットによって内容が異なります。表 225 はタイマごとの 8 つの利用可能なオプションを示します。CMPx は、カウンタリセットから比較一致までのブランキングを指し、Tx2 は、HRTIM_SETx2 および HRTIM_RSTx2 レジスタで定義される、タイミングユニット TIMx 出力 2 の波形を指します。たとえば、タイマ B（TIMFLTR6）はタイマ C の出力 2 の波形となります。

表 225. タイマごとのフィルタリング信号のマッピング

転送元		タイマ A				タイマ B				タイマ C				タイマ D				タイマ E				タイマ F			
		CMP1	CMP2	CMP4	TA2	CMP1	CMP2	CMP4	TB2	CMP1	CMP2	CMP4	TC2	CMP1	CMP2	CMP4	TD2	CMP1	CMP2	CMP4	TE2	CMP1	CMP2	CMP4	TF2
転送先	タイマ A	-				1	-	2	3	4	-	5	-	7	-	-	-	-	8	-	-	6	-	-	-
	タイマ B	1	-	2	3	-				4	5	-	-	-	7	-	-	8	-	-	-	-	6	-	-
	タイマ C	-	1	-	-	2	-	3	-	-				5	-	6	7	-	-	8	-	4	-	-	-
	タイマ D	1	-	-	-	-	2	-	-	3	4	-	5	-				6	-	7	-	-	-	8	-
	タイマ E	-	1	-	-	2	-	-	-	3	-	-	-	6	-	7	8	-				-	-	4	5
	タイマ F	-	-	1	-	-	2	-	-	-	-	3	-	-	4	5	-	6	-	7	8	-			

図 221 および 図 222 に、通常および延期モードでの、すべてのエッジ対応とレベル対応に対する外部イベントのブランキングの例を示します。

図 221. エッジ対応トリガによる外部トリガブランキング

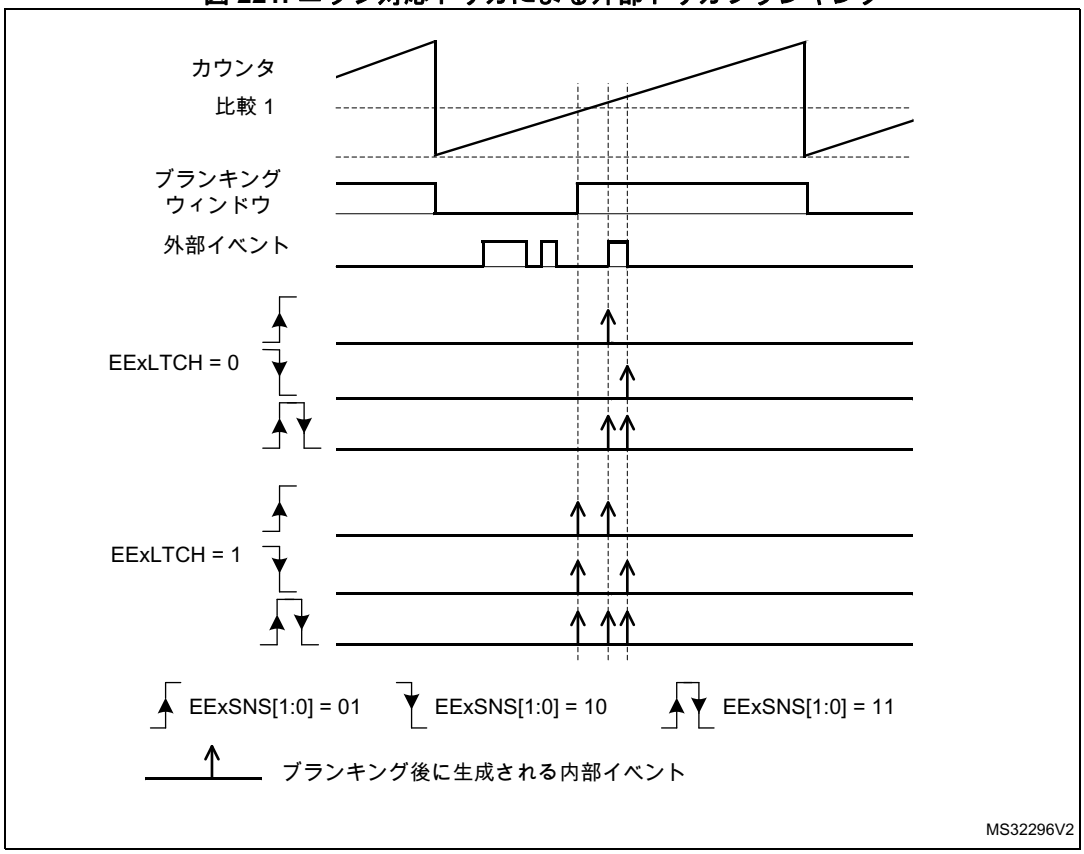
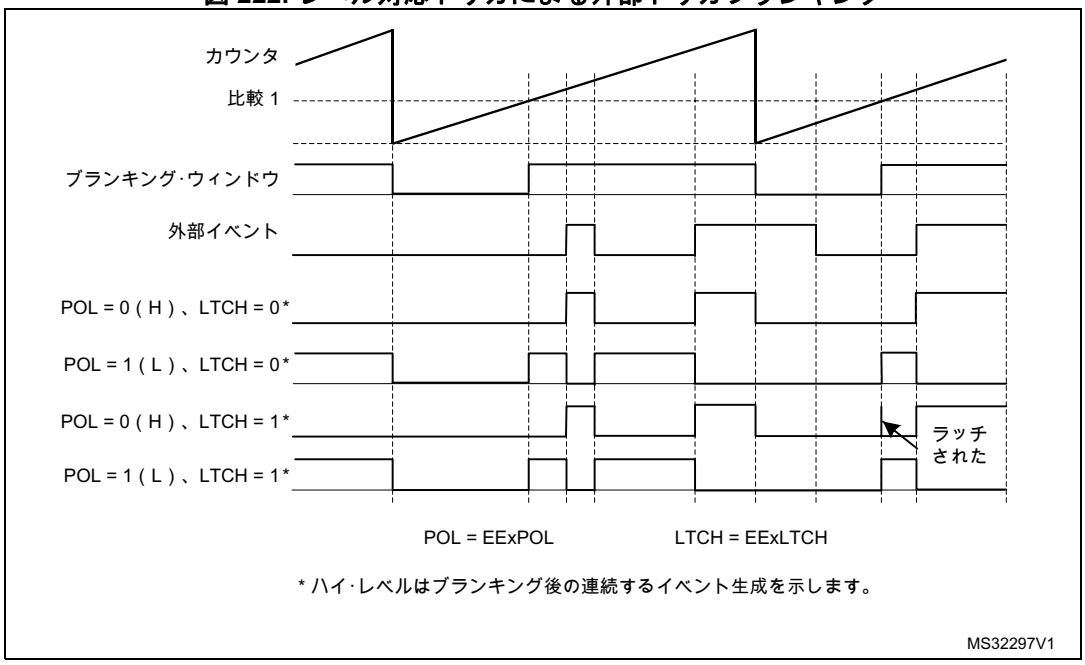


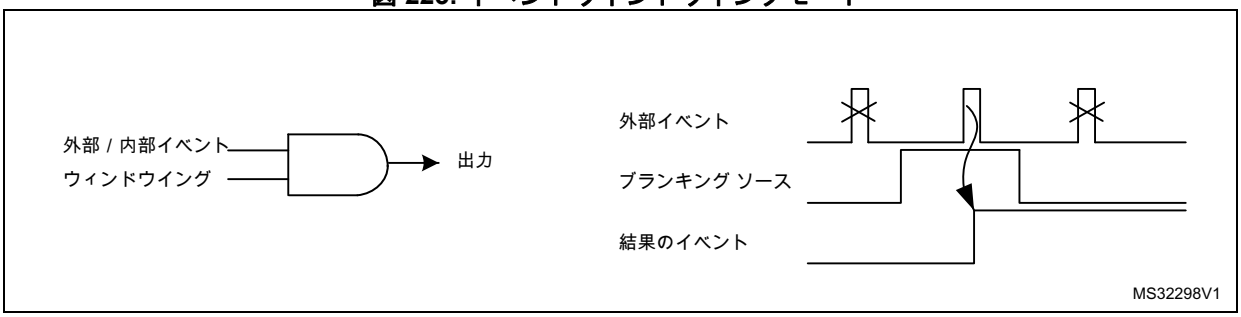
図 222. レベル対応トリガによる外部トリガブランキング



ウィンドウイングモード

イベントウィンドウイングモードでは、イベントは与えられた時間ウィンドウ内で発生した場合のみ、考慮され、そうでない場合は無視されます。このモードは、1101～1111 の範囲の EExFLTR[3:0] に対してアクティブになります。

図 223. イベントウインドウイングモード



EEFxR1 および EEFxR2 レジスタの EExLTCH ビットが 1 にセットされた場合、信号をラッチすることができます。この場合、イベントがウィンドウ内で発生した場合、受け付けられますが、そのイベントはウィンドウの終りまで遅延されます。

- EExLTCH ビットがリセットされていて、信号がウィンドウ内で発生した場合は、それは直接、パススルーされます。
- EExLTCH ビットがリセットされていて、信号が発生しない場合は、ウィンドウの終りにタイムアウトイベントが生成されます。

ウインドウイングモードの使用例は同期信号のフィルタリングです。期待される同期イベントが欠如したとき（たとえば、コンバータの起動中）、タイムアウトを発生させてデフォルトの同期イベントを強制実行できます。

各外部イベントのウィンドウイングのために 3 つのソースがあり、以下のようにコーディングされます。

- 1101 および 1110 : ウィンドウイングはカウンタリセットから比較一致まで続きます（それぞれ比較 2 と比較 3 に対応）。アップ／ダウンモード（UDM ビットを 1 に設定）では、カウンタリセットイベントは ROM [1:0] ビットの設定に従って定義されます。
- 1111 : ウィンドウイングは別のタイミングユニットに関係しており、そのカウンタリセットからその比較 2 の 一致まで続きます。そのソースはビットの説明において TIMWIN として説明されており、表 226 に示されています。例として、タイマ B での外部イベントを、タイマ A のカウンタリセットから始まりタイマ A の比較 2 までのウィンドウによってフィルタリングすることができます。

表 226. タイマごとのウィンドウイング信号のマッピング（EEFLTR[3:0] = 1111）

転送先	タイマ A	タイマ B	タイマ C	タイマ D	タイマ E	タイマ F
TIMWIN（ソース）	タイマ B CMP2	タイマ A CMP2	タイマ D CMP2	タイマ C CMP2	タイマ F CMP2	タイマ E CMP2

注： **外部イベントが高速モードでプログラムされている場合、タイムアウトイベントの生成はサポートされません。**

図 224 および 図 225 に、さまざまなエッジ対応およびレベル対応に対して、また EExLTCH ビットの設定に応じて、イベントがどのように生成されるかが示されています。タイムアウトイベントは、明確にするために、特に注記しています。

図 224. エッジ対応トリガによる外部トリガウィンドウイング

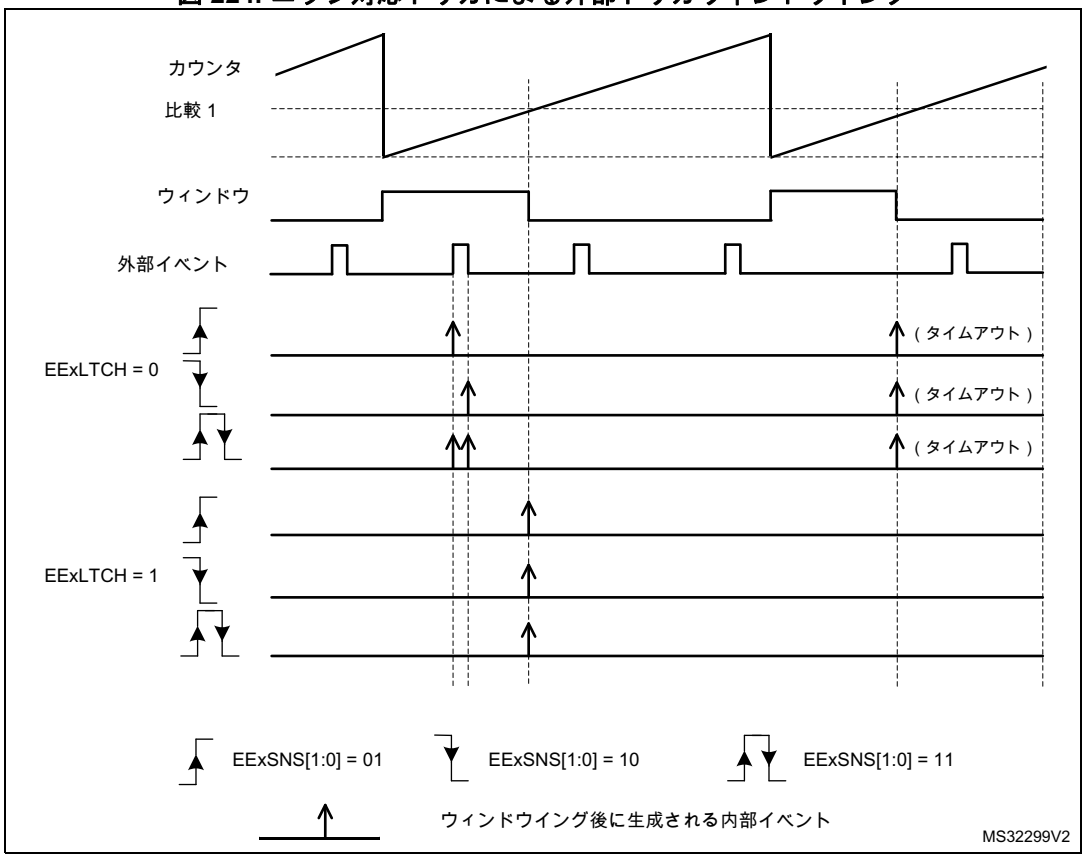
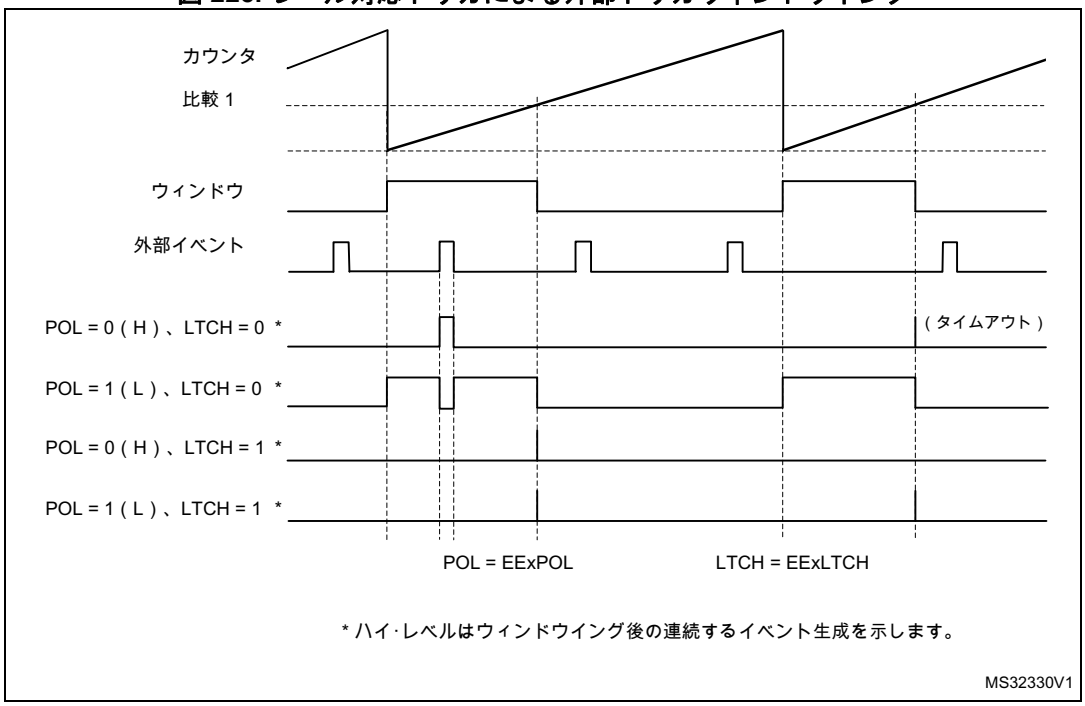


図 225. レベル対応トリガによる外部トリガウィンドウイング

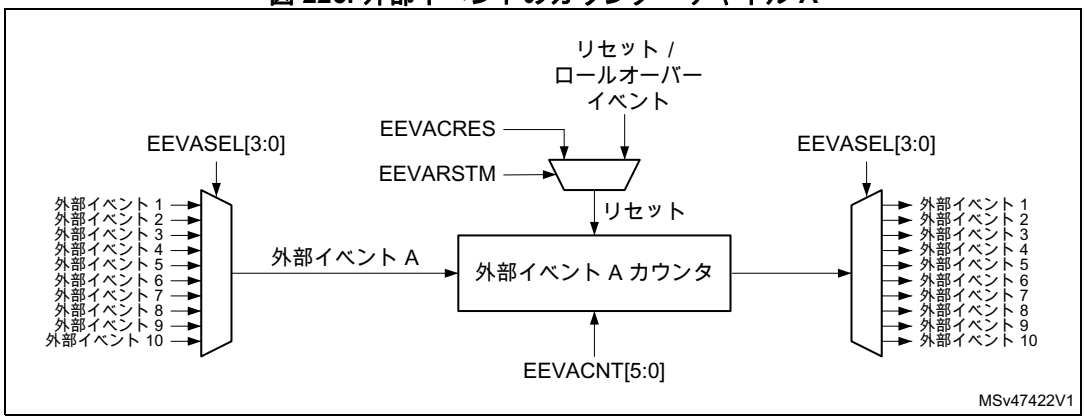


外部イベントのカウンタ

また、各タイミングユニットは、一般的にバレー（谷）スキップ実装のために、フィルタリングユニットに続く外部イベントカウンタも備えています。

図 226 に示すように、この回路により、フィルタリングされた 10 個の外部イベントのいずれかをフィルタリングできます。

図 226. 外部イベントのカウンタ – チャネル A



このカウンタを有効にするには、HRTIM_EEFxR3 レジスタの EEVACE ビットを使用します。このモードは、エッジ対応の外部イベント（EEASNS[1:0] ビット = 01、10、または 11）に対してのみ有効です。

外部イベントは、アクティブエッジの数が（EEVACNT[5:0] +1）でプログラムされた値以上である場合にのみタイマに伝播されます。

以下の 2 つの動作モードがあります。

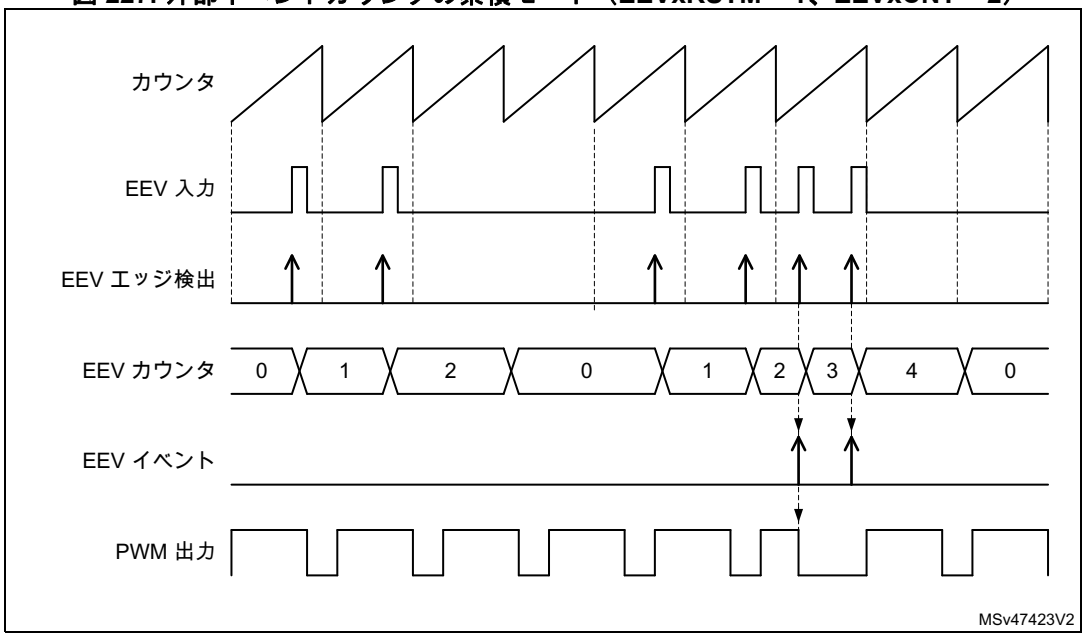
- EEVARSTM ビットがリセットされていると、リセット／ロールオーバーイベントごとに外部イベントカウンタがリセットされます。外部イベントは、特定の PWM 周期内に複数回出現する場合にのみアクティブになります。
- EEVARSTM/ ビットがセットされていると、外部イベントカウンタは、最後の PWM 周期中にイベントが出現しなかった場合にのみ、リセットされます。これは累積モードであり、以下の図 227 に示すように、イベントは複数の PWM 周期中に少なくとも 1 回発生する必要があります。

カウンタ値をプログラムした後、外部イベントカウンタを有効にする必要があります（EEVACNT[5:0] ビットを書き込んだ後、EEVACE ビットを設定する必要があります）。

カウンタが有効になると、EEVACNT[5:0] ビットは動作中にいつでも変更できます。新しい値は、EEVARSTM ビットのプログラミングに従って、またはソフトウェアリセット（EEVACRES ビットのセット）後に、次のリセット/ロールオーバーイベントで考慮されます。

EEVACE ビットがセットされた後は、EEVASEL[3:0] ビットは修正してはいけません。

図 227. 外部イベントカウンタの累積モード（EEVxRSTM = 1、EEVxCNT = 2）



27.3.10 遅延保護

HRTIM は、共振コンバータに典型的ですが、アクティブパルスが完了した後、またはプッシュプル周期が完了した後に PWM 出力を遅延方式で停止する必要があるときのために、特定の保護スキームを備えています。これらの機能は HRTIM_OUTxR レジスタの DLYPRTEN ビットで有効になり、特定の外部イベントチャネルを使用しています。

遅延アイドル

このモードでは、保護が有効化される前にアクティブパルスが完了します。選択された外部イベントは、（HRTIM_RSTx1R または HRTIM_RSTx2R で出力リセットイベントによって定義された）アクティブパルスの終りに出力をアイドルモードに移行させます。

保護にトリガがかかると、アイドルモードは永続的に維持されますが、カウンタは出力が再び有効化されるまで動作し続けます。Tx1OEN および Tx2OEN ビットは、遅延アイドルへの移行の影響を受けません。遅延アイドルから出て動作を再開するには、Tx1OEN および Tx2OEN ビットを 1 で上書きする必要があります。出力の状態は、出力イネーブルコマンドの後、アクティブ状態への最初の遷移時に変わります。

注： **遅延アイドルモードは、アクティブパルスが完了する前には、モードに入った直後に終了することはできません。実行モードを再開する前に出力がアイドル状態にあることを確認することが必須です。これは、たとえば、次の周期まで待つことによって、あるいは TIMxISR レジスタの O1CPY および／または O2CPY ステータスビットを調査することによって、行うことができます。**

遅延アイドルモードは単一出力（DLYPRT[2:0] = x00 または x01）または両出力（DLYPRT[2:0] = x10）に適用できます。

遅延アイドルモードへの移行に反応して割込みまたは DMA リクエストを生成することができます。出力上のアクティブパルスの終了とは独立に、外部イベントが着信するとすぐに、HRTIM_TIMxISR の DLYPRT フラグがセットされます。

遅延アイドルモードにトリガがかかると、出力状態は HRTIM_TIMxISR の O1STAT および O2STAT を使用して決定できます。遅延アイドルが単一の出力に適用されている場合でも、両方のステータス

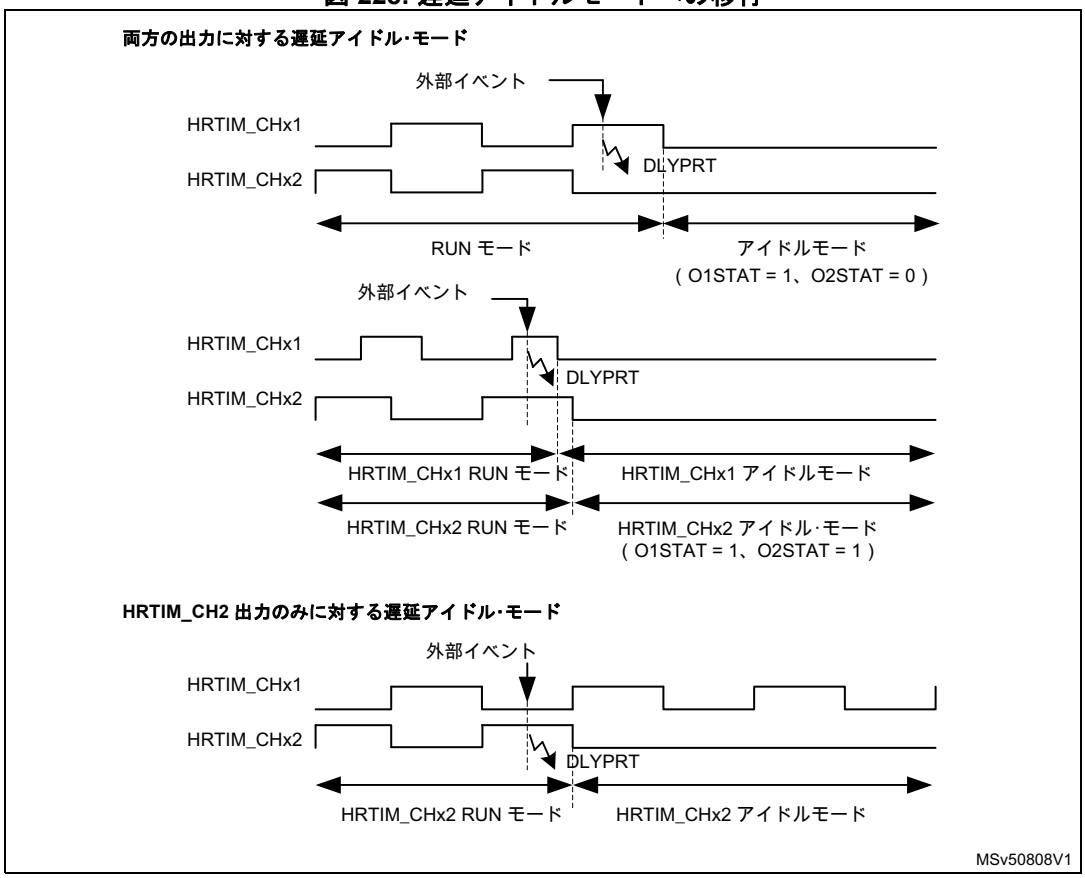
ビットは更新されます。プッシュプルモードが有効になっているとき、HRTIM_TIMxISR の IPPSTAT フラグは、どの周期内で遅延保護リクエストが発生したかを示します。

このモードは、タイマ動作モードが何（通常、プッシュプル、デッドタイム）であれ、使用できます。これは次の 2 つの外部イベントのみで使用できます。

- タイマ A、B、および C に対する hrtim_eev6 および hrtim_eev7
- タイマ D、E、および F に対する hrtim_eev8 および hrtim_eev9

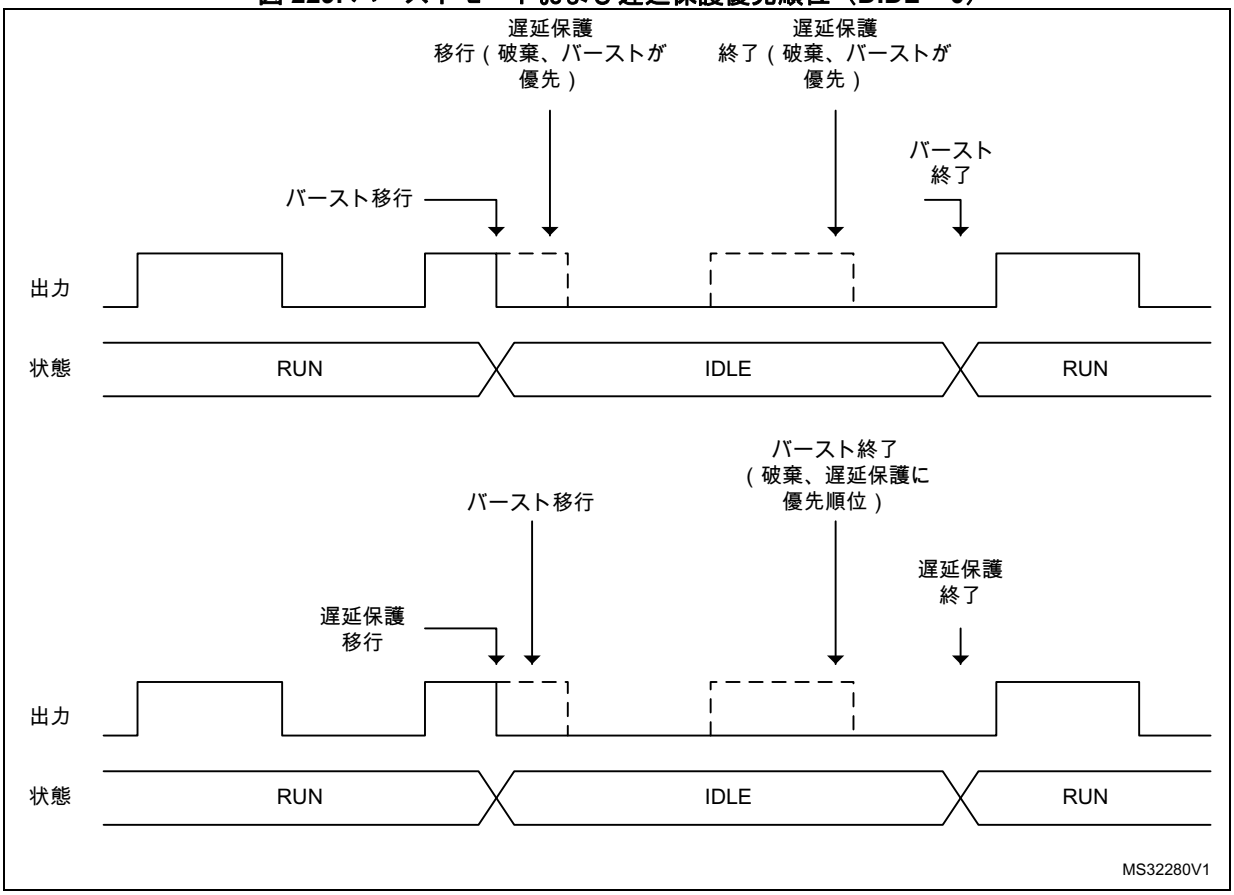
遅延保護モードは、カウンタが有効になっているときのみ（TxCEN ビットのセット）、トリガできます。TxEN ビットがリセットされても、TxyOEN ビットがセットされるまでアクティブのままです。

図 228. 遅延アイドルモードへの移行



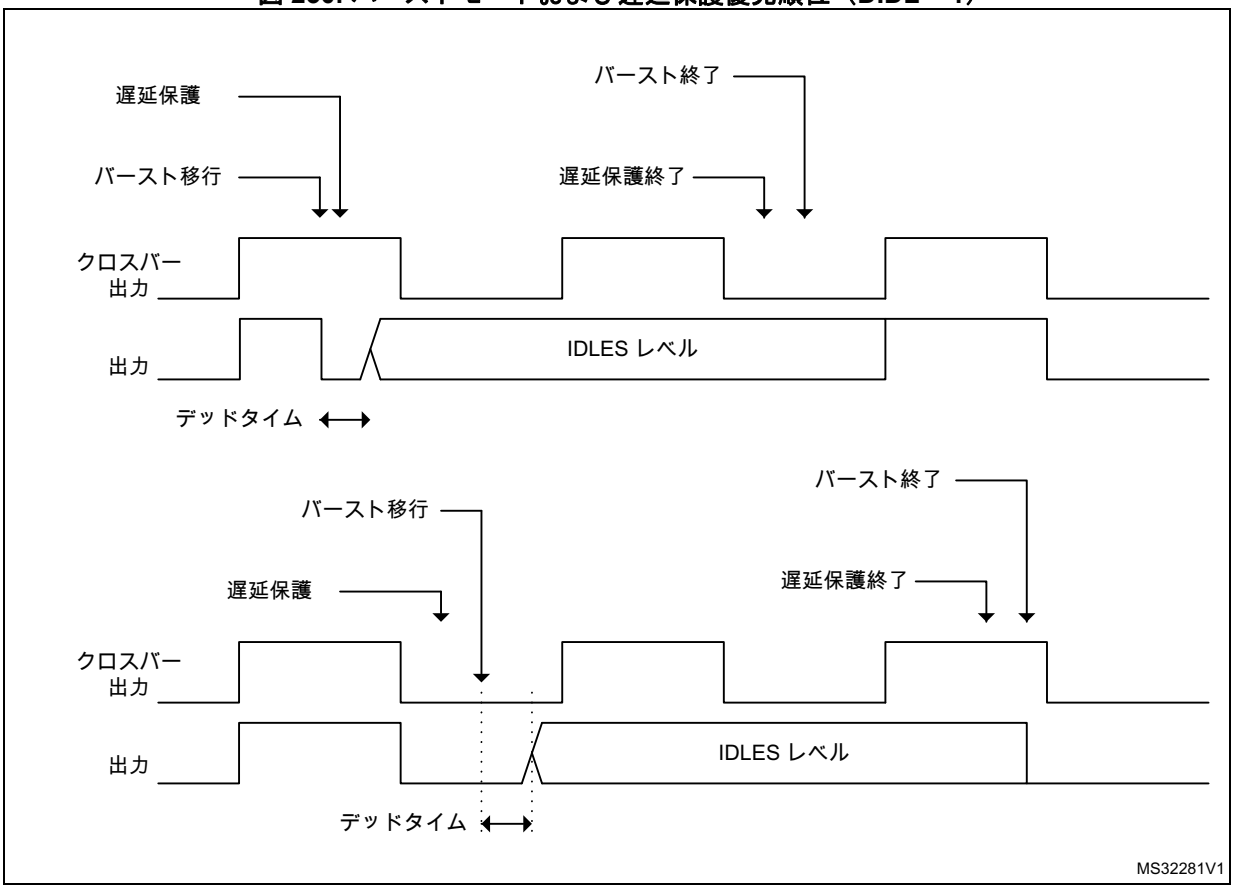
遅延アイドルモードはバーストモードよりも高い優先順位を持ちます。いったん遅延アイドル保護がトリガされるとバーストモードの終了リクエストは破棄されます。反対に、バーストモードがアクティブなときに、遅延保護を終了した場合、バーストモードは正常に再開され、バーストモードが終了するまで、出力はアイドル状態に保持されます。図 229 に、これらのさまざまなシナリオの概要を示します。

図 229. バーストモードおよび遅延保護優先順位（DIDL = 0）



遅延バーストモードへの移行が有効になっているとき（DIDL ビットのセット）、以下の図 230 に示すように、同じ優先順位が適用されます。

図 230. バーストモードおよび遅延保護優先順位（DIDL = 1）



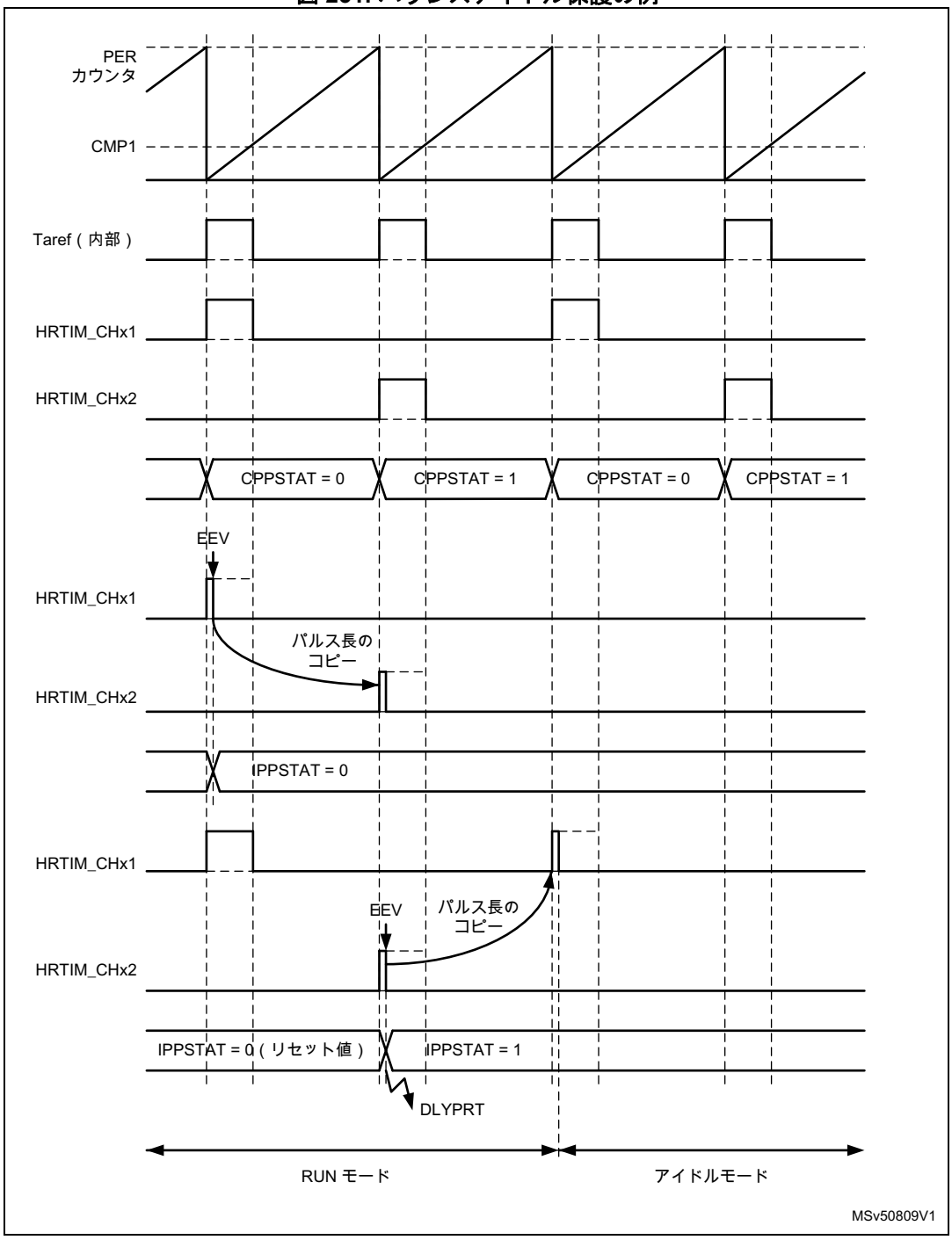
バランスアイドル

バランスアイドルはプッシュプルモードでのみ利用可能で、これにより、アクティブパルスの一方の幅が保護によって短くされたとき、2 つの出力のパルス幅をバランスさせることができます。プログラムされた幅より早く終了されたパルス幅がオルタネート出力にコピーされ、その後、2 つの出力はアイドル状態に入り、ソフトウェアによって通常動作が再開されるまで続きます。このモードは、HRTIM_OUTxR レジスタの DLYPRT[2:0] ビットフィールドに x11 を書き込むことによって有効になります。

このモードは次の 2 つの外部イベントのみで使用できます。

- タイマ A、B、および C に対する hrtim_eev6 および hrtim_eev7
- タイマ D、E、および F に対する hrtim_eev8 および hrtim_eev9

図 231. バランスアイドル保護の例



バランスアイドルモードが有効になっているとき、選択された外部イベントは、比較 4 アクティブレジスタ（この値はユーザにはアクセス不可）へのカウンタ値の取得をトリガします。短縮されたパルスが繰り返されるように、プッシュプルはさらに 1 周期の間、維持されます。通常の出力セットイベントが維持されている間に、新しい出力リセットイベントが生成されます。

その後、アイドルモードに入り、出力は、HRTIM_OUTxR レジスタの IDLESx ビットで定義されたレベルになります。バランスアイドルモードへの移行は DLYPRT フラグによって示され、一方、IPPSTAT

フラグはどの周期中に外部イベントが発生したかを示し、短縮されたパルスの順序を決定します（A1 から A2、または逆）。

タイマの動作は中断されません（カウンタは実行を続けます）。

バランスアイドルモードを有効にするには、以下の初期化が必要です。

- 連続モードでタイマを動作（CONT = 1）
- プッシュプルモードは有効です。
- HRTIM_CMP4xR を 0 にセットし、内容をアクティブレジスタに転送する必要があります（たとえば、ソフトウェア更新を強制することによって）。
- DELCMP4[1:0] ビットフィールドを 00 にセットする必要があります（自動遅延モードを無効化）。
- DLYPRT[2:0] = x11（遅延保護を有効化）

注： **バランスアイドル動作中には、HRTIM_CMP4xR レジスタへは書込みを行わないで ください。CMP4 イベントは予約済みであり、別の目的に使用することはできません。**

バランスアイドルモードでは、複数の外部イベントや出力リセットの原因になるソフトウェアによるリセットイベントは避けることを推奨します。バランスアイドルリクエストの前に同じ周期内でそのようなイベントを受信すると、出力パルスがアンバランスになります（最初のパルス長は外部イベントまたはソフトウェアリセットによって定義されますが、2 番目のパルスはバランスアイドルモードへの移行によって定義されます）。

バランスアイドルモードで処理できる最小のパルス幅は 4 f_{HRTIM} クロック周期（CKPSC[2:0] = 0 の場合は 0x80、CKPSC[2:0] = 1 の場合は 0x40、CKPSC[2:0] = 2 の場合は 0x20、・・・）です。

カウンタがこの最小値に達する前にキャプチャが起きると、現在のパルスは、二次出力にコピーされる前に、最大 f_{HRTIM} クロックの 4 周期分まで拡張されます。どの場合でも、パルス幅は常にバランスがとられます。

Tx1OEN および Tx2OEN ビットは、バランスアイドルへの移行の影響を受けません。バランスアイドルから出て動作を再開するには、Tx1OEN および Tx2OEN ビットを同時に 1 で上書きする必要があります。出力の状態は、出力の有効化の後、アクティブへの最初の遷移時に変わります。

遅延アイドルへの移行と同じように動作を再開することができます。たとえば、出力 1 がアクティブのときに外部イベントが到着した場合（出力 2 パルスの後、遅延アイドルが有効）、出力 1 の再開シーケンスを最初に開始できます。こうするには、HRTIM_TIMxISR レジスタの CPPSTAT ビットを調査する必要があります。上記の例を使用して（IPPSTAT フラグ = 0）、CPPSTAT ビットが 0 のとき、動作は再開されます。

特定の再開シーケンスを行うために、CPPSTAT をポーリングして、どの出力が最初にアクティブになるかを知ることができます。こうすることで、たとえば、アイドル移行シーケンスと同じシーケンスで再開することができます。つまり、出力 1 がアクティブの間に外部イベントが来た場合、出力 1 がアクティブ（CPPSTAT = 0）のときに再開シーケンスが開始されます。

注： **パルスバランスシーケンスが進行中のときは、バランスアイドルモードは無効にしてはいけません。CMP4 フラグがセットされて、シーケンスが完了したことを示すまで待って、DLYPRTEN ビットをリセットする必要があります。**

バランスアイドル保護モードは、カウンタが有効になっているときのみ（TxCEN ビットのセット）、トリガできます。TxCEN ビットがリセットされても、TxyOEN ビットがセットされるまでアクティブのままです。

以下の条件の下では、バランスアイドルはバーストモードと共に使用できます。

- TxBM ビットをリセットしなければならない（バースト中にカウンタクロックが維持される。セクション 27.3.15： を参照）。
- 出力がバーストアイドル状態になっているときにバランスアイドル保護をトリガしてはならない。

バランスアイドルモードはバーストモードよりも高い優先順位を持ちます。いったんバランスアイドル保護がトリガされるとバーストモードの終了リクエストは破棄されます。反対に、バーストモードがアクティブなときに、遅延保護を終了した場合、バーストモードは正常に再開されます。

注： **出力の状態がアイドルモードに固定されていても、遅延保護の後のアイドル期間中に下記のような複数のイベントが補助出力上に生成されます（セクション 27.3.18：参照）。**
- 出力のセット／リセット割込みまたは DMA リクエスト
- 出力信号に基づく外部イベントのフィルタリング
- セット／リセットにトリガされたキャプチャイベント

バランスアイドルの自動再開

バランスアイドルモードは、トリガ後に動作を自動的に再開するように設定できます。

短縮されたパルスがオルタネート出力にコピーされると、パルス幅が元の値にリセットされ、タイマが動作を再開します。2 つの出力は RUN モードのままです。

このモードを有効にするには、HRTIM_OUTxR レジスタの BIAR ビットをセットします。

このモードは、HRTIM_PERxR の周期が f_{HRTIM} クロックの 6 周期（CKPSC[2:0] = 0 の場合は 0xC0、CKPSC[2:0] = 1 の場合は 0x60、CKPSC[2:0] = 2 の場合は 0x30、・・・）より大きい場合にのみ、使用する必要があります。

注： **このビットは、DLYPRT[2:0] = 011 または 111 の場合にのみ、意味があります。そうでない場合は、無視されます。**

注： **バランスアイドルの自動再開モードでは、IDLES 状態を非アクティブに設定する必要があります。**

27.3.11 レジスタのプリロードおよび更新管理

大部分の HRTIM レジスタはバッファされ、必要な場合、プリロードできます。典型的には、これによって、アクティブなイベントに同期しないレジスタ更新（セット／リセット）によって波形が変更されるのを防ぐことができます。

プリロードモードが有効になっているとき、アクセスされるレジスタはシャドウレジスタです。ソフトウェアによる更新リクエスト、またはイベントに同期した更新リクエストの後に、その内容がアクティブレジスタに転送されます。

デフォルトでは、HRTIM_MCR および HRTIM_TIMxCR レジスタの PREEN ビットはリセットされ、レジスタはプリロードされません。どの書込みも直接、アクティブレジスタを更新します。タイマが動作中で、プリロードが有効になっていたときに PREEN ビットがリセットされた場合、プリロードレジスタの内容は直接、アクティブレジスタに転送されます。

各タイミングユニットおよびマスタタイマには独自の PREEN ビットがあります。PRREN がセットされると、プリロードレジスタが有効になり、更新イベント時にのみ、アクティブレジスタに転送されます。

プリロード機能が必要なとき、タイマを初期化するには以下の 2 つのオプションがあります。

- タイマの初期化の一番最後に PREEN ビットを有効にし、（MCEN および TxCEN ビットをセットすることで）タイマを有効にする前にプリロードレジスタをアクティブレジスタに転送します。
- 初期化中の任意の時点で PREEN ビットを有効にし、開始直前にソフトウェアによる更新を強制します。

表 227 に、プリロード可能なレジスタを、利用できる更新イベントの概要と共に一覧表示します。

表 227. HRTIM のプリロード可能な制御レジスタおよび関連する更新ソース

タイマ	プリロード可能なレジスタ	プリロード イネーブル	更新ソース
マスタタイマ	HRTIM_DIER HRTIM_MPER HRTIM_MREP HRTIM_MCMP1R HRTIM_MCMP2R HRTIM_MCMP3R HRTIM_MCMP4R	HRTIM_MCR の PREEN ビット	ソフトウェア 繰り返しイベント バースト DMA イベント バースト DMA イベント後の繰り返しイベント
タイマ x x = A～F	HRTIM_TIMxDIER HRTIM_TIMxPER HRTIM_TIMxREP HRTIM_TIMxCMP1R HRTIM_TIMxCMP1CR HRTIM_TIMxCMP2R HRTIM_TIMxCMP3R HRTIM_TIMxCMP4R HRTIM_DTxR HRTIM_SETx1R HRTIM_RSTx1R HRTIM_SETx2R HRTIM_RSTx2R HRTIM_RSTxR	HRTIM_TIMxCR の PREEN ビット	ソフトウェア TIMx 繰り返しイベント TIMx リセットイベント バースト DMA イベント 他のタイマ（TIMy、マスタ）からの更新イベント バースト DMA イベント後の更新イベント 更新イネーブル入力 hrtim_upd_en[3:1] hrtim_upd_en[3:1] 入力による更新イベントに続く更新イネーブル入力後の更新イベント
HRTIM 共通	HRTIM_ADC1R HRTIM_ADC2R HRTIM_ADC3R HRTIM_ADC4R	選択したタイマで PREEN = 1 の場合、HRTIM_CR1 の ADxUSRC[2:0] ビットに応じた、TIMx タイマ または マスタ·タイマの更新	

マスタタイマには次の 4 つの更新オプションがあります。

1. ソフトウェア : HRTIM_CR2 の MSWU ビットに 1 を書き込むことによって、レジスタの即時更新を強制します。この場合、保留になっていたハードウェアの更新リクエストはキャンセルされます。
2. マスタカウンタがロールオーバーし、マスタ繰り返しカウンタが 0 に等しくなるときに行われる更新。これは、HRTIM_MCR の MREPU ビットがセットされているとき、有効になります。
3. バースト DMA が完了した後、行われる更新（詳細はセクション 27.3.23 : を参照）。これは、HRTIM_MCR の BRSTDMA[1:0] = 01 のとき、有効になります。MREPU=1 および BRSTDMA=01 の両方をセットできます。
   注 : SWU ビットがセットされた場合（すなわち、強制更新モード）、更新はバーストシーケンスの終了直後に起こすことができます。SWU ビットがリセットされた場合、更新は、バーストシーケンスの終了後の次の更新イベント時に行われます。
4. バースト DMA 完了後にマスタカウンタがロールオーバーするときに行われる更新。これは、HRTIM_MCR の BRSTDMA[1:0] = 10 のとき、有効になります。

割込みまたは DMA リクエストはマスタ更新イベントによって生成できます。

各タイマ（TIMA～F）は、以下のように更新させることもできます。

- ソフトウェアによる：HRTIM_CR2 の TxSWU ビットに 1 を書き込むことによって、レジスタの即時更新を強制します。この場合、保留になっていたハードウェアの更新リクエストはキャンセルされます。
- カウンタがロールオーバーし、繰り返しカウンタが 0 に等しくなるときに行われる更新。これは、HRTIM_TIMxCR の TxREPU ビットがセットされているとき、有効になります。
- カウンタがリセットされた、または連続モードでロールオーバーしたときに行われる更新。これは、HRTIM_TIMxCR の TxRSTU ビットがセットされているとき、有効になります。これは、たとえば、シングルショットモードで動作しているタイマに使用されます。
- バースト DMA が完了した後、行われる更新。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0001 のとき、有効になります。
- バースト DMA の完了後の更新イベント時に行われる更新（イベントは TxRSTU、TxREPU、MSTU 、または TxU により有効化できます）。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0010 のとき、有効になります。
- hrtim_upd_en[3:1] によるリクエストを受信したときに行われる更新。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0011、0100、0101 のとき、有効になります。
- hrtim_upd_en[3:1] によるリクエスト後の更新イベント時に行われる更新（イベントは TxRSTU、TxREPU、MSTU、または TxU により有効化できます）。これは、HRTIM_TIMxCR の UPDGAT[3:0] = 0110、0111、1000 のとき、有効になります。
- 他のタイマまたはマスタ更新と同期して行われる更新（たとえば、TIMA は TIMB と同時に更新できます）。これは、複数のタイマを必要とするコンバータ用に使用され、HRTIM_TIMxCR レジスタの MSTU および TxU ビットのセットによって有効になります。

更新イネーブル入力 hrtim_upd_en[3:1] によって、更新イベントを汎用タイマから来るオンチップイベントと同期させることができます。これらの入力は立ち上がりエッジ対応です。

表 210 に、更新イネーブル入力とオンチップソースとの接続を示します。

これによって、低周波数の更新リクエストを高周波信号と同期させることができます（たとえば、100 Hz の速度で行う必要のある、100 kHz PWM のカウンタロールオーバー時の更新）。

注： **CKPSC[2:0] > 5 のとき、更新イベントはプリスケーラクロックに同期します。**

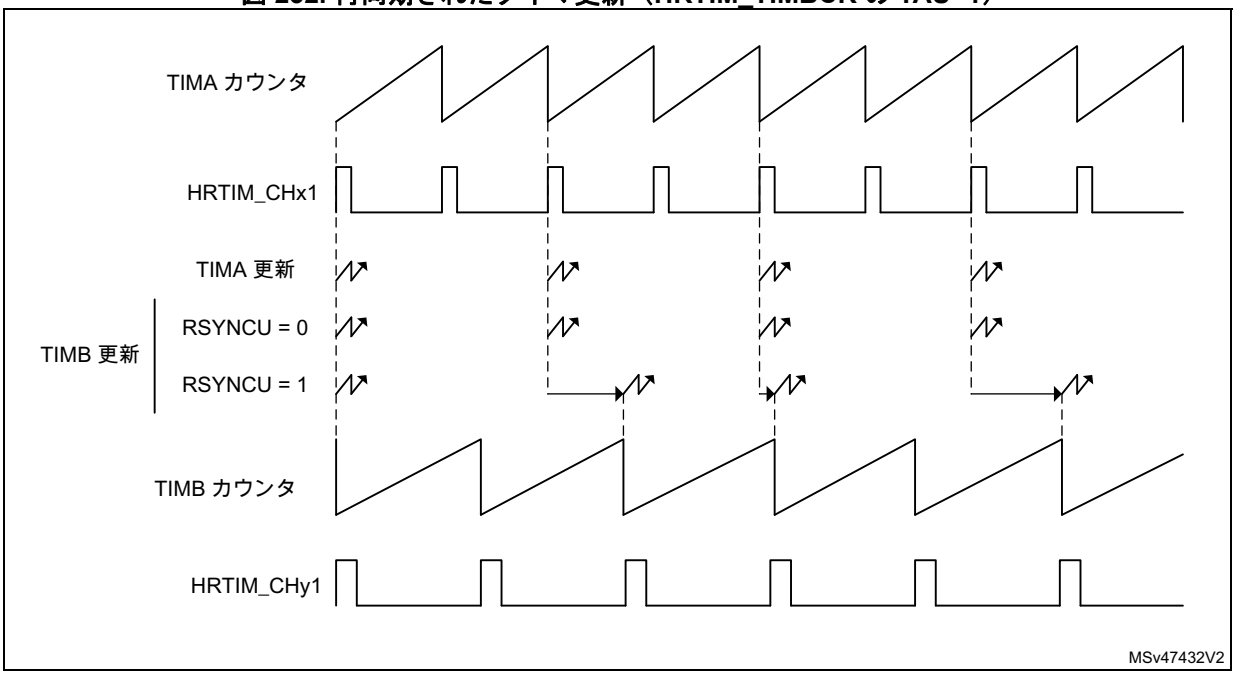
隣接するタイマからの更新（MSTU、TAU、TBU、TCU、TDU、TEU、TFU ビットがセットされている場合）またはソフトウェア更新（TxSWU ビット）からの更新は、すぐに考慮されるか、またはタイマのリセット／ロールオーバーイベントと再同期させることができます。これは、下記図 232 に示すように、HRTIM_TIMxCR レジスタの RSYNCU ビットによって行われます。

- RSYNCU = 0：隣接するタイマからの更新はただちに有効になります。
- RSYNCU = 1：隣接するタイマからの更新は、次のリセット／ロールオーバーイベントで有効になります。

RSYNCU ビットは、UPDGAT[3:0] = 0000 の場合にのみ、意味があります。そうでない場合は、無視されます。

割込みまたは DMA リクエストは Timx 更新イベントによって生成できます。

図 232. 再同期されたタイマ更新（HRTIM_TIMBCR の TAU=1）



選択された更新イベントが何であれ、HRTIM_CR1 レジスタの MUDIS および TxUDIS ビットによって、プリロードレジスタからアクティブレジスタへの転送を一時的に無効にすることができます。これによって、複数のタイマのいくつかのレジスタが変更できるようになります。これらのビットがリセットされると、通常の更新イベントが発生します。

MUDIS および TxUDIS ビットはすべて同じレジスタにグループ化されています。このため、複数のタイマの更新を（必ずしも同期する必要はないが）同時に無効にして再開することが可能になります。

以下の例は実際的な使用事例です。最初の電力コンバータはマスタ、TIMB、および TIMC によって制御されています。TIMB および TIMC はマスタタイマの繰り返しイベントによって同時に更新する必要があります。2 番目のコンバータは TIMA、TIMD、および TIME と並行に動作し、TIMD、TIME は TIMA の繰り返しイベントによって更新する必要があります。

最初のコンバータ

HRTIM_MCR で、MREPU ビットをセットします。マスタタイマのカウンタの繰り返し周期の終りで更新が起きます。HRTIM_TIMBCR および HRTIM_TIMCCR の MSTU ビットがセットされ、TIMB および TIMC タイマがマスタタイマによって同時に更新されます。

電力コンバータの設定値をソフトウェアで調節する必要があるとき、レジスタに書込みアクセスして値（たとえば、比較値）を更新する前に HRTIM_CR レジスタの MUDIS、TBUDIS、および TCUDIS ビットをセットする必要があります。この時点以降、ハードウェア更新リクエストは無視され、プリロードレジスタは、アクティブレジスタに転送される恐れなく、アクセスできます。ソフトウェア処理が終わると、MUDIS、TBUDIS、および TCUDIS ビットはリセットする必要があります。プリロードからアクティブレジスタへの転送は、マスタ繰り返しイベントが発生するとすぐに行われます。

2 番目のコンバータ

HRTIM_TIMACR で、TAREPU ビットをセットします。タイマ A カウンタの繰り返し周期の終りで更新が起きます。HRTIM_TIMDCR および HRTIM_TIMECR の TAU ビットがセットされ、TIMD および TIME タイマがタイマ A によって同時に更新されます。

電力コンバータの設定値をソフトウェアで調節する必要があるとき、レジスタに書込みアクセスして値（たとえば、比較値）を更新する前に HRTIM_CR レジスタの TAUDIS、TDUDIS、および TEUDIS

ビットをセットする必要があります。この時点以降、ハードウェア更新リクエストは無視され、プリロードレジスタは、アクティブレジスタに転送される恐れなく、アクセスできます。ソフトウェア処理が終わると、TAUDIS、TDUDIS、および TEUDIS ビットはリセットできます。プリロードからアクティブレジスタへの転送は、タイマ A の繰り返しイベントが発生するとすぐに行われます。

27.3.12 「より大きい」比較を備えた PWM モード

CMP1 および CMP3 レジスタで生成された PWM 信号に対して、特定の遅延なし更新モードが利用可能です。これによって、現在の PWM 周期の完了を待たずに、PWM サイクル内でできるだけ早く新しいデューティサイクル値を適用できます。このため、ソフトウェア制御ループの全体的な遅延時間が短縮されます。下記の図 233 に示すように、これによって最終的に以下のことが可能になります。

- 新しい比較値が書き込まれる時点で、新しい比較値が現在のカウンタ値より低く、現在の比較値がカウンタ値より高い場合に、出力の早期オフ。
- 新しい比較値が書き込まれる時点で、新しい比較値がカウンタ値より高く、現在の比較値がカウンタ値より高い場合に、出力の早期オン、出力の再有効化。

新しい比較値と現在の比較値が両方ともカウンタ値より低い場合、出力信号は変更されません。

この機能は、CMP1 または CMP3 RESET イベントに対してのみ利用でき、HRTIM_TIMxCR2 レジスタの GTCMP1 および GTCMP3 イネーブルビットを使用して有効にできます。

対応する GTCMPx ビットが設定されている場合、PREEN ビット値に関係なく、プリロードメカニズムは比較レジスタに対して非アクティブになります。このモードは、プリロードからアクティブなレジスタへの転送を待たずに、新しい比較値の書込み後できるだけ早く新しい値を考慮に入れさせることを目的としています。

これらのビットは、比較 1 と 比較 3 の動作モードを下記のように定義しています。

- GTCMPx = 0：比較 x イベントは、カウンタが比較値と等しいときに生成されます（比較一致モード）。比較値が動作中に変更された場合、比較イベントは生成されないことがあります。
- GTCMPx = 1：比較 x イベントは、カウンタが比較値より大きいときに生成されます。比較値が動作中に変更された場合、新しい比較値は現在のカウンタ値と比較され、SET または RESET の出力が生成されます。

「より大きい」比較モードでは、比較結果に応じてクロスバーの動作が異なってきます。CMP1 イベントが出力の RESET を行っていると考えてみましょう。新しい比較値が書き込まれるとき、次の 2 つの状況が考えられます。

- 新しい比較値がカウンタ値より小さい場合、RESET イベントが発行され、最終的に早期のターンオフを引き起こせます。
- 新しい比較値がカウンタ値より大きい場合、SET イベントが生成され、カウンタが現在の比較値を超えたときに実際に RESET される前に出力値を再設定できます（早期ターンオン）。

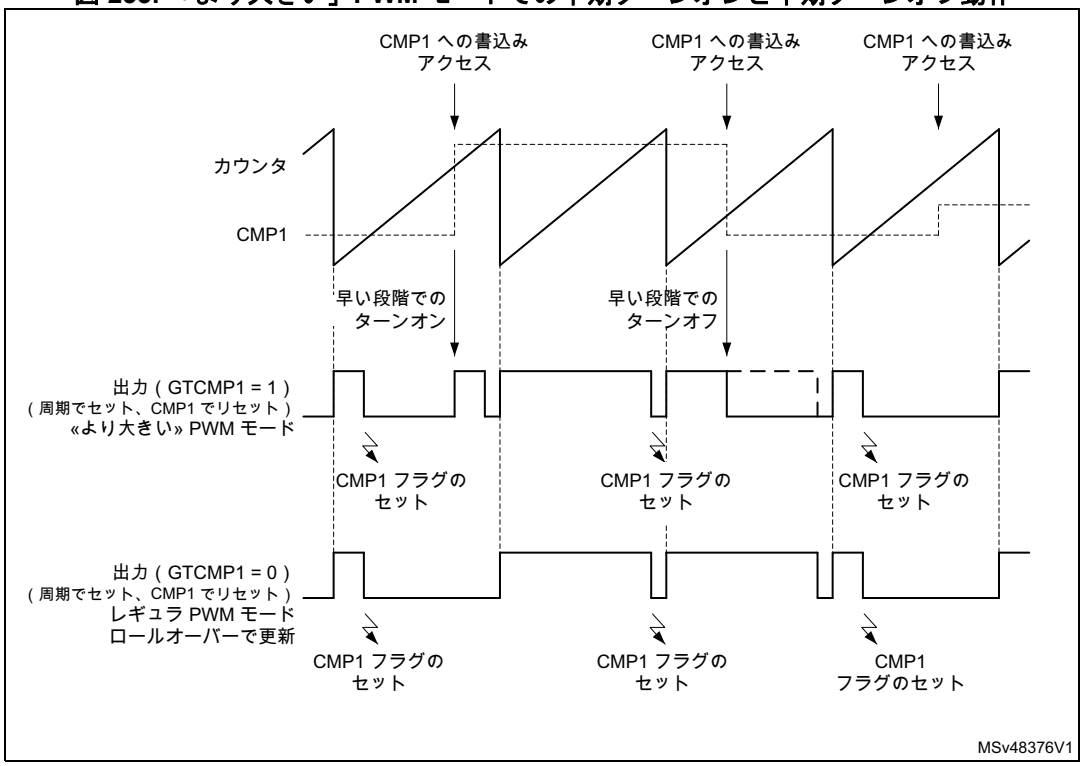
「より大きい」比較モードは、SET と RESET の両動作に対してサポートされています。

「より大きい」比較モードは、以下の設定に対してのみ使用する必要があります。

1. 固定周波数の設定では、周期イベントが出力セットをトリガし、「より大きい」比較が出力リセットをトリガする必要があります（または逆に、「より大きい」比較がセットをトリガする場合、周期はリセットをトリガする必要があります）。
2. 可変周波数の設定の場合、カウンタリセットソースとして選択したイベントは、タイマ出力のセットまたはリセットソースとしても（「より大きい」比較イベントと反対方向に）選択する必要があります。

注： **CMP1 および／または CMP3 モードがハーフモードおよびインタリーブモードでハードウェアによって制御されているときは、「より大きい」モードは使用してはなりません。**

図 233. 「より大きい」PWM モードでの早期ターンオンと早期ターンオフ動作



即時更新モードは、レジスタの書込みと同時に、プリロードレジスタの内容がアクティブレジスタに転送されることを意味します。GTCMP1 および／または GTCMP3 ビットがセットされると、PREEN ビットの値に関係なく、それぞれのプリロードメカニズム（HRTIM_TIMxCMP1 および／または HRTIM_TIMxCMP3 レジスタに対して）が無効になります。

注： **比較割込みフラグ（HRTIM_TIMxISR の CMP1 および CMP3）は、図 233 に示すように、遅いターンオンと早いターンオフの場合には生成されません。**

注： **「より大きい」比較は、同じ出力に対して CMP1 と CMP3 の両方で実行しないでください（GTCMP1 ビットと GTCMP3 ビットを同時にセットしないでください）。**

27.3.13 複数タイマ内またはそれらをまたがったイベントの伝播

HRTIM は、複数のタイミングユニット（マスタタイマを含む）にまたがってイベントをカスケード伝達または共有させる多くの可能性を提供し、そのモジュラーアーキテクチャの利点をフルに活用します。これらは複数の同期出力を必要とするコンバータ用の重要な機能です。

このセクションでは、さまざまなオプションを概説し、イベントが HRTIM 内で伝播されるかどうか、またどのように伝播されるかを具体的に述べます。

マスタタイマの更新によってトリガされる TIMx の更新

表 228 に列挙されたソースはマスタタイマの更新を生成します。表は、ソースイベントが、どの TIMx タイミングユニットにも同時に更新のトリガをかけるのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの MSTU ビットがセットされていることです。

表 228. マスタタイマ更新イベントの伝播

転送元	状態	伝播	コメント
バースト DMA 終了	BRSTDMA[1:0] = 01	不可	TIMxCR で行われる必要あり（UPDGAT[3:0] = 0001）
バースト DMA 終了後のロールオーバーイベント	BRSTDMA[1:0] = 10	可能	-
カウンタのロールオーバーに起因する繰り返しイベント	MREPU = 1	可能	-
カウンタのリセットに起因する繰り返しイベント（HRTIM_SCIN またはソフトウェアから）		不可	-
ソフトウェア更新	MSWU = 1	不可	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、同時更新に使用可能

TIMy の更新によってトリガされる TIMx の更新

表 229 に列挙されたソースは TIMy の更新を生成します。表は、与えられたイベントが、他のまたは複数の TIMx タイマに同時に更新のトリガをかけるのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの TyU ビットをセット（ソース = TIMy および転送先 = TIMx）

表 229. TIMx 更新イベントの伝播

転送元	状態	伝播	コメント
バースト DMA 終了	UPDGAT[3:0] = 0001	不可	直接、HRTIM_TIMxCR で行われる必要あり（UPDGAT[3:0] = 0001）
更新イネーブル入力 hrtim_upd_en[3:1] に起因する更新	UPDGAT[3:0] = 0011、0100、0101	不可	直接、HRTIM_TIMxCR で行われる必要あり（UPDGAT[3:0] = 0011、0100、0101）
マスタ更新	HRTIM_TIMyCR で、MSTU = 1	不可	HRTIM_TIMxCR の MSTU = 1 で行われる必要あり
別の TIMx 更新（TIMz>TIMy>TIMx）	HRTIM_TIMyCR で、TzU = 1 TIMxCR で、TyU=1	不可	HRTIM_TIMxCR の TzU = 1 で行われる必要あり HRTIM_TIMyCR で、TzU = 1
カウンタのロールオーバーに起因する繰り返しイベント	TyREPU = 1	可能	-
カウンタのリセットに起因する繰り返しイベント	TyREPU = 1	-	下記のカウンタリセットの場合を参照
カウンタのロールオーバー	TyRSTU = 1	可能	-
カウンタのソフトウェアリセット	HRTIM_CR2 で、TyRST = 1	不可	HRTIM_CR2 レジスタの更新で同時に行うことが可能
TIMz 比較に起因するカウンタリセット	HRTIM_RSTyR の TIMzCMPn	可能	-
外部イベントに起因するカウンタリセット	HRTIM_RSTyR の EXTEVNTn	可能	-

表 229. TIMx 更新イベントの伝播 （続き）

転送元	状態	伝播	コメント
マスタ比較または マスタ周期に起因する カウンタリセット	HRTIM_RSTyR の MSTCMPn または MSTPER	可能	-
TIMy 比較に起因する カウンタリセット	HRTIM_RSTyR の CMPn	可能	-
更新に起因する カウンタリセット	HRTIM_RSTyR の UPDT	不可	伝播はロックアップ状況を招く恐れあり（更新はリセットを引き起こし、それが更新を引き起こす）
HRTIM_SCIN に起因する カウンタリセット	HRTIM_TIMyCR の SYNCRSTy	不可	-
ソフトウェア更新	TySWU = 1	不可	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、 同時更新に使用可能

TIMx 更新を引き起こす TIMx カウンタリセット

表 230 はカウンタリセットのソースを挙げ、それらが更新を生成するのに使用できるかどうかを示しています。

動作条件：HRTIM_TIMxCR レジスタの TxRSTU ビット

表 230. 更新を生成できるリセットイベント

転送元	状態	伝播	コメント
カウンタのロールオーバー	-	可能	-
更新イベント	HRTIM_RSTxR の UPDT	不可	伝播はロックアップ状況を招く恐れあり（更新はリセットを引き起こし、それが更新を引き起こす）
外部イベント	HRTIM_RSTxR の EXTEVNTn	可能	-
TIMy 比較	HRTIM_RSTxR の TIMyCMPn	可能	-
マスタ比較	HRTIM_RSTxR の MSTCMPn	可能	-
マスタ周期	HRTIM_RSTxR の MSTPER	可能	-
比較 2 および 4	HRTIM_RSTxR の CMPn	可能	-
ソフトウェア	HRTIM_CR2 で、TxRST = 1	可能	-
HRTIM_SCIN	HRTIM_TIMxCR の SYNCRSTx	可能	-

TIMx カウンタのリセットを引き起こす TIMx 更新

表 231 は更新イベントのソースを挙げ、それらがカウンタリセットを生成するのに使用できるかどうかを示しています。

動作条件：HRTIM_RSTxR で UPDT ビットのセット

表 231. タイマのリセットのための更新イベントの伝播

転送元	状態	伝播	コメント
バースト DMA 終了	UPDGAT[3:0] = 0001	可能	-
更新イネーブル入力 hrtim_upd_en[3:1] に起因する更新	UPDGAT[3:0] = 0011、0100、0101	可能	-
バースト DMA 後のロールオーバーに起因するマスタ更新	HRTIM_TIMxCR で、MSTU = 1 HRTIM_MCR で、BRSTDMA[1:0] = 10	可能	-

表 231. タイマのリセットのための更新イベントの伝播 （続き）

転送元	状態	伝播	コメント
ロールオーバー後の繰り返しイベントに起因するマスタ更新	HRTIM_TIMxCR で、MSTU = 1 HRTIM_MCR で、MREPU = 1	可能	-
カウンタリセット後の繰り返しイベントに起因するマスタ更新（ソフトウェアまたは HRTIM_SCIN による）		不可	-
ソフトウェアトリガによるマスタタイマの更新	HRTIM_TIMxCR で、MSTU = 1 MSWU = 1 HRTIM_CR2 で	不可	すべてのソフトウェア更新ビット（TxSWU）は HRTIM_CR2 レジスタにグループ化されており、同時更新に使用可能
TIMy カウンタのロールオーバーに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyRSTU = 1	可能	-
TIMy 繰り返しイベントに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyREPU = 1	可能	-
外部イベントまたは TIMy 比較に起因する TIMy 更新（TIMy リセットを通して）	HRTIM_TIMxCR で、TyU = 1 HRTIM_TIMyCR で、TyRSTU = 1 EXTEVNTn または CMP4/2 （HRTIM_RSTyCR で）	可能	-
上記以外のソースに起因する TIMy 更新	HRTIM_TIMxCR で、TyU = 1	不可	-
ロールオーバー後の繰り返しイベント	HRTIM_TIMxCR で、TxREPU = 1	可能	-
カウンタのリセット後の繰り返しイベント		不可	-
タイマのリセット	HRTIM_TIMxCR で、TxRSTU = 1	不可	伝播はロックアップ状況を招く恐れあり（リセットは更新を引き起こし、それがリセットを引き起こす）
ソフトウェア	HRTIM_CR2 の TxSWU	不可	-

27.3.14 出力管理

各タイミングユニットは 1 対の出力を制御します。出力には、3 つの動作状態があります。

- RUN：これは主動作モードであり、出力はクロスバーでプログラムされたとおり、アクティブまたは非アクティブになります。
- IDLE：この状態は、出力がソフトウェアによって無効にされているとき、またはバーストモード動作中の（そこでは、通常動作モード中、出力は一時的に無効にされます。詳細はセクション 27.3.15：を参照）、HRTIM リセット後のデフォルトの動作状態です。永続的にアクティブまたは非アクティブになります。
- FAULT：これは安全状態であり、FAULTx 入力時の停止リクエストの場合に、この状態になります。永続的にアクティブ、非アクティブ、またはハイインピーダンスになります。

表 232 に示すように、出力ステータスは HRTIM_OENR レジスタの TxyOEN ビットおよび HRTIM_ODSR レジスタの TxyODS ビットによって示されます。

表 232. 出力状態のプログラミング（x = A～F、y = 1 または 2）

TxyOEN（制御／ステータス）（ソフトウェアでセットされ、ハードウェアでクリアされる）	TxyODS（ステータス）	出力動作状態
1	x	RUN
0	0	IDLE
0	1	FAULT

TxyOEN ビットは制御ビットとステータス･ビットの両方として機能します。出力を RUN モードにするにはソフトウェアでセットする必要があります。出力が IDLE または FAULT モードになったときにハードウェアによってクリアされます。TxyOEN ビットがクリアされたとき、TxyODS ビットは出力が IDLE 状態なのか、または FAULT 状態なのかを示します。HRTIM_ODISR レジスタの 3 番目のビットによって、出力をソフトウェアで無効にできます。

図 234. 出力管理の概要

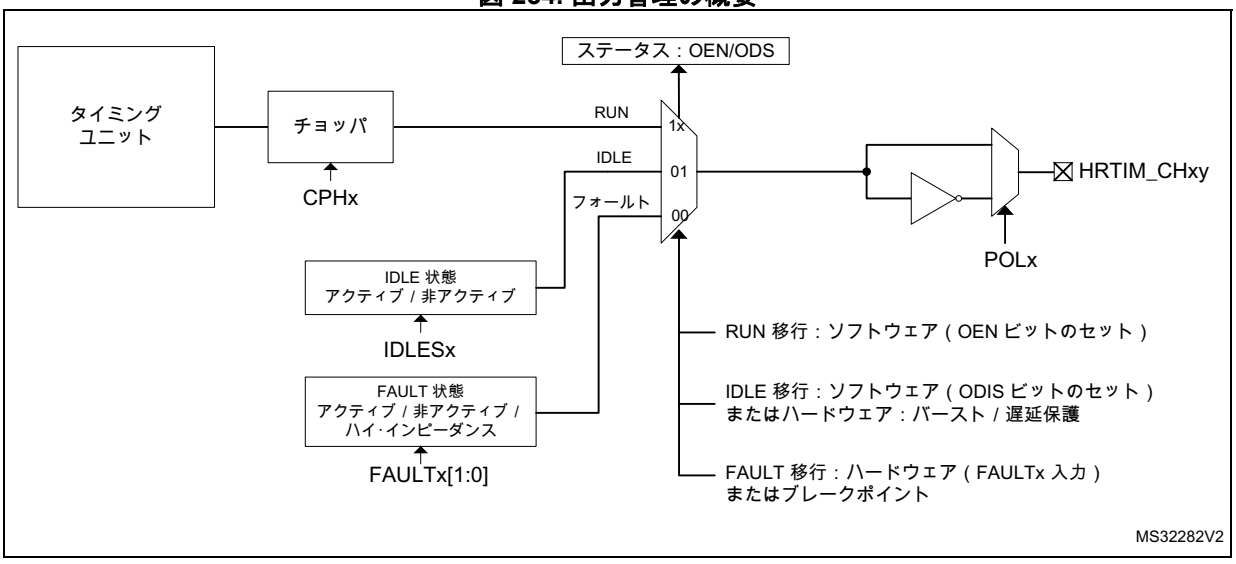
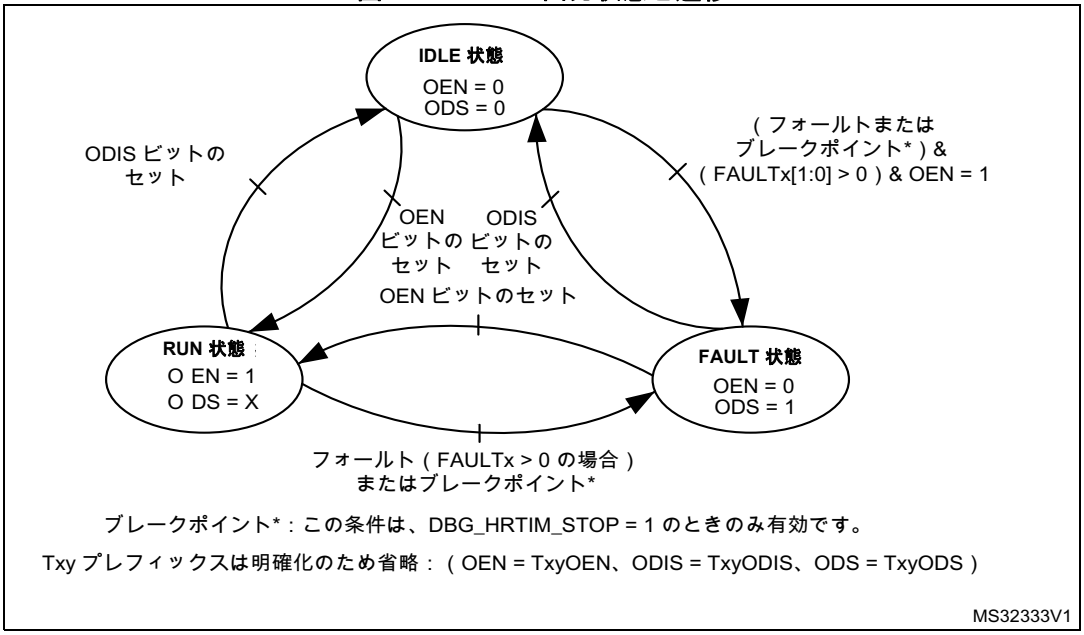


図 235 は、3 つの状態に対するビット値と、遷移がどのようにトリガされるかを示します。セクション 27.3.17：に挙げたように、フォールトは外部または内部フォールトソースによってトリガされま

すが、バーストモードまたは遅延保護がアクティブになっているとき、アイドル状態に移行することができます。

図 235. HRTIM 出力状態と遷移



FAULT および IDLE レベルはアクティブまたは非アクティブとして定義されます。アクティブ（または非アクティブ）は、電源スイッチを閉（非アクティブ状態の場合は開）に導くタイマ出力のレベルを指します。

IDLE 状態は最も高い優先順位です。FAULT 状態が有効でも、ODIS ビットのセットにトリガされて FAULT → IDLE の遷移は可能です。

FAULT 状態は RUN 状態より優先順位が高く、TxyOEN ビットがフォールトイベントと同時にセットされた場合、FAULT 状態に移行します。IDLE → FAULT の遷移は、図 235 のように、次の条件で起きます。すなわち、フォールト保護が有効化され（FAULTx[1:0] ビット = 01、10、11）、かつ、フォールトがアクティブで（または DBG_HRTIM_STOP = 1 の場合、ブレークポイント中に）OEN ビットがセットされる必要があります。

出力の極性は HRTIM_OUTxR の POLx ビットを使用してプログラムされます。POLx = 0 のとき極性は正（出力はアクティブハイ）で、負の極性（POLx = 1）の場合は、アクティブローとなります。実用的には、極性は駆動する電源スイッチ（PMOS 対 NMOS）に応じて、またはゲートドライバの極性に応じて定義されます。

FAULT 状態の出力レベルは、以下のように各出力に対して、HRTIM_OUTxR の FAULTx[1:0] ビットを使用して設定します。

- 00：出力はフォールト状態には移行せず、RUN または IDLE 状態に留まります。
- 01：FAULT 状態のとき、出力はアクティブレベルになります。
- 10：FAULT 状態のとき、出力は非アクティブレベルになります。
- 11：FAULT 状態のとき、出力はトライステートになります。たとえば、安全状態はプルアップ抵抗またはプルダウン抵抗によって外部的に強制する必要があります。

注： **FAULTx[1:0] ビットは、出力が FAULT 状態である限り、変更してはなりません。**

IDLE 状態の出力レベルは、HRTIM_OUTxR の IDLESx ビットを使用して以下のように設定します。

- 0：IDLE 状態のとき、出力は非アクティブレベルになります。
- 1：IDLE 状態のとき、出力はアクティブレベルになります。

TxyOEN ビットがセットされて RUN 状態に移行すると、出力は直ちにクロスバー出力に接続されます。タイマクロックが停止すると、レベルは非アクティブになるか（HRTIM リセット後）、または RUN レベルに対応します（タイマが停止して出力が無効になるとき）。

HRTIM 初期化中に、HRTIM_SETx1R および HRTIM_RSTx1R レジスタでのソフトウェアによる出力セット・リセットを使用して、出力レベルを RUN モードに移行する前に事前設定できます。

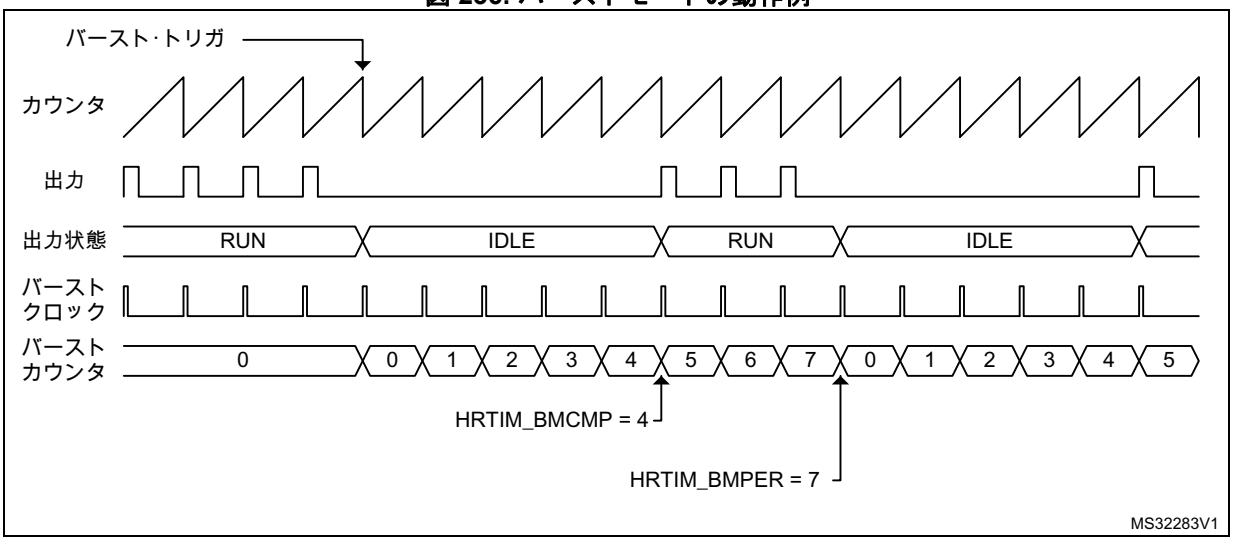
27.3.15 バーストモードコントローラ

バーストモードコントローラによって、出力をハードウェアで交互に IDLE 状態と RUN 状態にすることができます。こうすることによって、いくつかのスイッチング周期をプログラム可能な周期性とデューティサイクルでスキップすることができます。

バーストモード動作は、軽負荷で動作するときの電力コンバータで一般に使用されています。これによって、出力の遷移回数とそれに伴うスイッチング損失を低減することによって、コンバータの効率を大幅に向上できます。

バーストモードで動作するとき、カウントの数周期に等しいアイドル期間に続いて 1 つまたは数パルスが出力されます。一般的にアイドル期間では、図 236 の例に示すように、出力パルスは生成されません。

図 236. バーストモードの動作例



バーストモードコントローラは以下のものから構成されます。

- HRTIM の内部または外部（典型として PWM 周期の終端）のさまざまなソースによってクロック供給されるカウンタ。
- アイドル周期の数を定義する比較レジスタ：HRTIM_BMCMP。
- バーストの繰り返しレート（アイドル期間と実行期間の合計に相当）を定義する周期レジスタ：HRTIM_BMPER。

バーストモードコントローラは 10 個の PWM 出力を制御することができます。表 233 に示すように、バーストモード動作中の各出力の状態は、HRTIM_OUTxR レジスタの IDLESx および IDLEMx ビットを使用してプログラムされます。

表 233. バーストモードのためのタイマ出力のプログラミング

IDLEMx	IDLESx	バーストモード中の出力の状態
0	X	作用なし：出力はバーストモード動作の影響を受けません。
1	0	バーストモード中、出力は非アクティブ
1	1	バーストモード中、出力はアクティブ

注： **IDLEMx ビットは、バーストモードがアクティブである間は、変更してはなりません。**

バーストモードコントローラは出力ステージにのみ作用します。アイドル期間中も、以下の多くのイベントが発生します。

- 出力のセット／リセット割込みまたは DMA リクエスト
- Tx2 出力信号に基づく外部イベントのフィルタリング
- 出力セット／リセットにトリガされたキャプチャイベント

バーストモード中には、TxBM ビットがセットされていても、hrtim_out_sync[2:1] 出力には開始イベントもリセットイベントも生成されません。

動作モード

所定のタイミングユニットにバーストモードを使用する前にカウンタを有効に（TxCEN ビットをセット）する必要があります。バーストモードは HRTIM_BMCR レジスタの BME ビットで有効にします。

HRTIM_BMCR レジスタの BMOM ビットを使用して、連続またはシングルショットモードで動作させることができます。連続モードは、BMOM =1 のときに有効になります。バースト動作は、HRTIM_BMCR の BMSTAT ビットをリセットして動作を終了させるまで維持されます。

シングルショットモード（BMOM = 0）では、バーストモードのトリガ後、アイドルシーケンスが 1 回実行され、その後すぐに、通常のタイマ動作が再開されます。

アイドル期間および実行期間の持続時間はバーストモードカウンタと 2 つのレジスタで定義されます。HRTIM_BMCMPR レジスタは、選択されたタイマがアイドル状態にある間（アイドル期間）のカウント数を定義します。HRTIM_BMPER は、全体のバーストモードの周期（アイドル期間と実行期間の合計）を定義します。最初のバーストモードのトリガが発生すると、アイドル期間の長さは HRTIM_BMCMPR+1 となり、全体のバースト周期は HRTIM_BMPER+1 となります。

注： **バーストモードの周期は、DTRx[8:0] および DTFx[8:0] ビットフィールドで定義されたデッドタイム期間以下であってはなりません。**

バーストモード動作中、タイミングユニットおよびマスタタイマのカウンタは停止させてリセットすることができます。このために、HRTIM_BMCR には、次の 6 つの制御ビットがあります。MTBM（マスタ）および、タイマ A～E 用の TABM～TEBM です。

MTBM または TxBM ビットがリセットされているとき、カウンタクロックは維持されます。これによって、たとえば、多相システムにおいて他のタイマとの位相関係を保つことができます。

MTBM または TxBM ビットがセットされているとき、対応するカウンタは、アイドル期間中に停止され、リセット状態に維持されます。これによって、アイドルから出た時に、タイマをフル周期でリスタートさせることができます。SYNCSRC[1:0] = 00 または 10（マスタ開始時またはタイマ A 開始時に同期出力）である場合、バーストモードが終了した時、HRTIM_SCOUT 出力にパルスが送信されます。

注： **TxBM ビットは、バランスアイドルモードがアクティブなときは（DLYPRT[1:0] = 0x11）、セットしてはいけません。**

バーストモードクロック

バーストモードコントローラのカウンタは、HRTIM_BMCR レジスタの BMCLK[3:0] ビットで選択される下記のいくつかのソースをクロックとして動作できます。

- BMCLK[3:0] = 0000～0101 : マスタタイマおよび TIMA～E リセット／ロールオーバーイベント。これによって、バーストモードのアイドルおよび実行周期をタイミングユニットのカウント周期にそろえることができます（フリーランニングモードおよびカウンタリセットモードの両方で）。
- BMCLK[3:0] = 0110～1001 : 表 211 のように、クロックは、汎用タイマに接続された hrtim_bm_ck[4:1] 入力によって供給されます。この場合、バーストモードのアイドルおよび実行周期は、必ずしもタイミングユニットのカウント周期にそろうとは限りません（たとえば、出力上のパルスは中断されて、修正されたデューティサイクルの波形になることもあります）。
- BMCLK[3:0] = 1010 : HRTIM_BMCR レジスタの BMPRSC[3:0] ビットで定義された分周比によってプリスケールされた f_{HRTIM} クロック。この場合、バーストモードのアイドルおよび実行周期は、必ずしもタイミングユニットのカウント周期にそろうとは限りません（たとえば、出力上のパルスは中断されて、修正されたデューティサイクルの波形になることもあります）。

TIMx OC 出力のパルス幅は、HRTIM バーストモードコントローラが検出できるように、少なくとも f_{HRTIM} の N クロックサイクルの長さでなければなりません。

バーストモードトリガ

バースト動作をトリガするために、以下の 32 個のソースが利用でき、HRTIM_BMTRGR レジスタを使用して選択します。

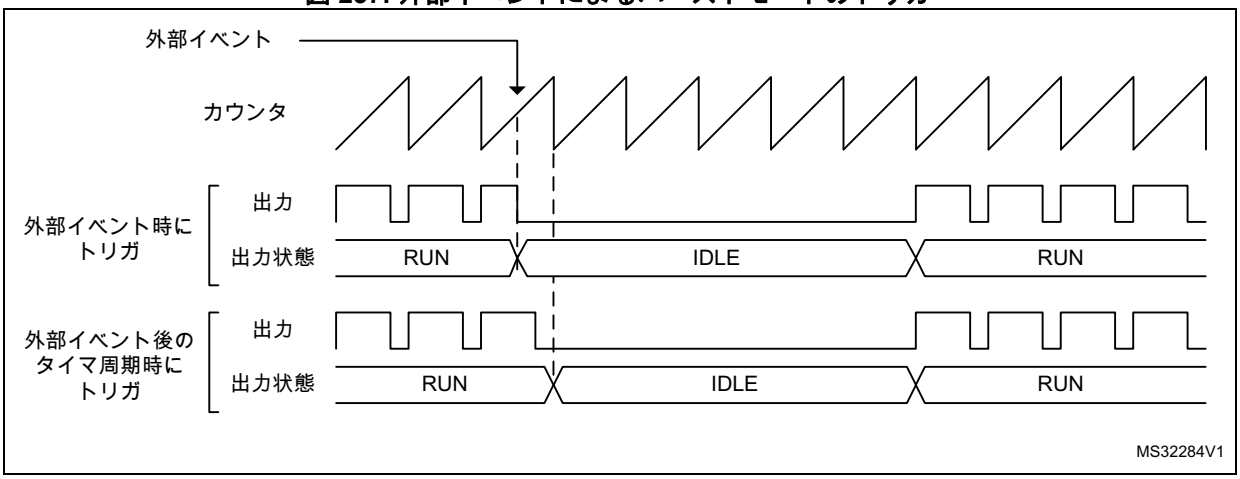
- ソフトウェアトリガ（ソフトウェアによってセットされ、ハードウェアによってリセットされます）
- 6 つのマスタタイマイベント：繰り返し、リセット／ロールオーバー、比較 1～4
- タイマ A～F からの 5 x 4 イベント：繰り返し、リセット／ロールオーバー、比較 1 および 2
- 外部イベント 7（TIMA イベントフィルタリングを含む）および 8（TIMD イベントフィルタリングを含む）
- 外部イベント 7（TIMA イベントフィルタリングを含む）後のタイマ A 周期
- 外部イベント 8（TIMD イベントフィルタリングを含む）後のタイマ D 周期
- hrtim_bm_trg 入力（汎用タイマ TRGO 出力に接続されている）上のオンチップイベント（詳細は、表 208 を参照 ）

これらのソースは組み合わせて複数同時トリガとすることができます。

バーストモードは再トリガ可能ではありません。連続モードでは、バーストモードが終了するまで新しいトリガは無視されますが、シングルショットモードでは、実行期間（HRTIM_BMPER+1 サイクル）を含む現在のバーストが完了するまで、トリガが無視されます。これは、ソフトウェアトリガについてもあてはまります（ソフトウェアビットは、破棄された場合、ハードウェアイベントによってリセットされます）。

図 237 に、バーストモードがどのように外部イベントに反応して開始されるか（直ちにか、あるいはイベント後のタイマ周期時か）を示します。

図 237. 外部イベントによるバーストモードのトリガ



TAEEV7 と TDEEV8 の組み合わせトリガ（外部イベント後のタイマ周期時にトリガ）に対しては、バーストモードのプログラミングおよび進行中のバースト動作にかかわらず、以下のように、外部イベントの検出は常にアクティブです。

- 外部イベントとタイマ周期イベントの間で、バーストモードが有効（BME=1）、またはトリガが有効（BMTRG レジスタの TAEEV7 または TDEEV8 ビットがセット）になると、バーストはトリガされます。
- シングルショットバーストモードは、バーストが終わる前に外部イベントが発生しても（対応する周期がバーストの後に起きる限り）、再トリガされます。

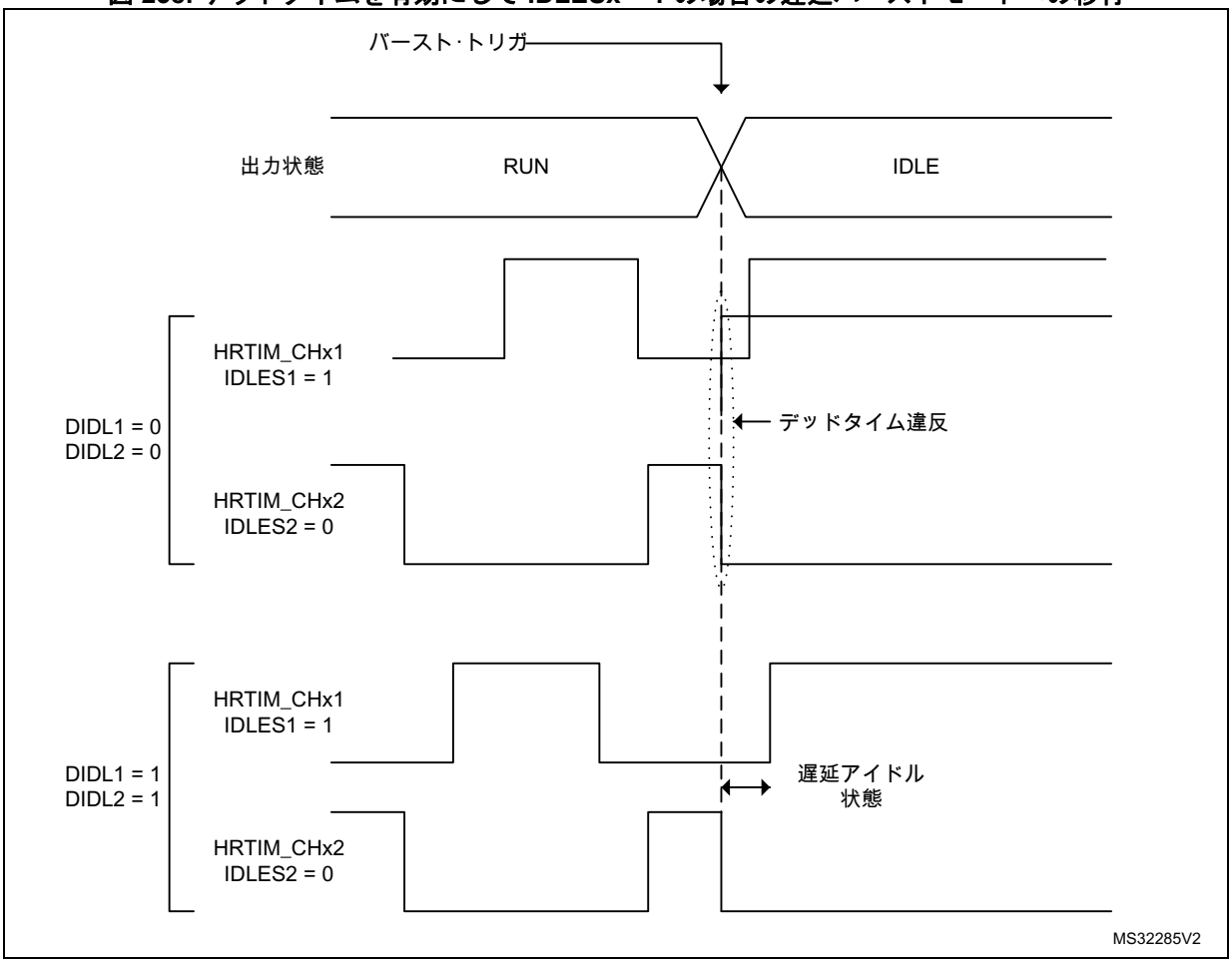
注： **TAEEV7 および TDEEV8 トリガは周期イベントの後にのみ有効です。周期イベントの前にカウンタがリセットされると、保留中の hrtim_eev7 または hrtim_eev8 イベントは破棄されます。**

バーストモードの遅延移行

デフォルトにより、バーストモードのトリガ直後、出力は（IDLES1 および IDLES2 の設定により）アイドルレベルになります。

バーストモードへの移行を遅らせて、出力がそのアイドル状態になる前にプログラム可能な期間の間、出力を非アクティブ状態に強制することも可能です。これは、図 238 に示すように、2 つの相補出力を駆動していて、そのうちの 1 つがアクティブアイドル状態であるとき、デッドタイム違反を避けるのに有用です。これはハーフブリッジでのシュートスルー電流のリスクを防ぎますが、バーストモードへの移行の応答が遅延します。

図 238. デッドタイムを有効にして IDLESx = 1 の場合の遅延バーストモードへの移行



遅延バースト移行モードは HRTIM_OUTxR レジスタの DIDLx ビット（出力あたり 1 つのイネーブルビット）によって有効になります。出力がアイドル状態になる前にデッドタイムの挿入が強制されます。TIMx 出力ごとに、それぞれ独自のデッドタイム値を次のように設定します。

- DIDL1 = 1 のとき、出力 1 で DTRx[8:0]
- DIDL2 = 1 のとき、出力 2 で DTFx[8:0]

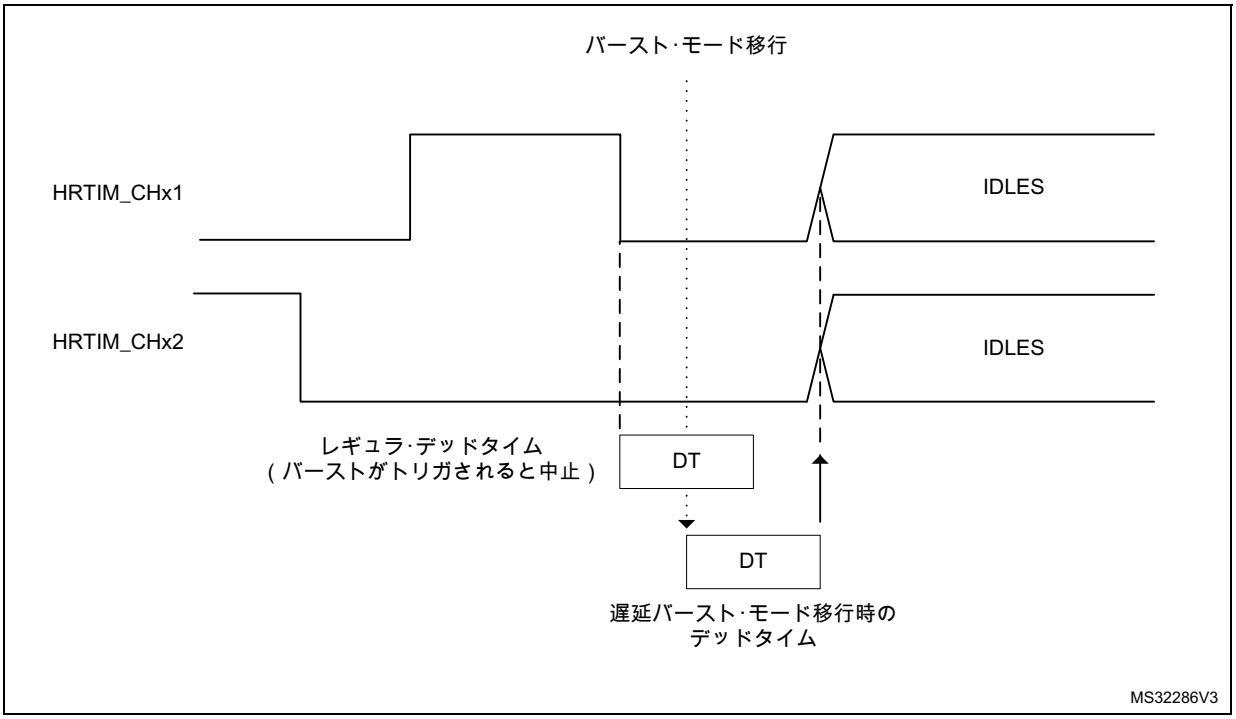
バーストモード中に出力の 1 つがアクティブアイドルレベルとなっている場合（IDLES = 1）のみ、および、正のデッドタイムが使用される（SDTR/SDTF が 0 にセットされている）ときのみ、DIDLx ビットをセットすることができます。

注： **遅延バースト移行モードはデッドタイムジェネレータリソースを使用します。したがって、2 つの DIDLx ビットのどちらかがセットされ、対応するタイミングユニットがデッドタイム挿入を使用する（HRTIM_OUTxR の DTEN ビットのセット）ときは、タイマ x 出力 2 を外部イベント用のフィルタとして使用することはできません（Tx2 フィルタ信号は利用できません）。**

DTRx[8:0] および DTFx[8:0] によって定義された期間が f_{HRTIM} クロックの 3 サイクル周期より小さいとき、セクション 27.3.7: に記載されている狭パルス管理に関する制限を適用する必要があります。

バーストモード移行が通常のデッドタイムの間に起きると、デッドタイムは中止され、図 239 のように、非アクティブ期間に対応して新しいデッドタイムが再び開始されます。

図 239. デッドタイム中の遅延バーストモードへの移行



バーストモードの終了

バーストモードの終了は、ソフトウェアによって強制されるか（連続モード時）、またはアイドル期間が経過したときに起きます（シングルショットモード時）。どちらの場合も、カウンタは直ちに再開されますが（MTBM または TxBM ビット = 1 によってリセット状態に保持されていた場合）、出力状態のアイドルからアクティブモードへの実効遷移はプログラムされたセット／リセットイベント後にのみ起きます。

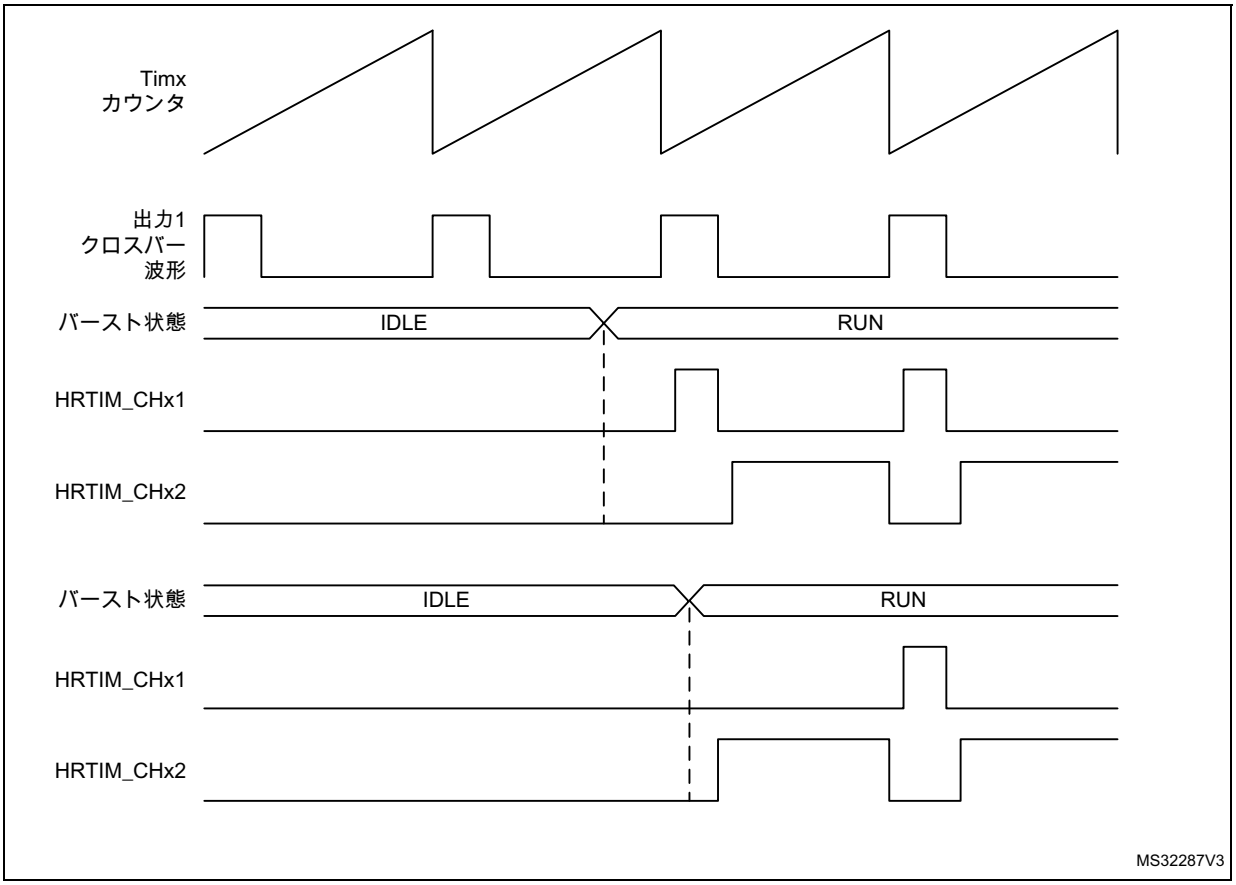
HRTIM_IER レジスタにおいて BMPERIE イネーブルビットがセットされていると、シングルショットおよび連続モードにおいてバースト周期の割込みが生成されます。この割込みはバーストモードの終了を連続バーストモードでのバースト周期に同期させるために使用できます。

図 240 に、デッドタイムが有効になっているとき、通常の動作がどのように再開されるかを示します。バーストモードの終了が即時であっても、これは、相補出力のどちらかの最初のセットイベントでのみ有効になります。

次の 2 つの異なるケースを示します。

1. 信号がクロスバー出力波形上で非アクティブの間にバーストモードが終了します。Tx1 出力のセットイベント時に、Tx1 および Tx2 上でアクティブ状態が再開され、Tx2 出力はバースト終了時には相補レベルになりません。
2. クロスバー出力波形がアクティブの間にバーストモードが終了します。動作は Tx2 出力のセットイベント時に再開され、Tx1 はバースト終了時にすぐにはアクティブレベルになりません。

図 240. デッドタイムジェネレータが有効なときのバーストモードの終了



プッシュプルモードが有効なときは、上述の動作は若干異なります。出力が非アクティブの場合、プッシュプルモードは周期の始めに出力を強制的にリセットするか、あるいは、出力が前の周期の間にハイであった場合、アクティブレベルに強制します。

したがって、アクティブアイドル状態の出力は、遷移が明示的にプログラムされていなくても、バーストモードが終了する時点でリセットできます。反対に、出力がアイドル状態に入った時にアクティブであった場合、遷移が明示的にプログラムされていなくても、出力はバーストモードが終了する時点でセットできます。

バーストモードレジスタのプリロードおよび更新

BMPREN ビット（バーストモードプリロードイネーブル）によって、バーストモードの比較および周期レジスタ（HRTIM_BMCMP および HRTIM_BMPER）をプリロードさせることができます。

BMPREN がセットされていると、以下の時、プリロードからアクティブレジスタへの転送が起きます。

- バーストモードが有効になる時（BME = 1）
- バーストモード周期の終了時

レジスタが変更されるときの 2 つのレジスタの一貫性を確保するために、HRTIM_BMPER 周期レジスタへの書込み時は、HRTIM_BMCMP 比較レジスタが書き込まれるまで、更新が一時的に無効になります。

比較レジスタのみ変更する必要がある場合は、シングル書込みをする必要があります。周期のみを変更する必要がある場合は、新しい値が考慮されるように比較も再書込みが必要です。

BMPREN ビットがリセットされているとき、BMCMPR および BMPER へ書込みアクセスすると、アクティブレジスタが直接、更新されます。この場合、以下の 2 つのケースについて、全体バースト周期中のどの時点で更新するかを考慮する必要があります。

a) 比較レジスタの更新

新しい比較値が現在のバーストモードカウンタ値より大きい場合、新しい比較は現在の周期に考慮されます。

新しい比較値が現在のバーストモードカウンタ値より小さい場合、連続モードでの新しい比較は次のバースト周期で考慮され、シングルショットモードでは無視されます（比較一致は起きず、アイドル周期の終りまでアイドル状態が続きます）。

b) 周期レジスタの更新

新しい周期値が現在のバーストモードカウンタ値より大きい場合、変更は現在の周期に考慮されます。

注： **新しい周期値が現在のバーストモードカウンタ値より小さい場合、新しい周期は考慮されず、バーストモードカウンタは（0xFFFF で）オーバーフローし、変更は次の周期で有効になります。シングルショットモードでは、カウンタは 0xFFFF でロールオーバーし、バーストモードは新しいプログラム値までの別の周期で再開します。**

合成レジスタを使用したバーストモードのエミュレーション

バーストモードコントローラは 1 つのコンバータについて 1 つまたは 1 セットのタイマのみを制御します。複数の独立したタイマにバーストモードが必要なとき、DMA と HRTIM_CMP1CxR 合成レジスタ（繰り返しレジスタと比較 1 レジスタの両方のエイリアスを持つ）を使用して単純なバーストモードコントローラをエミュレートすることができます。

これは、デューティサイクルのみ更新する必要がある単純な PWM のみを必要とするコンバータ（降圧コンバータが典型的）に適用できます。この場合、出力をリセット（そしてデューティサイクルを定義）するのに CMP1 レジスタを使用し、セットは周期イベントで行います。

この場合、デューティサイクル（CMP1 値で定義）および、このデューティサイクルが維持される周期の数（繰り返し値で定義）を定義するためには、CMP1CxR へ 32 ビット書込みアクセスを 1 回するだけで十分です。バーストモードを実装するには、次のように構成した 2 つの 32 ビットデータを DMA によって（繰り返しイベント時に）連続モードで転送することしか必要ありません。

CMPC1xR = {REP_Run; CMP1 = デューティサイクル}、{REP_Idle; CMP1 = 0}

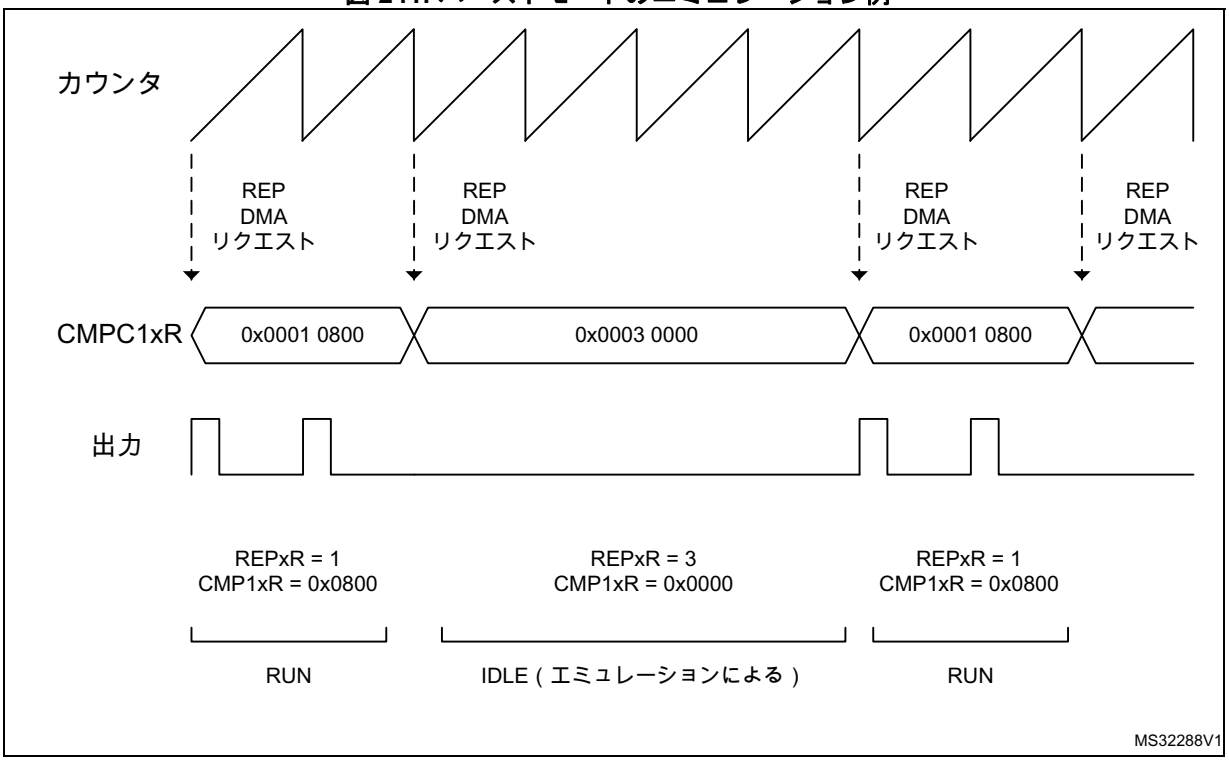
たとえば、次の値：

{0x0003 0000}：3 周期間、CMP1 = 0

{0x0001 0800}：1 周期間、CMP1 = 0x0800

は、図 241 に示すように、PWM の 6 周期ごとに 2 周期アクティブであるバーストモードを提供します。

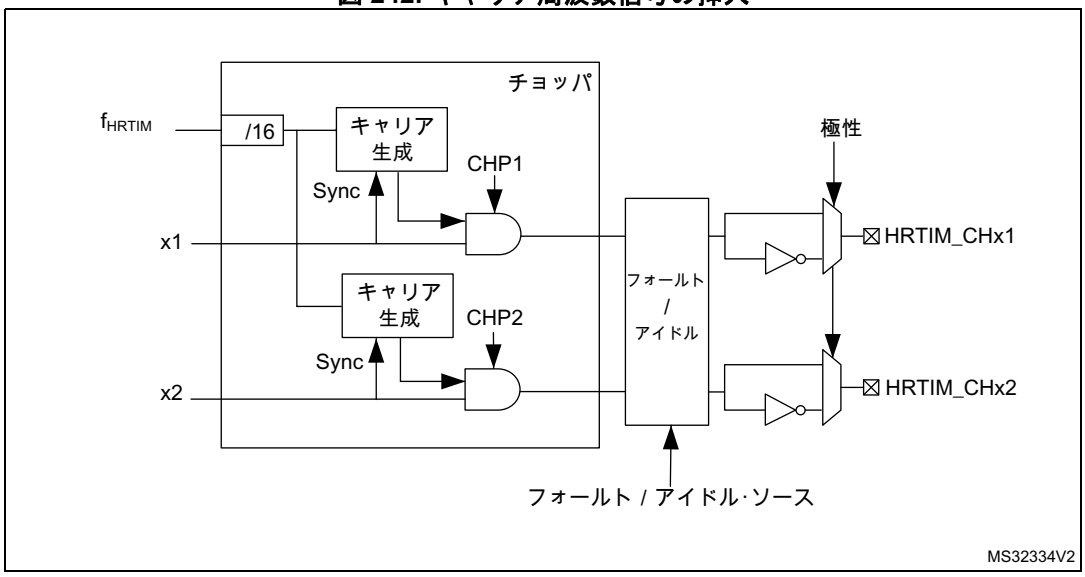
図 241. バーストモードのエミュレーション例



27.3.16 チョッパ

タイミングユニットの出力信号の先頭で高周波キャリアを付加して絶縁型変成器を駆動できます。これは、図 242 に示すように、出力 1 と 2 それぞれでチョッパを有効にするために、HRTIM_OUTxR レジスタの CHP1 および CHP2 ビットを使用して、出力ステージで極性の挿入の前に行われます。

図 242. キャリア周波数信号の挿入



チョッパのパラメータは HRIM_CHPxR レジスタを使用して調節できます。すなわち、図 243 のように、パルスの始めに特定のパルス幅を定義し、その後に、プログラム可能な周波数とデューティサイクルを持ったキャリア周波数が続くように指定できます。

CARFRQ[3:0] ビットによって、式 $F_{CHPFRQ} = f_{HRTIM} / （16 x （CARFRQ[3:0]+1））$に従って、664.06 kHz～10.625 MHz（f_{HRTIM} = 170 MHz に対して）の範囲の周波数が定義されます。

デューティサイクルは、CARDTY[2:0] によって 0/8 から 7/8 デューティサイクルまで 1/8 ステップずつ調節することができます。CARDTY[2:0] = 000（デューティサイクル = 0/8）のとき、出力波形はリファレンス波形の立ち上がりエッジの後に、搬送波のない開始パルスのみを含むものになります。

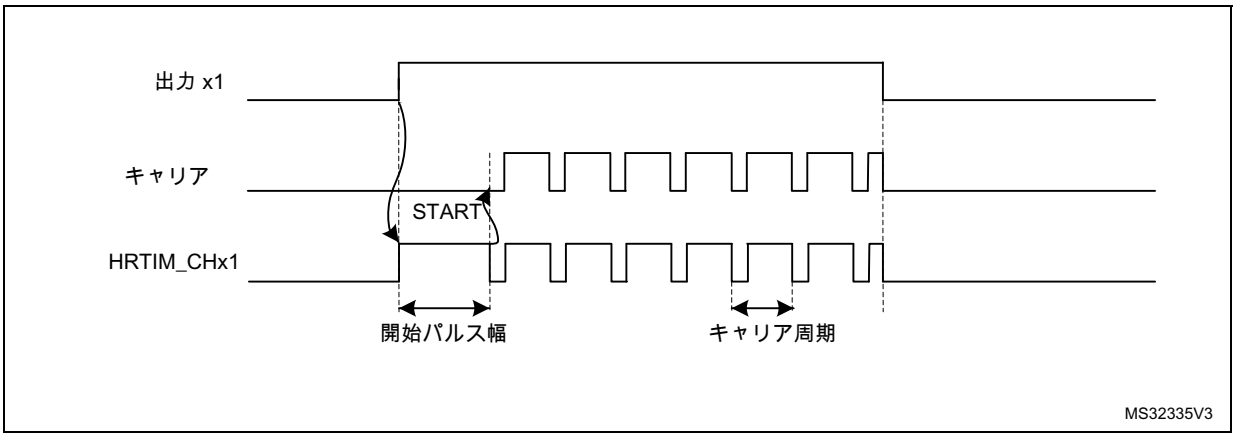
最初のパルスのパルス幅は STRPW[3:0] ビットフィールドを使用して次のように定義されます：t1STPW = （STRPW[3:0]+1）x 16 x t_{HRTIM} であり、その範囲は 94 ns～1.51 µs（f_{HRTIM}=170 MHz に対して）です。

キャリア周波数のパラメータは f_{HRTIM} 周波数に基づいて定義されており、CKPSC[2:0] の設定には依存しません。

チョッパモードでは、キャリア周波数と最初のパルス幅は、AND 機能を使用してリファレンス波形と結合されます。同期は最初のパルスの終りに行われ、繰り返し信号の形になります。

チョッピング信号は、現在のキャリア周期が完了するまで待たずに、出力波形のアクティブ状態の終りに停止します。よって、プログラムされたパルス幅より短いパルスが含まれることがあります。

図 243. チョッパモードが有効化された HRTIM 出力



注： **CHP1 および CHP2 ビットは、HRTIM_OENR レジスタの TxyOEN ビットで出力を有効にする前にセットする必要があります。**

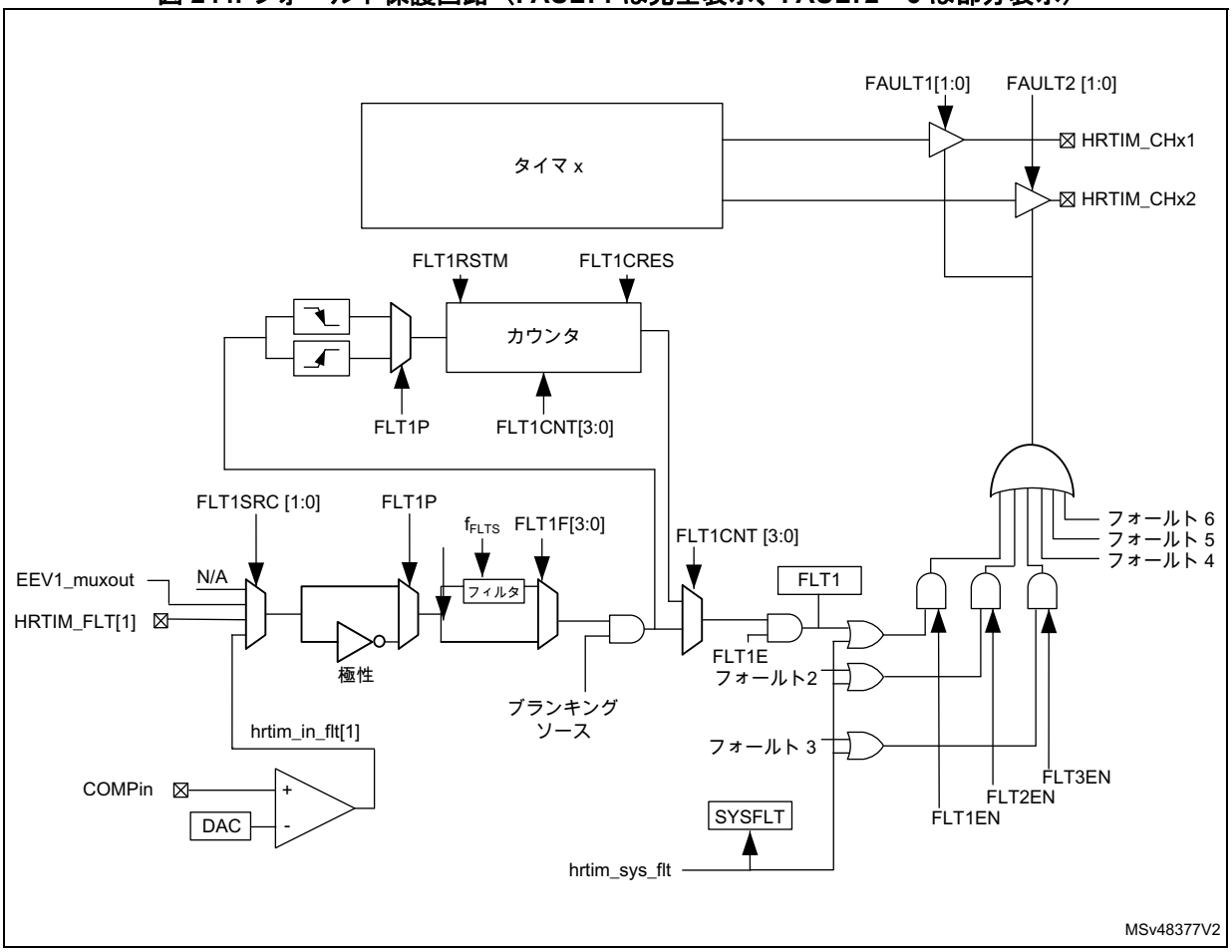
CARFRQ[2:0]、CARDTY[2:0]、および STRPW[3:0] ビットフィールドは、チョッパモードがアクティブである（2 つの CHPx ビットの少なくとも 1 つがセットされている）間は変更できません。

27.3.17 フォールト保護

HRTIM は、多目的なフォールト保護回路を備えており、異常な動作の場合に出力を無効にします。フォールトがトリガされると、出力は事前に定義された安全状態になります。この状態は出力がソフトウェアによって再度有効にされるまで維持されます。永続的フォールトのリクエストの場合は、出力は、ソフトウェアが再度有効化しようとしても、フォールトソースが消えるまで、フォールト状態に留まります。

HRTIM には 6 つの FAULT 入力チャネルがあり、図 244 に示すように、それらすべては使用可能で、6 つのタイミングユニットのそれぞれに組み合わせられます。

図 244. フォールト保護回路（FAULT1 は完全表示、FAULT2～6 は部分表示）



各フォールトチャネルは、タイミングユニットに接続される前に HRTIM_FLTINR1 および HRTIM_FLTINR2 レジスタを使用して完全に設定可能です。FLTxSRC FLTxSRC[1:0] ビットによってフォールト信号のソースを選択します。このソースは、デジタル入力または内部イベント（内蔵コンパレータの出力）のいずれかです。

表 234 と表 212 に、6 個のフォールト･チャネルの各々に対して利用可能なソースをまとめます。

表 234. フォールト入力

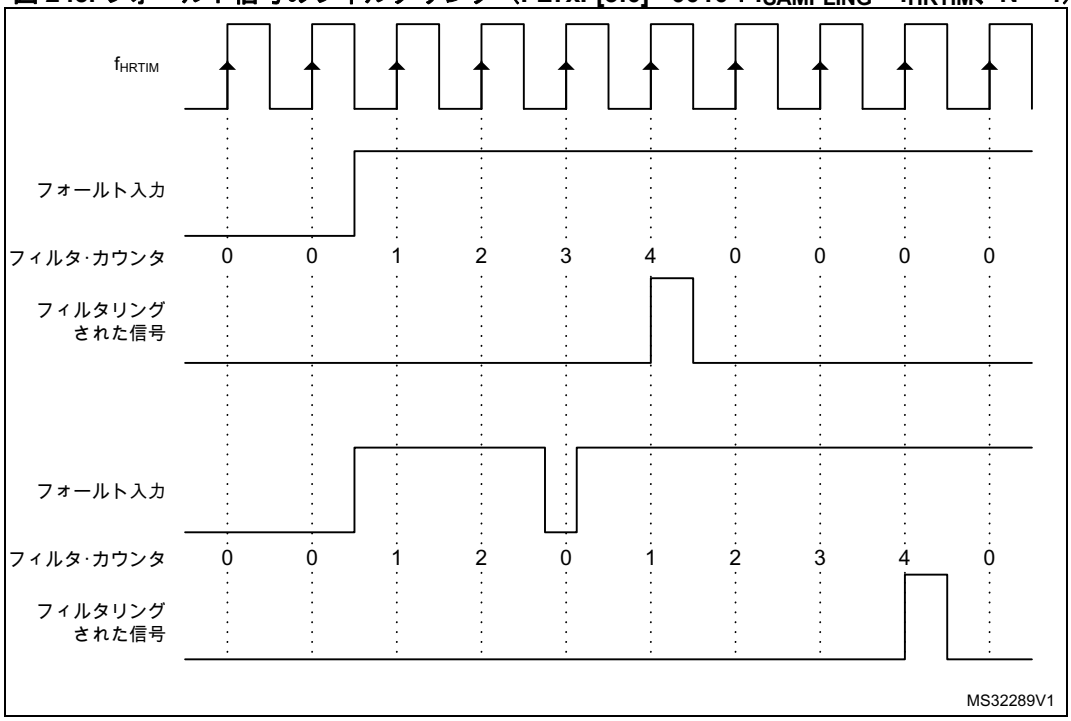
フォールト チャネル	外部入力 FLTxSRC[1:0] = 00	オンチップソース FLTxSRC[1:0] = 01	外部入力 FLTxSRC[1:0] = 10	オンチップソース FLTxSRC[1:0] = 11
hrtim_flt1[4:1]	HRTIM_FLT1	コンパレータ出力	EEV1_muxout	N/A
hrtim_flt2[4:1]	HRTIM_FLT2	コンパレータ出力	EEV2_muxout	N/A
hrtim_flt3[4:1]	HRTIM_FLT3	コンパレータ出力	EEV3_muxout	N/A
hrtim_flt4[4:1]	HRTIM_FLT4	コンパレータ出力	EEV4_muxout	N/A
hrtim_flt5[4:1]	HRTIM_FLT5	コンパレータ出力	EEV5_muxout	N/A
hrtim_flt6[4:1]	HRTIM_FLT6	コンパレータ出力	EEV6_muxout	N/A

上記の表 234 に述べた EEVx_muxout イベントは、EExSRC[1:0] ビットによって制御される hrtim_eevx[4:1] 入力マルチプレクサの後に取得されます。詳細については、図 216 を参照してください。

HRTIM_FLTINRx レジスタの FLTxP 極性ビットを使用して、アクティブレベルを定義するために信号の極性を選択できます。FLTxP = 0 の場合は、信号はローレベルでアクティブであり、FLTxP = 1 の場合は、ハイレベルでアクティブです。

フォールト情報は極性の設定の後、フィルタリングできます。FLTxF[3:0] ビットフィールドを 0000 にセットした場合は、信号はフィルタリングされず、f_{HRTIM} クロックと独立に非同期で動作します。その他の FLTxF[3:0] ビットフィールド値すべてに対しては、信号はデジタルフィルタリングされます。デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。カウンタが N の値に達する前に入力値が変化した場合、カウンタはリセットされ、変化は破棄されます（誤ったイベントとみなされます）。カウンタが N に達した場合、変化は有効とみなされ、正しい外部イベントとして転送されます。したがって、デジタルフィルタは、サンプリングクロックとフィルタ長（予想される有効なサンプルの数）に応じて、フィルタリングされる外部イベントに遅延時間を付加します。図 245 に、誤ったフォールト信号がどのようにフィルタリングされるかを示します。

図 245. フォールト信号のフィルタリング（FLTxF[3:0]= 0010：$f_{SAMPLING} = f_{HRTIM}$、N = 4）



フィルタ周期は f_{HRTIM} クロックの 2 サイクルから f_{FLTS} クロックの 8 サイクルを 32 で割った値までの範囲になります。f_{FLTS} は HRTIM_FLTINR2 レジスタの FLTSD[1:0] ビットを使用して定義されます。表 235 に、サンプリングレートとフィルタ長をまとめます。サンプリングによる不確かさを考慮し、有効なフィルタリングを行うために、1 サンプリングクロック周期のジッタをフィルタ長から差し引く必要があります。

表 235. サンプリングレートおよびフィルタ長 対 FLTFxF[3:0] およびクロック設定

-	f_{FLTS} 対 FLTSD[1:0]				f_{HRTIM}= 170 MHz に対するフィルタ長	
FLTFxF[3:0]	00	01	10	11	最小値	最大値
0001、0010、0011	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}	f_{HRTIM}、N =2 11.8 ns	f_{HRTIM}、N =8 47.1 ns
0100, 0101	f_{HRTIM} /2	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /2、N =6 70.6 ns	f_{HRTIM} /16、N =8 753 ns
0110, 0111	f_{HRTIM} /4	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /4、N =6 141 ns	f_{HRTIM} /32、N =8 1.51 µs
1000, 1001	f_{HRTIM} /8	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /8、N =6 282 ns	f_{HRTIM} /64、N =8 3.01 µs
1010、1011、1100	f_{HRTIM} /16	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /16、N =5 471 ns	f_{HRTIM} /128、N =8 6.02 µs
1101、1110、1111	f_{HRTIM} /32	f_{HRTIM} /64	f_{HRTIM} /128	f_{HRTIM} /256	f_{HRTIM} /32、N =5 941 ns	f_{HRTIM} /256、N =8 12.05 µs

フォールトのブランキングおよびイベントのカウント

フォールト入力は、一時的に無効にして誤ったフォールトイベントをブランキングすることができます。ブランキングソースを次の表 236 に示します。

表 236. フォールト入力のブランキングイベント

-	FLTxBLKS = 0、リセットにアラインされたウィンドウ		FLTxBLKS = 1、移動ウィンドウ	
フォールト入力	ブランキングウィンドウ開始	ブランキングウィンドウ終了	ブランキングウィンドウ開始	ブランキングウィンドウ終了
hrtim_flt1[4:1]	タイマ A リセット／ロールオーバー	タイマ A CMP3 イベント	タイマ A CMP4 イベント	タイマ A CMP3 イベント
hrtim_flt2[4:1]	タイマ B リセット／ロールオーバー	タイマ B CMP3 イベント	タイマ B CMP4 イベント	タイマ B CMP3 イベント
hrtim_flt3[4:1]	タイマ C リセット／ロールオーバー	タイマ C CMP3 イベント	タイマ C CMP4 イベント	タイマ C CMP3 イベント
hrtim_flt4[4:1]	タイマ D リセット／ロールオーバー	タイマ D CMP3 イベント	タイマ D CMP4 イベント	タイマ D CMP3 イベント
hrtim_flt5[4:1]	タイマ E リセット／ロールオーバー	タイマ E CMP3 イベント	タイマ E CMP4 イベント	タイマ E CMP3 イベント
hrtim_flt6[4:1]	タイマ F リセット／ロールオーバー	タイマ F CMP3 イベント	タイマ F CMP4 イベント	タイマ F CMP3 イベント

また、フォールトカウンタによって、複数の誤ったフォールトイベントを破棄し、受け入れ基準を定義することもできます。

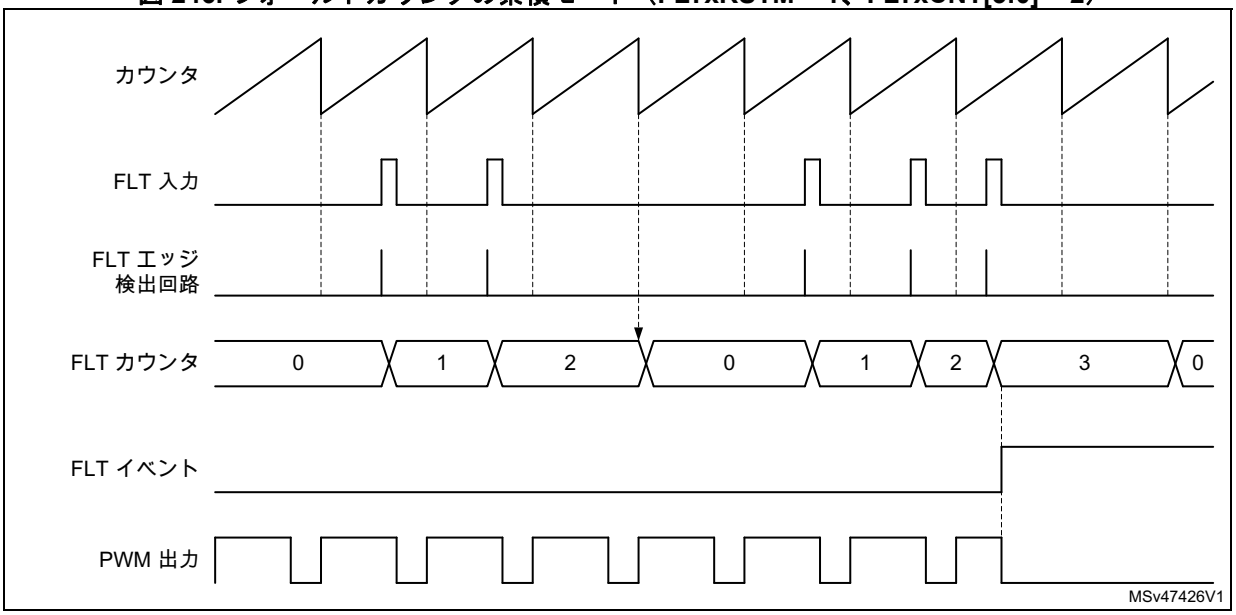
FLTxCNT[3:0] ビットフィールドによって FAULTx カウンタの閾値を選択できます。イベントの数が FLTxCNT[3:0] 値に等しいとき、フォールトは有効と見なされます。

この FLTxRSTM で FAULTx カウンタのリセットモードを選択します。

- 0 : 下記の表 237 に従って、フォールトカウンタはリセット／ロールオーバーイベントによってハードウェアリセットされます。
- 1 : 下記の図 246 に示すように、最後のカウント周期中にイベントが発生しない場合にのみ、各リセット／ロールオーバーイベントによってフォールトカウンタはリセットされます。

フォールトカウンタは、FLTxCRES ビットを使用してソフトウェアによって、いつでもリセットできます。

図 246. フォールトカウンタの累積モード（FLTxRSTM = 1、FLTxCNT[3:0] = 2）



特定の FLTx 入力カウンタは、単一のソースによってリセットすることができます。表 237 に、タイマユニットと特定のフォールトの関連付けを示します。ここでは、1 つのフォールトラインが複数のタイマで共有されてもかまいません（たとえば、タイマ A、タイマ B、およびタイマ C に同時に作用するイベントカウンタを FLT1 に有効にするなど）。

表 237. フォールト 1～6 のカウンタリセットソース

フォールト入力	フォールトカウンタのリセットソース
hrtim_flt1[4:1]	タイマ A リセット／ロールオーバー
hrtim_flt2[4:1]	タイマ B リセット／ロールオーバー
hrtim_flt3[4:1]	タイマ C リセット／ロールオーバー
hrtim_flt4[4:1]	タイマ D リセット／ロールオーバー
hrtim_flt5[4:1]	タイマ E リセット／ロールオーバー
hrtim_flt6[4:1]	タイマ F リセット／ロールオーバー

システムフォールト入力（hrtim_sys_flt）

このフォールトは MCU のクラス B 回路（詳細はシステム設定コントローラ（SYSCFG）セクションを参照）によって提供され、下記から受信するシステムフォールトに相当します。

- クロックセキュリティシステム
- SRAM パリティチェッカ
- FPU 搭載 Cortex®-M4 ロックアップ信号
- PVD 検出回路
- Flash ECC ダブルエラー検出

この入力は FAULT 入力より優先され、FAULTy[1:0] = 01、10、11 であるすべての出力を無効にします。

各 FAULT チャネルに対して、機能安全の目的で HRTIM_FLTxR レジスタの 1 回のみ書込み可能の FLTxLCK ビットによって、FLTxE、FLTxP、FLTxSRC、FLTxF[3:0] ビットをロックすることができます（これらを読出し専用にします）。有効になっていた場合は、フォールト調整の設定は、次の HRTIM リセットまたはシステム·リセットまで維持されます。

上で説明したように、フォールト信号が調整されると、それはタイミングユニット（複数）に配信されます。タイミングユニットのいずれに対しても、HRTIM_FLTxR レジスタの FLT1EN～FLT6EN ビットを使用して 6 つのフォールトチャネルが有効化され、同時に選択可能になります（システムフォールトは出力がフォールトメカニズムによって保護されている限り、自動的に有効になります）。これにより、たとえば以下のことが可能になります。

- 1 つのフォールトチャネルが同時に複数のタイミングユニットを無効にする。
- 複数のフォールトチャネルの論理和がとられて、1 つのタイミングユニットを無効にする。

機能安全の目的で、HRTIM_FLTxR レジスタの 1 回のみ書込み可能の FLTLCK ビットによって、次のリセットまで FLTxEN ビットをロックすることができます（これらを読出し専用にします）。有効になっていた場合は、タイミング·ユニットのフォールトに関連する設定は、次の HRTIM リセットまたはシステム·リセットまで維持されます。

タイマごとに、フォールト中の出力状態は HRTIM_OUTxR レジスタの FAULT1[1:0] および FAULT2[1:0] ビットによって定義されます（セクション 27.3.14：を参照）。

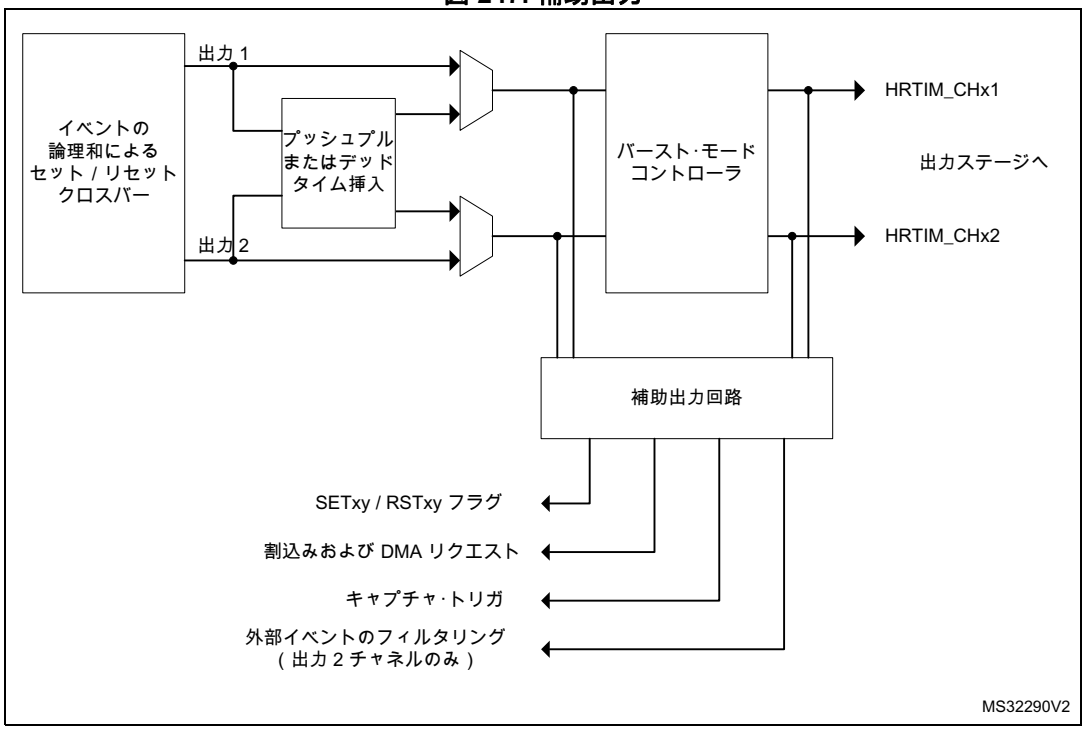
27.3.18 補助出力

タイマ A～E には、出力ステージへ行く通常の出力と並行して補助出力が備わっています。それらは以下の内部ステータス、イベント、および信号を提供します。

- SETxy および RSTxy ステータスフラグ、および対応する割込みおよび DMA リクエスト
- 出力セット／リセット時のキャプチャトリガ
- Tx2 出力コピー後の外部イベントフィルタ（セクション 27.3.9：の詳細を参照）。

補助出力は、HRTIM 動作モードに応じてバーストモードコントローラの前または後で取られます。概要を 図 247 に示します。

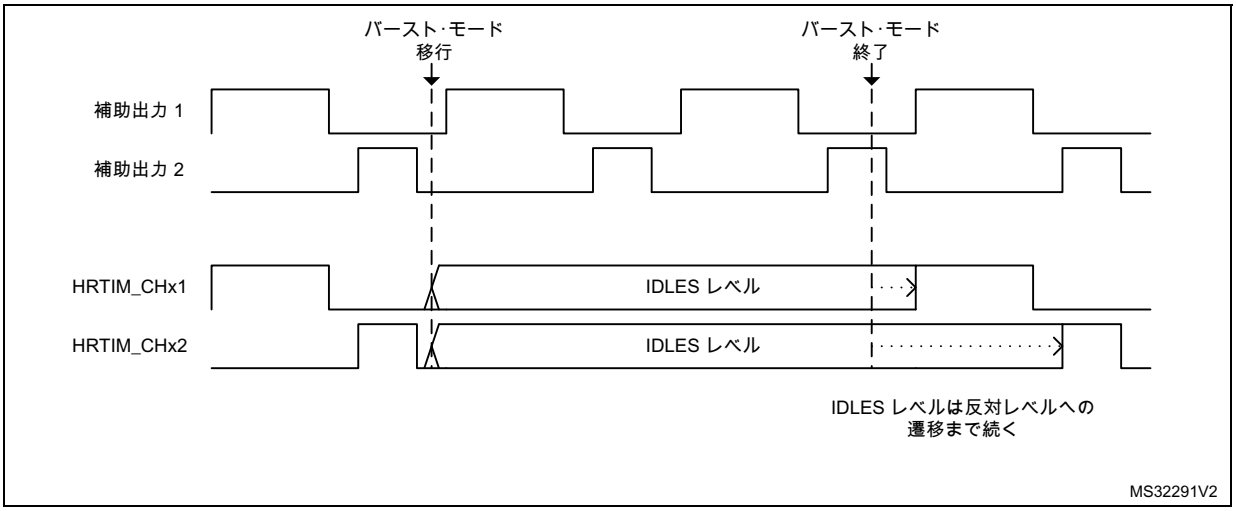
図 247. 補助出力



デフォルトでは、補助出力は出力 Tx1 と Tx2 のコピーです。ただし、例外は以下のとおりです。

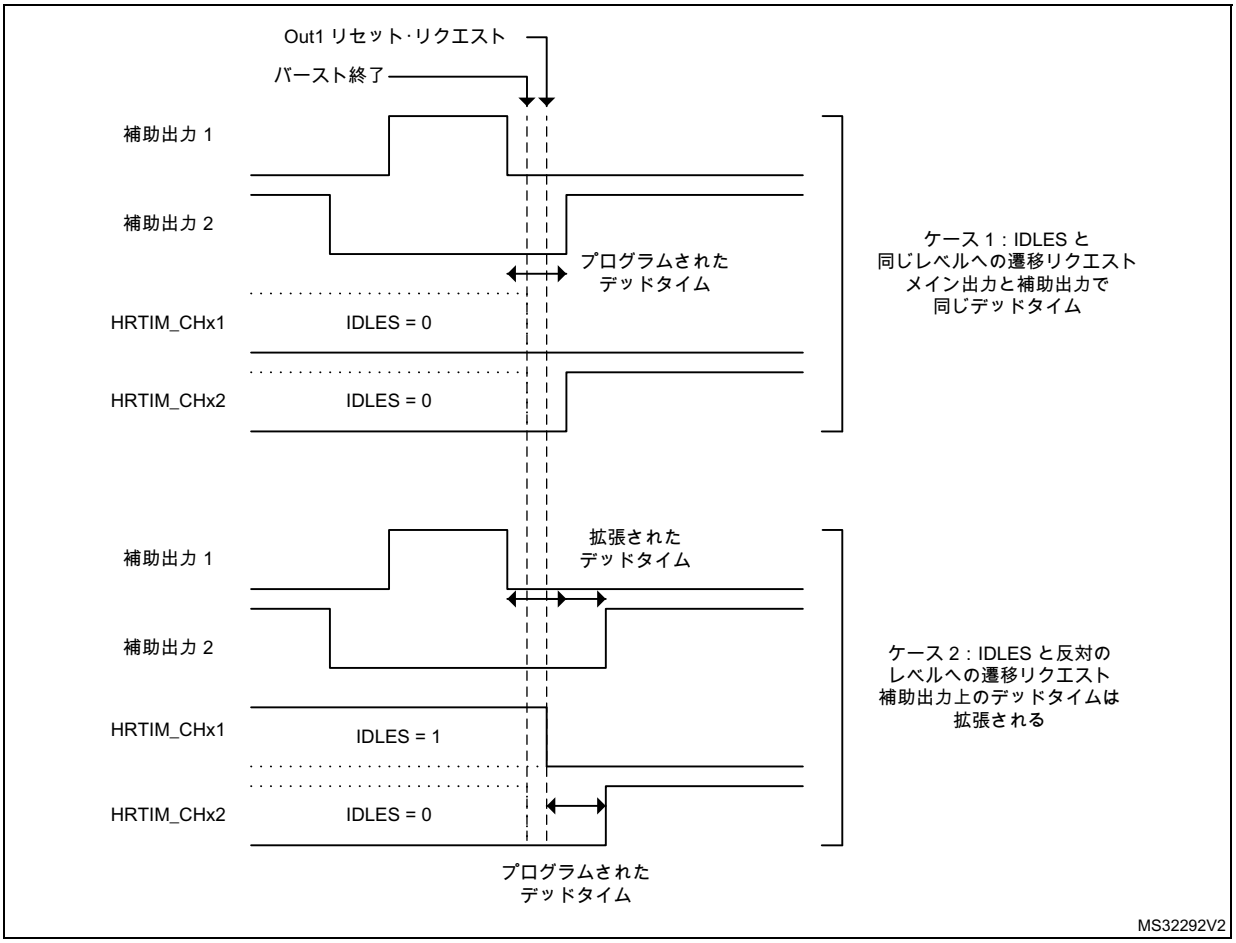
- デッドタイムが無効（DTEN = 0）になっているときの、遅延アイドルおよびバランスアイドル保護。保護がトリガされると、補助出力は維持され、クロスバーから来る信号に従います。反対に、デッドタイムが有効（DTEN = 1）になっている場合は、メイン出力と補助出力の両方とも非アクティブレベルに強制されます。
- バーストモード（TCEN=1、IDLEMx=1）。次の 2 つの場合があります。
  a) DTEN=0 または DIDLx=0 の場合、補助出力はバーストモードへの移行に影響されず、クロスバーから来るリファレンス信号に従い続けます（図 248 参照）。
  b) 遅延バーストモードへの移行（DIDLx=1）と共にデッドタイムが有効（DTEN = 1）になっている場合は、補助出力はメイン出力と同じ動作になります。デッドタイム期間の後、それらは IDLES レベルに強制され、すべてのバースト周期の間、このレベルを保持します。バーストモードが終了すると、メイン出力と同様に、IDLES レベルは反対レベルへの遷移が起きるまで維持されます。

図 248. バーストモード中の補助およびメイン出力（DIDLx = 0）



バーストモードを終了する時、または遅延保護の後に出力を再有効化する時、これがデッドタイム中に起きた場合には、補助出力上の信号はわずかに歪むことがあります。この場合、メイン出力上のデッドタイムが尊重されるように、補助出力に適用されるデッドタイムは拡張されます。図 249 に、いくつかの例を示します。

図 249. バーストモードを終了する時の補助出力上のデッドタイムの歪み



27.3.19 他のタイマまたは HRTIM インスタンスとの HRTIM の同期

HRTIM は、複数の HRTIM インスタンスをマスタユニット（同期信号を生成）として、またはスレーブ（同期されるトリガを待機）として、同期させるオプションを提供しています。この機能は、HRTIM を外部またはオンチップの他のタイマと同期させるためにも使用することができます。同期回路はマスタタイマの内部で制御されます。

同期出力

このセクションでは、外部リソースを同期させてマスタユニットとして動作するためには、HRTIM をどのように設定する必要があるかを説明します。

同期出力に転送されるソースとして 4 つのイベントを選択できます。これは、HRTIM_MCR レジスタの SYNCSRC[1:0] ビットを使用して、以下のように行います。

- 00：マスタタイマの開始
  このイベントが生成されるのは、MCEN ビットがセットされる時か、またはタイマがシングルショットモードで周期値に達した後に再開される時です。カウント中に起きるリセット時（CONT または RETRIG ビットがセットされているとき）にも生成されます。
- 01：マスタタイマの比較 1 イベント

- 10：タイマ A の開始
  このイベントが生成されるのは、TACEN ビットがセットされる時か、またはカウンタがリセットされて、このリセットに応答してカウントを再開する時です。連続モードでのカウンタロールオーバー、およびシングルショットの再トリガ不可能モードで破棄されたリセットリクエストといった、カウンタリセットイベントは同期出力には伝播されません。リセットは、カウント中に起きたとき（CONT または RETRIG ビットがセットされているとき）のみ考慮されます。
- 11：タイマ A の比較 1 イベント

HRTIM_MCR レジスタの SYNCOUT[1:0] ビットは、同期イベントがどのように生成されるかを指定します。

SYNCOUT[1:0] = 1x によって、同期パルスが HRTIM_SCOUT 出力ピンに生成されます。SYNCOUT[0] ビットは、同期信号の極性を指定します。SYNCOUT[0] = 0 の場合、HRTIM_SCOUT ピンはローアイドルレベルとなり、同期のために f_{HRTIM} クロックの 16 サイクル長の正のパルスを発行します。SYNCOUT[0] = 1 の場合、アイドルレベルはハイとなり、負のパルスが生成されます。

注： **同期パルスの後に、f_{HRTIM} クロックの 16 サイクル長のアイドルレベルが続き、その間は、新しい同期リクエストは破棄されます。したがって、最大の同期周波数は、f_{HRTIM}/32 となります。**

HRTIM_SCOUT ピンのアイドルレベルは、SYNCOUT[1:0] ビットが有効になる（つまり、ビットフィールドの値が 00 と異なる）とすぐに適用されます。

同期出力の初期化手順は、MCU 出力の設定およびカウンタ有効化の前に、以下の順に行う必要があります。

1. HRTIM_MCR の SYNCOUT[1:0] および SYNCSRC[1:0] ビットフィールドの設定
2. HRTIM_SCOUT ピンの設定（汎用 I/O のセクションを参照）
3. マスタまたはタイマ A のカウンタの有効化（MCEN または TACEN ビットのセット）

同期入力モードが有効になっていて、同期出力モード（SYNCSRC[1:0] = 00 または 10）と同時に（SYNCSTRTM/SYNCSTRTx ビットを使用して）カウンタを開始すると、カウンタが開始している時または実行中にリセットされた時にのみ出力パルスが生成されます。カウンタの開始を引き起こさずにそれをクリアするリセットリクエストは同期出力には影響しません。

同期入力

HRTIM_MCR レジスタの下記の SYNCIN[1:0] ビットにより、HRTIM は外部ソースによって同期させることができます。

- 00：同期入力は無効
- 01：予約済みの設定
- 10：hrtim_in_sync[2] 入力に接続されたオンチップタイマの TRGO 出力（詳細は表 208を参照）
- 11：HRTIM_SCIN 入力ピン上の正のパルス（hrtim_in_sync[3]）

このビットフィールドは、転送先のタイマ（マスタタイマまたはタイミングユニット）が有効化（MCEN および／または TxCEN ビットのセット）された後は、変更できません。

HRTIM_SCIN 入力は立ち上がりエッジ対応です。タイマの動作は、HRTIM_MCR および HRTIM_TIMxCR レジスタにある以下のビットによって定義されます（詳細は表 238 を参照）。

- 同期開始：到着信号がタイマのカウンタを開始させます（SYNCSTRTM および／または SYNCSTRTx ビットのセット）。タイマを有効にしてカウンタを開始準備させるためには、TxCEN（MCEN）ビットをセットする必要があります。連続モードでは、同期信号を受け取るまではカウンタは開始しません。
- 同期リセット：到着信号がカウンタをリセットします（SYNCRSTM および／または SYNCRSTx ビットのセット）。このイベントは他のリセットイベントと同様に繰り返しカウンタをデクリメントします。

同期イベントは、関連カウンタが有効化（MCEN または TxCEN ビットをセット）されてからのみ、考慮されます。同期リクエストは SYNC 割込みをトリガします。

注： **現在のカウンタ値がアクティブな周期値を超えている場合は、同期開始イベントはカウンタをリセットします。**

同期イベントの効果は、表 238 にまとめているように、タイマの動作モードに依存します。

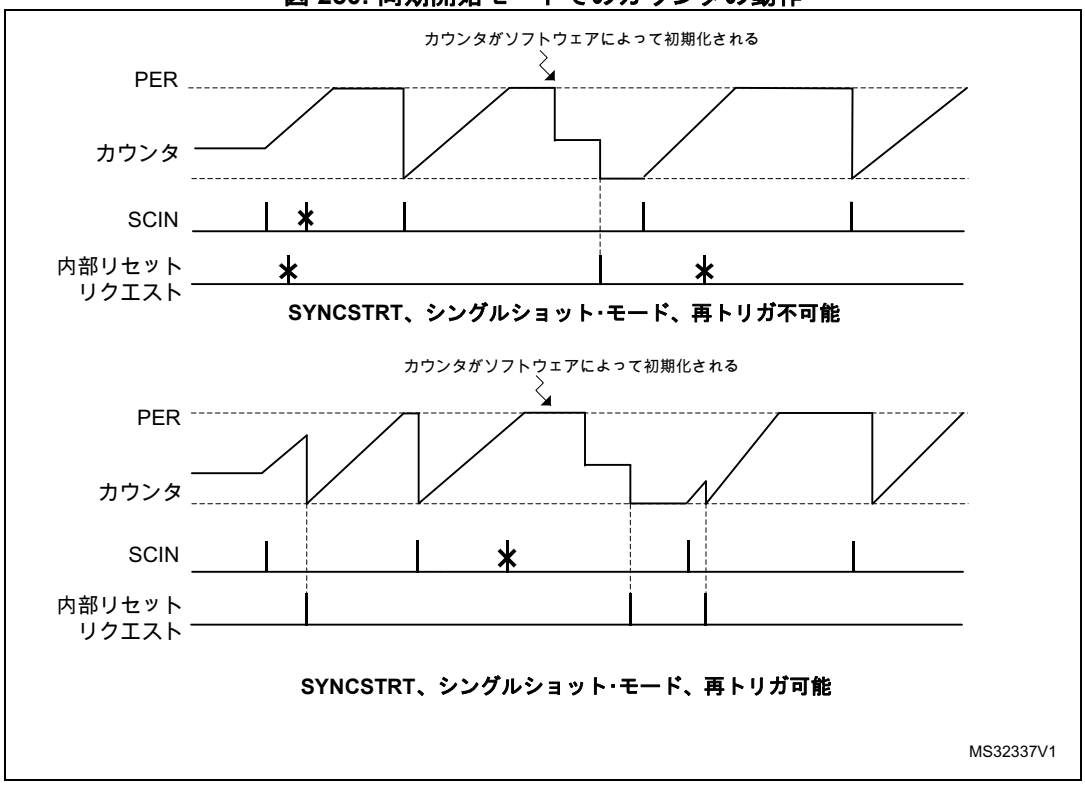
表 238. 同期イベントの効果とタイマの動作モードとの関係

動作モード	SYNC RSTx	SYNC STRTx	SYNC リセットまたは開始イベントの後の動作
シングルショット再トリガ不可能	0	1	開始イベントは、カウンタが停止していて、かつ、以下の場合に、考慮されます。 – MCEN または TxCEN ビットがセットされた後 – 周期に達した後 カウンタが周期値で停止しているときに起きる開始はカウンタをリセットします。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません（カウンタは同期によってのみ、再開できます）。カウント中に起きるリセットは無視されます（通常の再トリガ不可能モード中と同様）。
	1	X	リセットイベントはタイマカウントを開始します。カウンタが停止していて、かつ、以下の場合にのみ、考慮されます。 – MCEN または TxCEN ビットがセットされた後 – 周期に達した後 複数のリセットリクエストが選択されたとき（HRTIM_SCIN から、および内部イベントから）、最初に到着したリクエストのみが考慮されます。
シングルショット再トリガ可能	0	1	カウンタ開始は、カウンタが開始していない場合、または周期が経過している場合にのみ、有効です。カウンタ開始後に起きる同期イベントは効果がありません。 カウンタが周期値で停止しているときに起きる開始はカウンタをリセットします。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません（カウンタは同期によってのみ、開始できます）。カウント中に起きるリセットは考慮されます（通常の再トリガ可能モード中と同様）。
	1	X	HRTIM_SCIN からのリセットは、内部イベントからの HRTIM のタイマカウンタのリセットと同様に、考慮され、タイマのカウントを開始または再開します。 複数のリセットリクエストが選択されたとき、最初に到着したリクエストが考慮されます。
連続モード	0	1	タイマが有効化され（MCEN または TxCEN のビットセット）、カウンタを開始するための同期イベントを待ちます。カウンタ開始後に起きる同期イベントは効果がありません（カウンタは同期によってのみ、開始できます）。リセットリクエストはカウンタをクリアしますが、カウンタを開始しません。
	1	X	HRTIM_SCIN からのリセットは、内部イベントからの HRTIM のタイマカウンタのリセットと同様に、考慮され、タイマのカウントを開始または再開します。複数のリセットリクエストが選択されたとき、最初に到着したリクエストが考慮されます。

同期リセットイベントが周期イベントと同じ f_{HRTIM} クロックサイクル内で発生すると、このリセットはプログラムされた周期イベントまで延期されます（両方のイベントがカウンタロールオーバーを引き起こすため）。これは、高分解能がアクティブのとき（CKPSC[2:0] < 5）にのみ、適用されます。

図 250 に、シングルショットモードで同期開始がどのように行われるかを示します。

図 250. 同期開始モードでのカウンタの動作



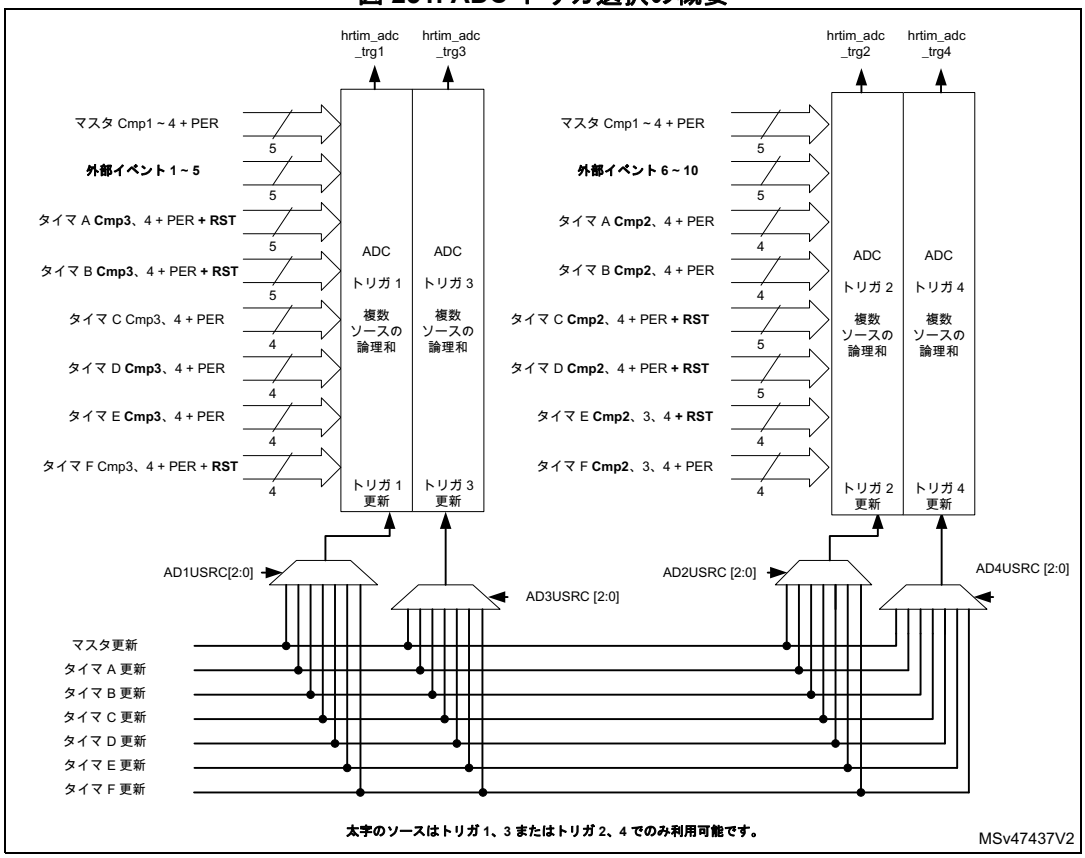
27.3.20 ADC トリガ

ADC は、マスタおよび 6 つのタイミングユニットによってトリガできます。

ADC のレギュラおよびインジェクトシーケンス両方のために、10 個の独立したトリガが利用できます。外部イベントをトリガとして使用できます。それらは、HRTIM_EECRx レジスタで定義された調整の直後に取り込まれ、EEFxR1 および EEFxR2 レジスタの設定に依存しません。

図 251 に示すように、HRTIM_ADC1R～HRTIM_ADC4R レジスタにおいて、ADC トリガ 1～4 に対して最大 32 のイベントを組み合わせる（論理和）ことができます。ADC トリガ 1/3 および 2/4 は同じソースセットを使用します。複数のソースを同時に選択することによって、1 つのスイッチング周期内で複数のトリガをかけることができます。典型的な使用事例は、単一の ADC トリガ出力を使用してすべての位相を連続的にサンプルできる、非オーバーラップ多相コンバータです。

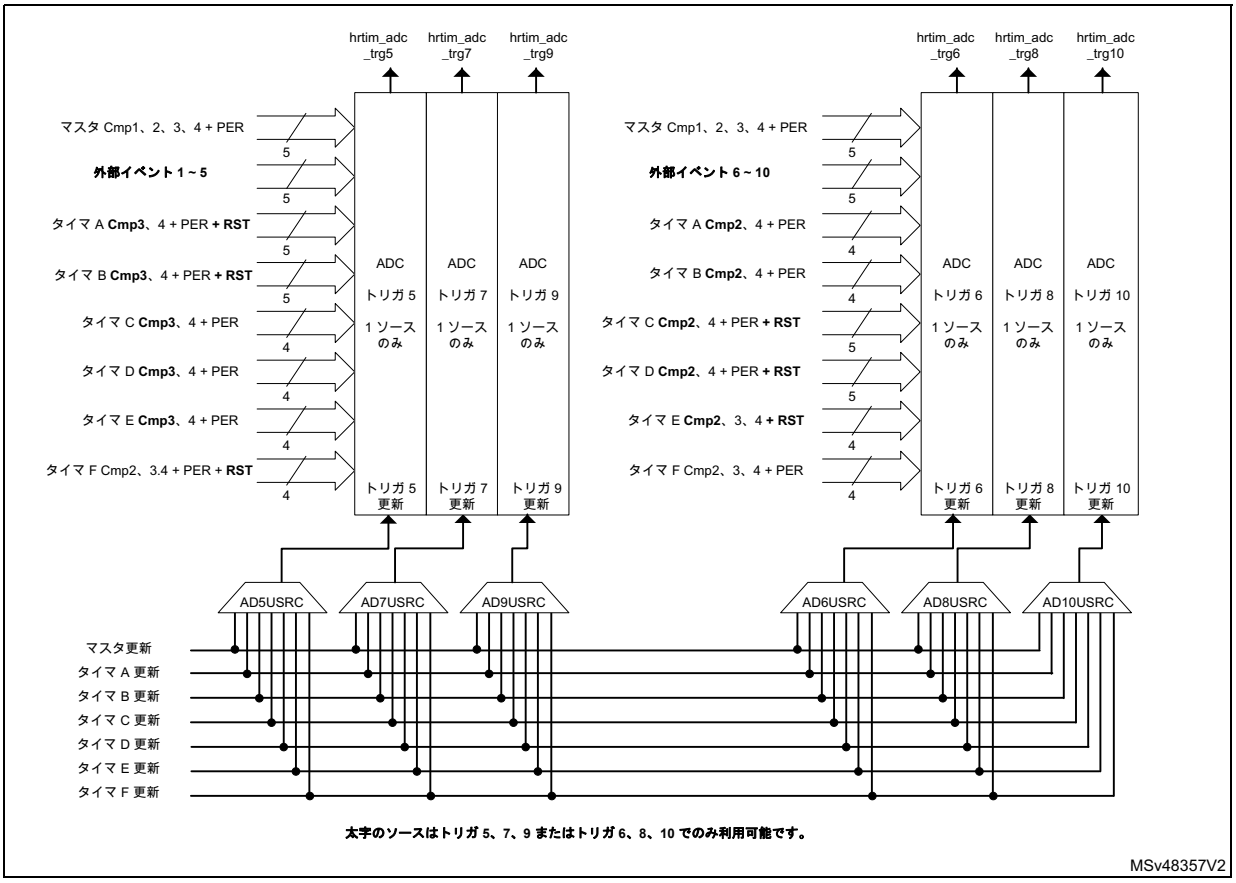
図 251. ADC トリガ選択の概要



ADC トリガ 5〜10 は、図 252 に示すように、HRTIM_ADCER レジスタで設定されます。ADC トリガ 5/7/9 および 6/8/10 は同じソースセットを使用します。

これらのトリガに対して、一度に1つのソースしか選択できません（32 個のイベントのうちの 1 つ）。

図 252. ADC トリガ



HRTIM_ADC1R～HRTIM_ADC4R および HRTIM_ADCER レジスタはプリロードされ、関連するタイマと同期して更新できます。更新ソースは、HRTIM_CR1 および HRTIM_ADCUR レジスタの ADxUSRC[2:0] ビットによって定義されます。

たとえば、ADC トリガ 1 がタイマ A CMP2 イベントを出力する場合（HRTIM_ADC1R = 0x0000 0400）、HRTIM_ADC1R は一般的にタイマ A（AD1USRC[2:0] = 001）と同時に更新されます。

ソースタイマでプリロードが無効になっている場合（PREEN ビットのリセット）、HRTIM_ADCxR レジスタもプリロードされません。書込みアクセスをするとトリガソースはすぐに更新されます。

ADC ポストスケーラ

下記の図 253 に示すように、ポストスケーリングユニットによって、ADC トリガレートを低減することができます。

各 ADC トリガレートは、HRTIM_ADCxPS1 および HRTIM_ADCxPS2 レジスタの ADCxPSC[4:0] ビットを使用して個別に調整できます。

センターアラインモードでは、ADC トリガレートは、図 254 に示すように、ソースタイマでプログラムされた ADROM[1:0] ビットフィールドにも依存します。ADROM[1:0] ビットフィールドは、ADC をトリガできるイベント（リセット、ロールオーバー（周期）、比較イベント）を以下のように規定しています。

- ADROM[1:0] = 00：アップおよびダウンカウントの両フェーズ中に生成されるイベント
- ADROM[1:0] = 01：ダウンカウントフェーズ中に生成されるイベント
- ADROM[1:0] = 10：アップカウントフェーズ中に生成されるイベント

ADC ポストスケーラのプログラミングレジスタはプリロードされ、タイマを停止することなく動作中に更新できます。

図 253. アップカウントモードでの ADC トリガのポストスケーリング

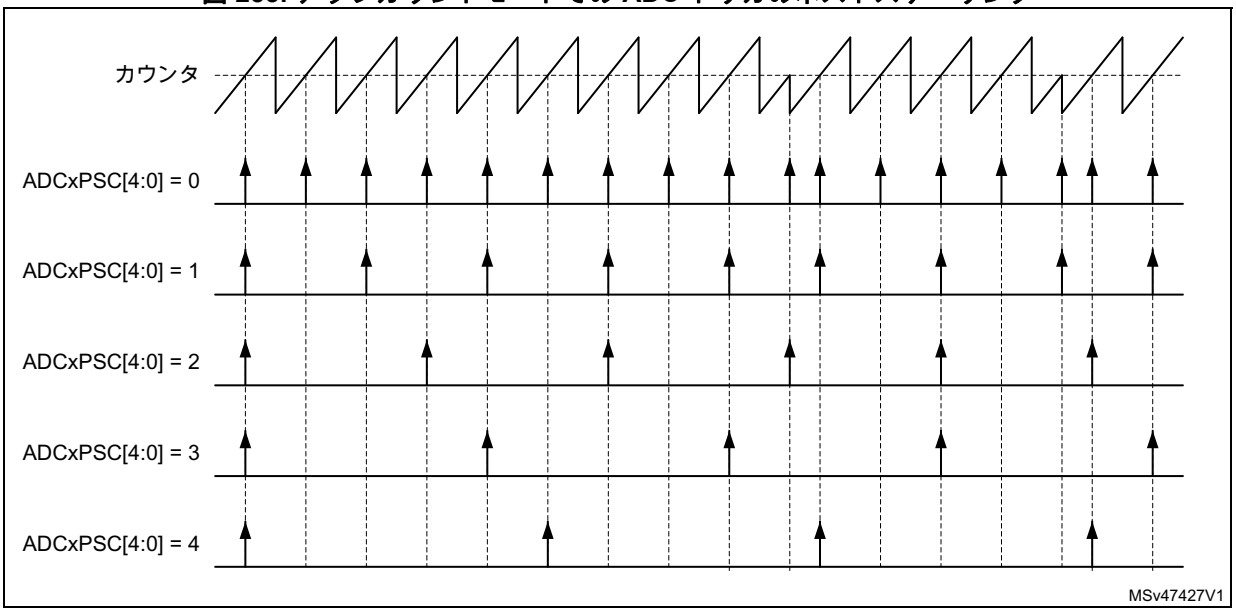
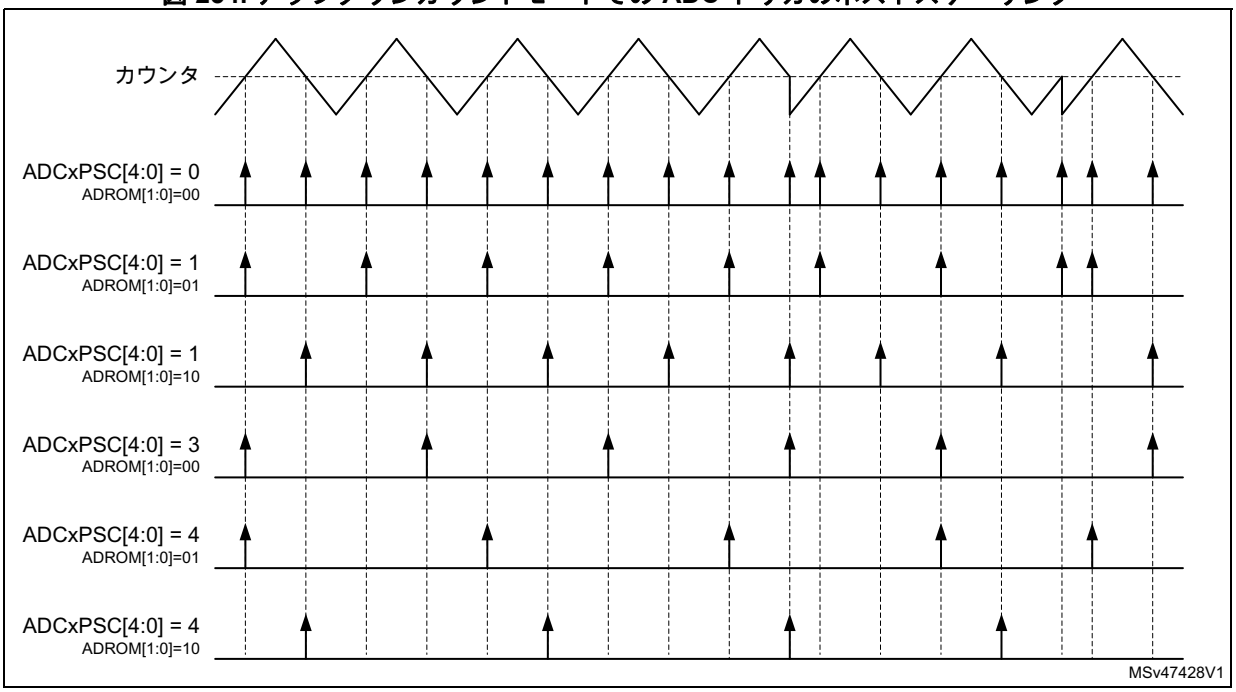


図 254. アップダウンカウントモードでの ADC トリガのポストスケーリング



27.3.21 DAC トリガ

HRTIM によって、内蔵 DAC をタイマの更新と同期して更新させることができます。

マスタタイマおよびタイマユニットからの更新イベントは、3 つの hrtim_dac_trgx 出力のいずれにも DAC 更新トリガを生成できます。

注： **各タイマには DAC 関連の独自の制御レジスタがあります。**

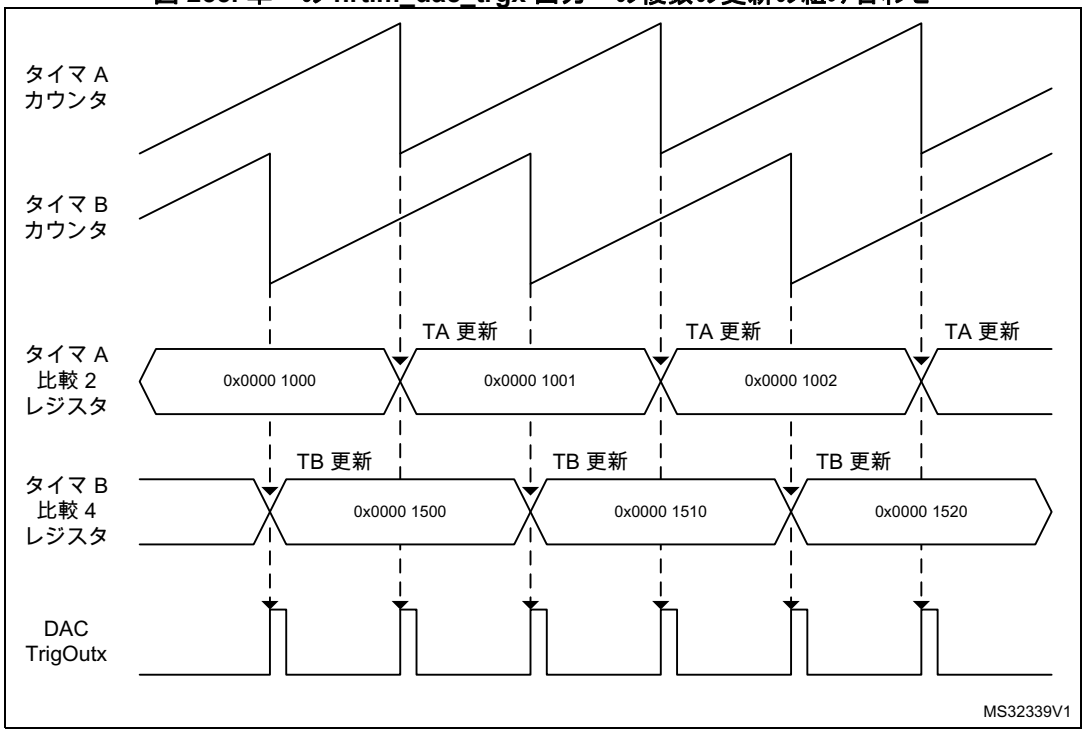
HRTIM_MCR および HRTIM_TIMxCR レジスタの DACSYNC[1:0] ビットは次のようにプログラムされます。

- 00：更新は生成されません。
- 01：更新が hrtim_dac_trg1 に生成されます。
- 10：更新が hrtim_dac_trg2 に生成されます。
- 11：更新が hrtim_dac_trg3 に生成されます。

f_{HRTIM} クロックの 1 周期の出力パルスが hrtim_dac_trgx 出力に生成されます。

DACSYNC[1:0] ビットが複数のタイマで有効にされると、hrtim_dac_trgx 出力はすべてのタイマの更新イベントの論理和から構成されます。たとえば、タイマ A とタイマ B で DACSYNC = 1 とすると、図 255 に示すように、タイマ A での更新イベントはタイマ B での更新イベントと論理和がとられ、対応する hrtim_dac_trgx 出力に DAC 更新トリガを生成します。

図 255. 単一の hrtim_dac_trgx 出力への複数の更新の組み合わせ



DAC への接続については、表 213：HRTIM DAC トリガ接続を参照してください。

デュアルチャネル DAC トリガ

HRTIM の内蔵機能と DAC のこぎり波ジェネレータを使用して、スロープ補正技術とヒステリシス制御を簡単に実装できます。原理は、DACを使用してPWM 周期に同期した減少するのこぎり波、または PWM 信号に同期した方形波を生成します。

このモードは、TIMxCR2 レジスタの DCDE ビットによって有効になります。このビットは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

以下の図 256 に示すように、次の 2 つのトリガ出力が使用されます。

- hrtim_dac_reset_trgx によって DAC リセット／更新イベントが生成されます。

- hrtim_dac_step_trgx によって DAC 値の増分変更のリクエストが生成されます。

hrtim_dac_reset_trgx トリガが生成されるタイミングは、TIMxCR2 レジスタの DCDR ビットによって次のように定義されます。

- DCDR = 0 : トリガは、カウンタのリセットまたはロールオーバーイベントによって生成されます。
- DCDR = 1 : トリガは、出力 1 のセットイベントによって生成されます。

注 : **DCDR ビットは、DCDE ビットがリセットされている（デュアルチャネル DAC トリガが無効の）ときは、意味がありません。**

hrtim_dac_step_trgx トリガが生成されるタイミングは、TIMxCR2 レジスタの DCDS ビットによって次のように定義されます。

- DCDS = 0 : トリガは、比較 2 のイベントによって生成されます。
- DCDS = 1 : トリガは、出力 1 のリセットイベントによって生成されます。

DCDR および DCDS ビットによって、以下の使用例を実現できます。

- エッジに合わせたスロープ補正（DCDR = DCDS = 0）: DAC ののこぎり波は PWM 周期の開始から始まり、周期中に複数のトリガが生成されます。
- センターに合わせたスロープ補正（DCDR = 1、DCDS = 0）: DAC ののこぎり波は 出力のセットイベントから始まり、周期中に複数のトリガが生成されます。
- ヒステリシスコントローラ : DAC の値は、出力状態が変化したときに、周期ごとに 2 回変更する必要があります。PWM 周期ごとに 2 つのトリガが生成されます。エッジアラインモード（DCDR=0、DCDS =1）では、トリガはカウンタのリセットまたはロールオーバーによって生成されます。センターアラインモード（DCDR=1、DCDS =1）では、トリガは、出力がセットされた時に生成されます。

DCDE がセットされ、DCDS ビットがリセットされていると、比較 2 は特定の動作モードを取ります。アクティブな比較値は、比較一致が発生するとすぐに自動的に更新されるため、図 256 に示すように、トリガは CMP2 値に等しい周期で定期的に繰り返されます。
DCDS ビットをリセットした（比較 2 イベントを使用）デュアルチャネル DAC トリガは、CMP2 を使用するモード（トリプル／クワッドインタリーブおよびトリガハーフモード）と同時に使用しないでください。

注 : **CMP2 値は動作中に変更することができます。新しい値は次の比較一致で有効になります。**

注 : **DCDS ビットがリセットされているとき、CMP2 値は他のメカニズムによって変更されてはなりません。すなわち、インタリーブ、トリガハーフ、およびバランスアイドルモードを無効にする必要があります。**

下記の表 239 に、PWM 周期内に 6 つのトリガを生成する例を示します。除算結果を上位数値に丸める必要があることが示されています。

TIMxPER = 8192 のカウンタ周期を考えましょう。8192 を 6 で割ると 1365.33 になります。

– 切り下げ値：1365：7 つのトリガが生成され、6 番目と 7 番目はきわめて短い間隔で発生します（それぞれのカウンタは 8190 と 8192 です）。

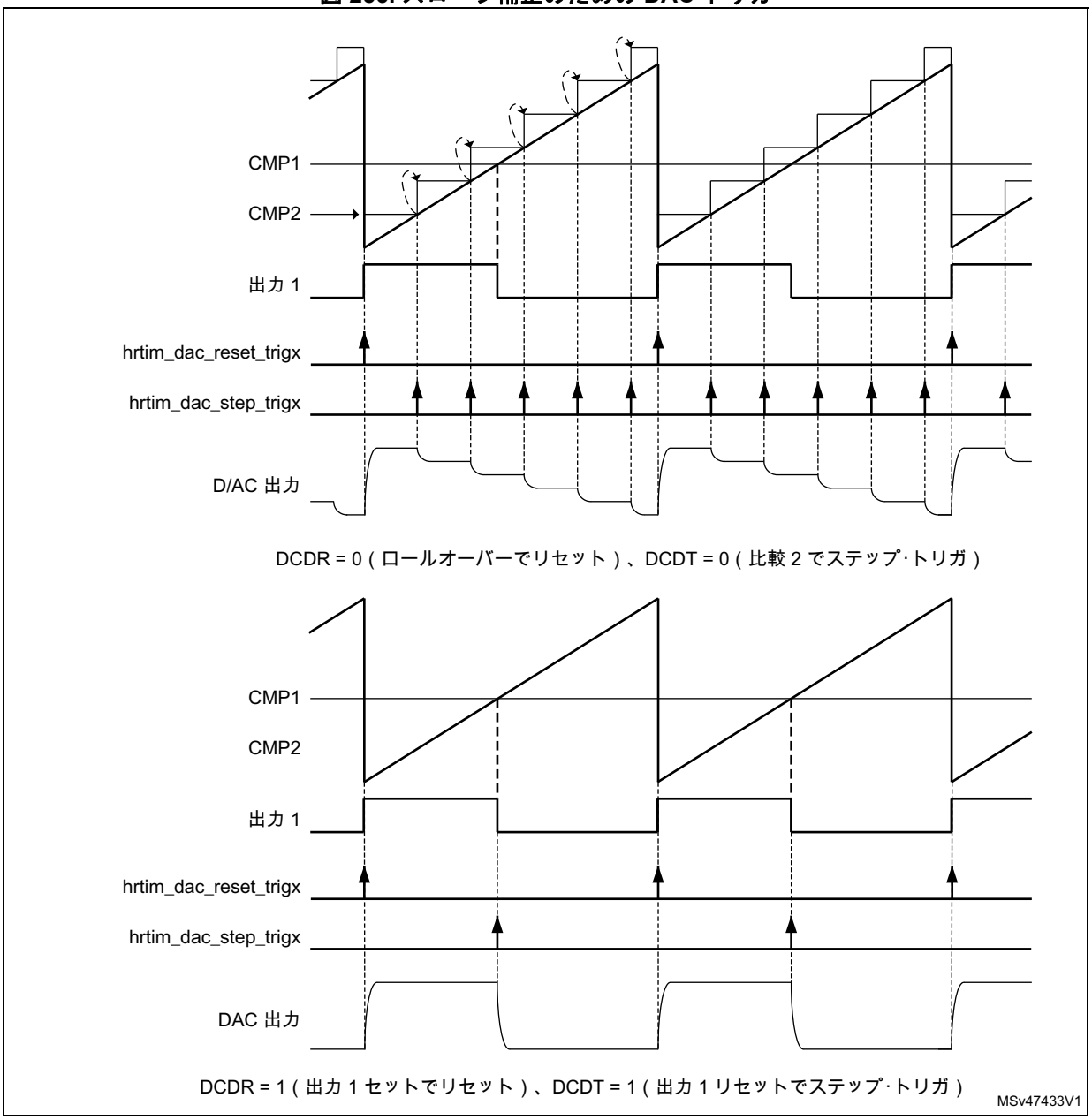
– 切り上げ値：1366：6 つのトリガが生成されます。dac_step_trg の 6 番目のトリガ（カウンタ = 8192）は、カウンタロールオーバー（8192 から 0）によって中止されます。

表 239. DAC デュアルチャネルトリガの例

-	CMP2 = 1365	dac_trg	dac_step_trg	CMP2 = 1366	dac_trg	dac_step_trg
カウンタ値	1365	-	1	1366	-	1
	2730	-	2	2732	-	2
	4095	-	3	4098	-	3
	5460	-	4	5464	-	4
	6825	-	5	6830	-	5
	8190	-	6	8192	6	-
	8192	7	-	1366	-	1
	1365	-	1	2732	-	2
	.....	-	-	.....	-	-

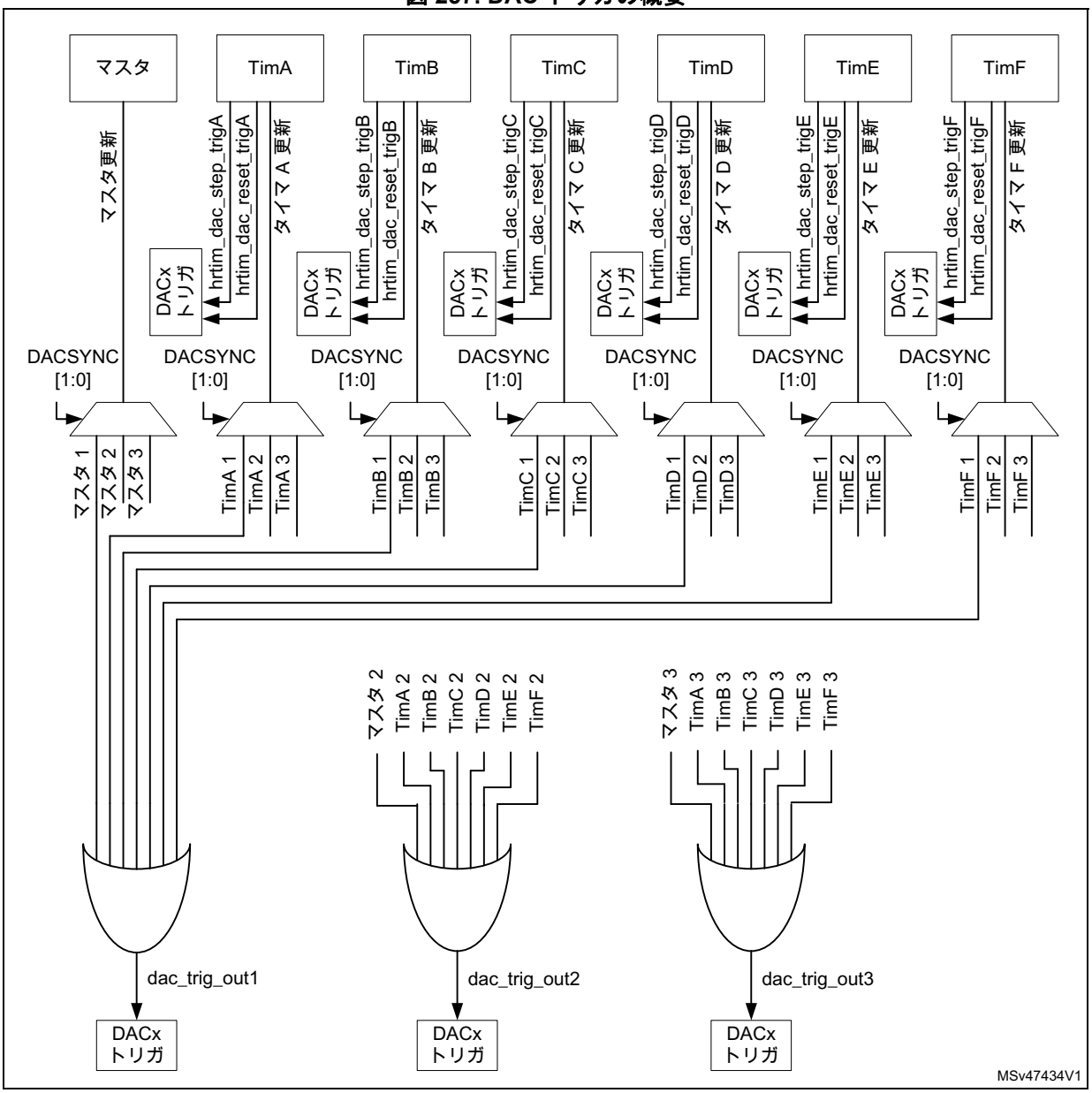
注： **センターパターンモードでは、カウンタのピーク値の周りに不均一な間隔でトリガが発生しないように、スイッチング周期あたり偶数のトリガを設定する必要があります。**

図 256. スロープ補正のための DAC トリガ



以下の図 257 に、使用できるすべての DAC トリガの概要を示します。

図 257. DAC トリガの概要



27.3.22 割込み

以下の 7 つの割込みがマスタタイマによって生成できます。

- マスタタイマのレジスタ更新
- 受信した同期イベント
- マスタタイマの繰り返しイベント
- マスタ比較 1～4 イベント

以下の 14 個の割込みが各タイミングユニットによって生成できます。

- トリガされた遅延保護
- カウンタのリセットまたはロールオーバーイベント
- 出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）
- 出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）
- 比較 1 および 2 のイベント
- タイミングユニットのレジスタ更新
- 繰り返しイベント
- 比較 1～4 イベント

HRTIM 全体に対して、8 つのグローバル割込みが生成されます。

- システム障害およびフォールト 1～6（タイミングユニットの属性にかかわらず）
- DLL 較正済み
- バーストモード周期完了

割込みリクエストは以下のように 8 つのベクトルにグループ分けされます。

- hrtim_it1：マスタタイマ割込み（マスタ更新、同期入力、繰り返し、MCMP1～4）、およびフォールトを除くグローバル割込み（バーストモード周期および DLL レディ割込み）
- hrtim_it2：TIMA 割込み
- hrtim_it3：TIMB 割込み
- hrtim_it4：TIMC 割込み
- hrtim_it5：TIMD 割込み
- hrtim_it6：TIME 割込み
- hrtim_it7：TIMF 割込み
- hrtim_it8：高優先順位割込み処理を可能にするすべてのフォールト割込みの専用ベクタ

表 240 に、割込みリクエスト、そのマッピングと関連制御およびステータスビットをまとめます。

表 240. HRTIM 割込みの概要

割込みベクタ	割込みイベント	イベントフラグ	有効制御ビット	フラグクリアビット
hrtim_it1	バースト･モード周期完了	BMPER	BMPERIE	BMPERC
	DLL 較正済み	DLLRDY	DLLRDYIE	DLLRDYC
	マスタタイマのレジスタ更新	MUPD	MUPDIE	MUPDC
	受信した同期イベント	SYNC	SYNCIE	SYNCC
	マスタ･タイマの繰り返しイベント	MREP	MREPIE	MREPC
	マスタ比較 1～4 イベント	MCMP1	MCMP1IE	MCP1C
		MCMP2	MCMP2IE	MCP2C
		MCMP3	MCMP3IE	MCP3C
		MCMP4	MCMP4IE	MCP4C
hrtim_it2 hrtim_it3 hrtim_it4 hrtim_it5 hrtim_it6h rtim_it7	トリガされた遅延保護	DLYPRT	DLYPRTIE	DLYPRTC
	カウンタのリセットまたはロールオーバーイベント	RST	RSTIE	RSTC
	出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）	RSTx1	RSTx1IE	RSTx1C
		RSTx2	RSTx2IE	RSTx2C
	出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）	SETx1	SETx1IE	SETx1C
		SETx2	SETx2IE	SETx2C
	比較 1 および 2 のイベント	CPT1	CPT1IE	CPT1C
		CPT2	CPT2IE	CPT2C
	タイミング･ユニットのレジスタ更新	UPD	UPDIE	UPDC
	繰り返しイベント	REP	REPIE	REPC
	比較 1～4 イベント	CMP1	CMP1IE	CMP1C
		CMP2	CMP2IE	CMP2C
		CMP3	CMP3IE	CMP3C
		CMP4	CMP4IE	CMP4C
hrtim_it8	システム障害	SYSFLT	SYSFLTIE	SYSFLTC
	フォールト 1～6	FLT1	FLT1IE	FLT1C
		FLT2	FLT2IE	FLT2C
		FLT3	FLT3IE	FLT3C
		FLT4	FLT4IE	FLT4C
		FLT5	FLT5IE	FLT5C
		FLT6	FLT6IE	FLT6C

27.3.23 DMA

割込みを生成可能なほとんどのイベントは DMA リクエストも生成できます。その両方を同時に生成することもできます。各タイマ（マスタ、TIMA～F）には独自の DMA イネーブルレジスタがあります。

個々の DMA リクエストは論理和がとられて以下のように 7 つのチャネルに入ります。

- マスタタイマ用に 1 チャネル
- タイミングユニット（TIMA～F）ごとに 1 チャネル

注： **DMA チャネルを無効に（TIMxDIER で DMA イネーブルビットをリセット）する前に、まず DMA コントローラを無効にする必要があります。**

表 241 は、イベントおよびそれと関連する DMA イネーブルビットの概要です。

表 241. HRTIM の DMA リクエストの概要

DMA チャネル	イベント	DMA 可能	DMA 有効
hrtim_dma1（マスタ･タイマ）	バーストモード周期完了	不可	N/A
	DLL 較正済み	不可	N/A
	マスタタイマのレジスタ更新	可能	MUPDDE
	受信した同期イベント	可能	SYNCDE
	マスタ･タイマの繰り返しイベント	可能	MREPDE
	マスタ比較 1～4 イベント	可能	MCMP1DE
		可能	MCMP2DE
		可能	MCMP3DE
		可能	MCMP4DE
hrtim_dma2（タイマ A） hrtim_dma3（タイマ B） hrtim_dma4（タイマ C） hrtim_dma5（タイマ D） hrtim_dma6（タイマ E） hrtim_dma7（タイマ F）	トリガされた遅延保護	可能	DLYPRTDE
	カウンタのリセットまたはロールオーバーイベント	可能	RSTDE
	出力 1 および出力 2 のリセット（アクティブから非アクティブへの遷移）	可能	RSTx1DE
		可能	RSTx2DE
	出力 1 および出力 2 のセット（非アクティブからアクティブへの遷移）	可能	SETx1DE
		可能	SETx2DE
	比較 1 および 2 のイベント	可能	CPT1DE
		可能	CPT2DE
	タイミング･ユニットのレジスタ更新	可能	UPDDE
	繰り返しイベント	可能	REPDE
	比較 1～4 イベント	可能	CMP1DE
		可能	CMP2DE
		可能	CMP3DE
		可能	CMP4DE
N/A	システム障害	不可	N/A
	フォールト 1～6	不可	N/A
	バーストモード周期完了	不可	N/A
	DLL 較正済み	不可	N/A

バースト DMA 転送

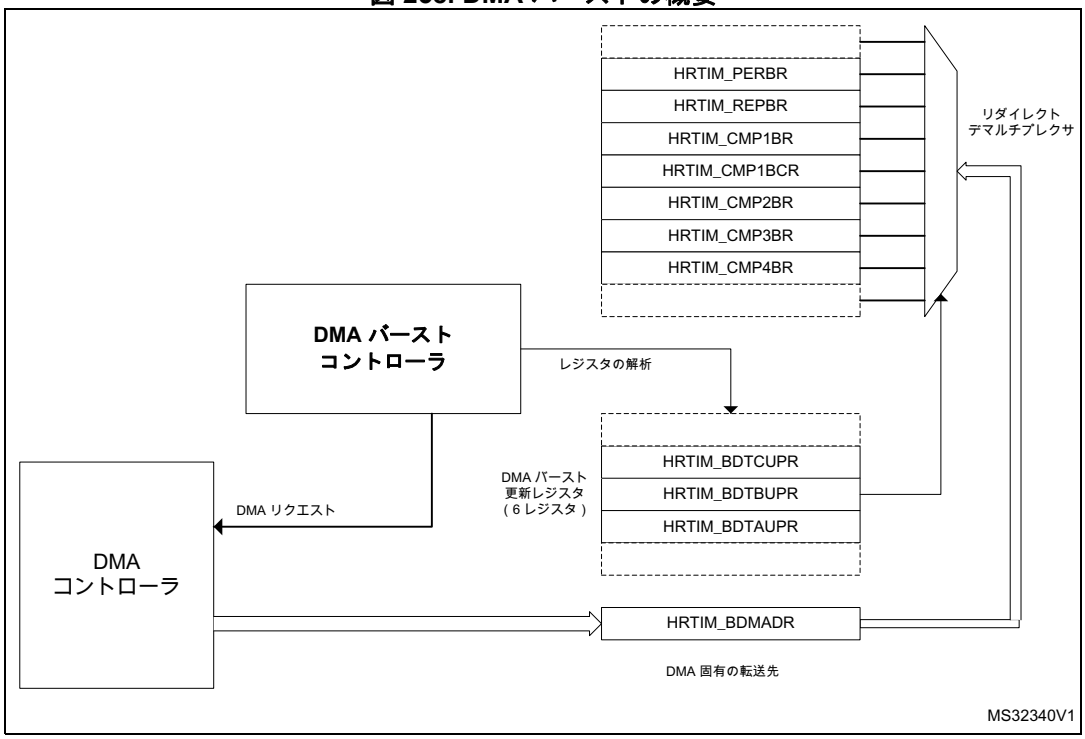
標準の DMA リクエストに加え、HRTIM は、1 つの DMA リクエストによって複数のレジスタを更新する DMA バーストコントローラを備えています。これによって次のことが可能です。

- 1 つの DMA チャネルのみで複数データレジスタを更新する。
- 複数のタイマ出力を使用するコンバータのために、1 つ以上のタイミングユニットを動的にプログラムする。

バースト DMA 機能は 1 つの DMA チャネルのみで使用可能ですが、6 つのチャネルのどれでもバースト DMA 転送用に選択できます。

原則は、どのレジスタが DMA によって書き込まれるかをプログラムすることです。マスタタイマおよび TIMA～E にはバースト DMA 更新レジスタがあり、そこで大部分の制御およびデータレジスタが次の選択ビットで関連付けられています：HRTIM_BDMUPR、HRTIM_BDTAUPR～HRTIM_BDTEUPR（これは書込みアクセス可能なレジスタのみに適用できます）。図 258 に示すように、リダイレクトメカニズムによって、DMA 書込みアクセスを HRTIM レジスタに自動的に転送することができます。

図 258. DMA バーストの概要



DMA トリガが発生すると、HRTIM は複数の 32 ビットの DMA リクエストを生成し、更新レジスタを解析します。制御ビットがセットされている場合、書込みアクセスは関連するレジスタにリダイレクトされます。ビットがリセットされている場合、レジスタ更新はスキップされ、新しいリクエストをトリガするためにレジスタの解析は新しいビットのセットが検出されるまで再開されます。6 つの更新レジスタ（HRTIM_BDMUPR、5 x HRTIM_BDTxUPR）の解析が終わると、バーストは完了し、システムは別の DMA トリガの受け入れが可能になります（図 259 のフローチャート参照）。

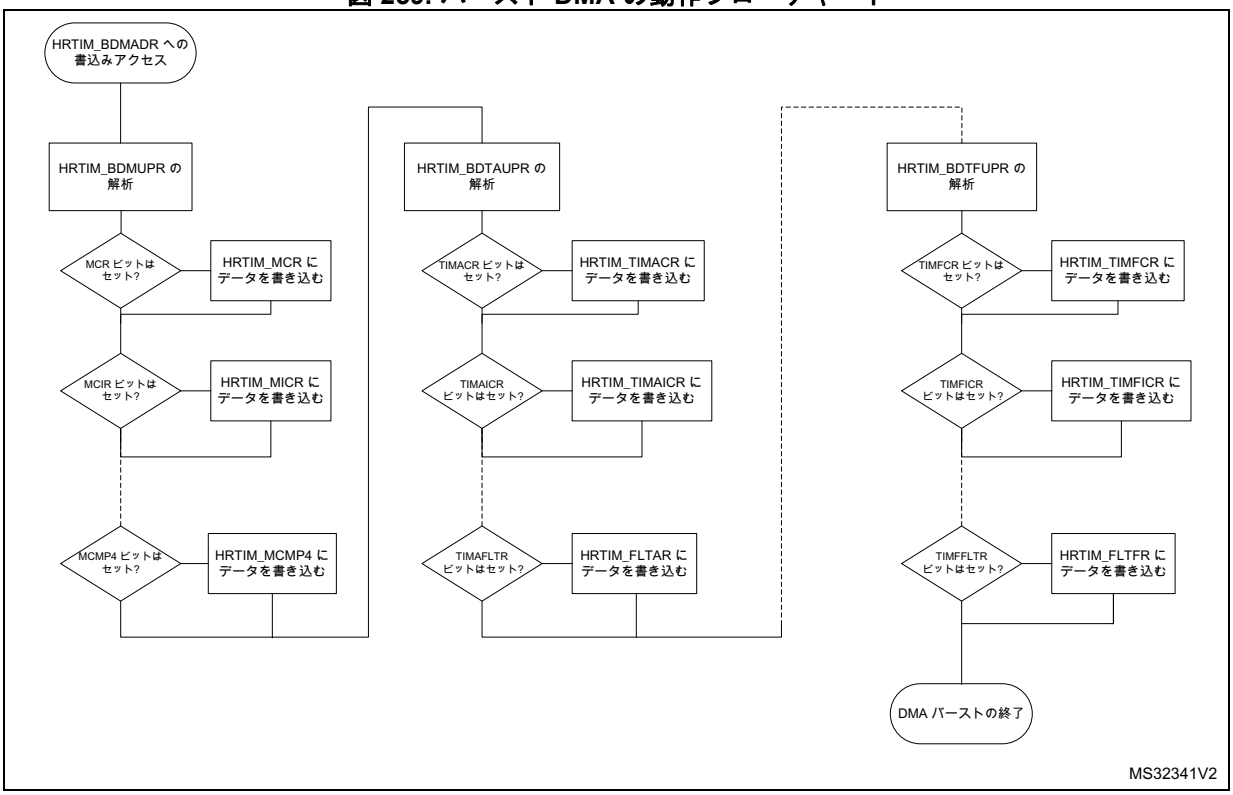
注： **バーストが進行中に起きるトリガは、一番最後のデータ転送中に起きた場合を除き、破棄されます。**

バースト DMA モードは永続的に有効です（イネーブルビットはありません）。HRTIM_BDMADR レジスタへの初回書込みアクセスで、バースト DMA 動作が開始します。

必要なことは、ペリフェラルアドレスのインクリメントモードを無効にしてメモリ上のペリフェラル設定を HRTIM_BDMADR レジスタへ転送するようにDMAコントローラを設定するだけです（HRTIM が最終転送先レジスタへデータを送る処理を内部で行います）。

バースト DMA モードがトランザクション中に中断された場合に、それを再初期化するには、6 つの更新レジスタのうち少なくとも 1 つに書き込む必要があります。

図 259. バースト DMA の動作フローチャート



DMA バーストが完了した後、レジスタの更新方法に応じて、いくつかのオプションが利用可能です。

PREEN ビットがリセット（プリロードは無効）されている場合、DMA によって書き込まれた値は直ちにアクティブレジスタへ転送され、レジスタは、DMA のトランザクションのペースに従って、順次更新されます。

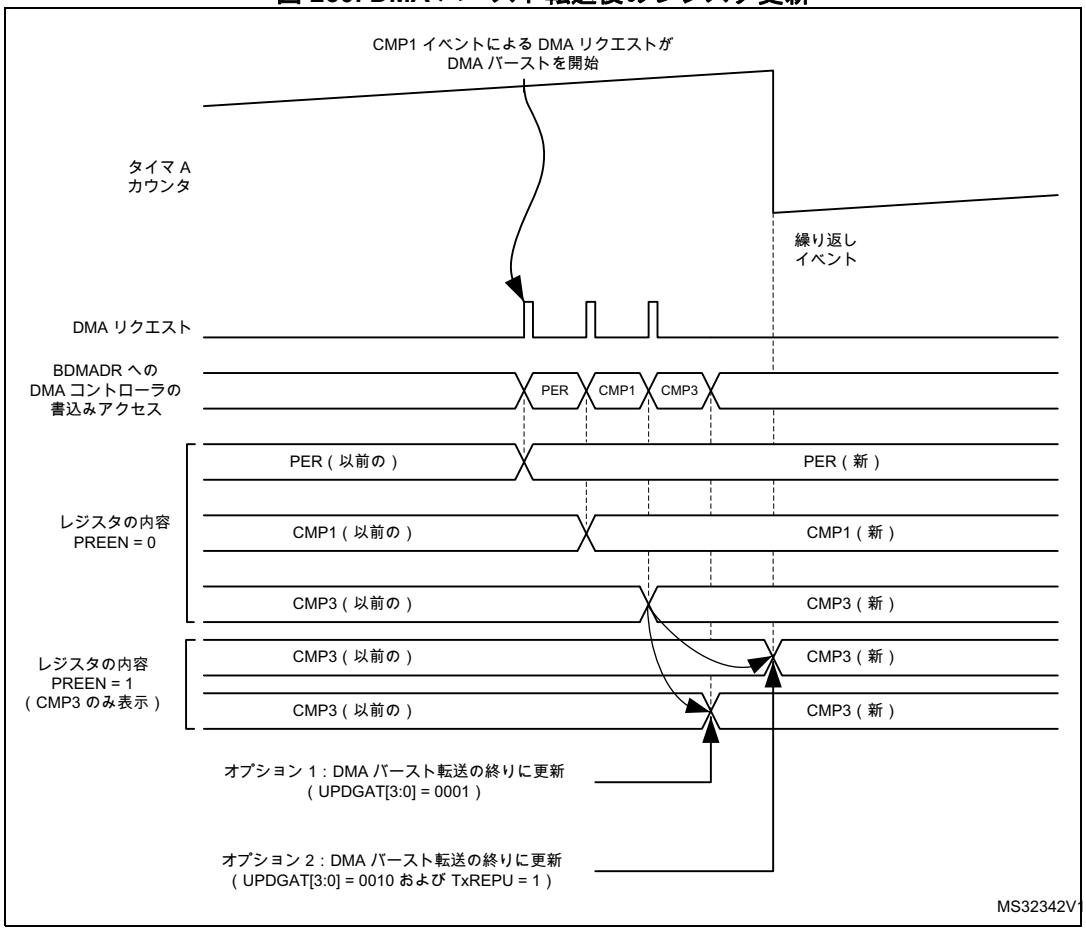
プリロードが有効になっている（PREEN ビットのセット）ときは、以下の 3 つの使用事例があります。

1. 更新は DMA バースト転送とは独立に行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0000 および HRTIM_MCR の BRSTDMA[1:0] = 00）。この場合、転送されたすべてのデータを同時に考慮させる必要がある場合、ユーザは、更新イベントが起きる前に DMA バーストが完了していることをチェックする必要があります。反対に、DMA 転送が進行中に更新イベントが起きた場合、レジスタの一部のみがロードされ、完全にレジスタを更新するには 2 つの連続する更新イベントが必要です。
2. 更新は DMA バースト転送が完了したときに行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0000 および HRTIM_MCR の BRSTDMA[1:0] = 01）。このモードでは、すべての新しいレジスタ値が同時に転送されることが保証されます。これはカウンタ値とは独立に行われ、必要なら、通常の更新イベントと組み合わせることができます（たとえば、TxRSTU がセットされているとき、カウンタリセット時に更新）。
3. 更新は DMA バースト転送が完了した後、更新イベント時に行われます（HRTIM_TIMxCR の UPDGAT[3:0] = 0010 および HRTIM_MCR の BRSTDMA[1:0] = 10）。このモードでは、転送さ

れたすべてのデータの首尾一貫した更新、およびタイマカウンタを使用した、通常の更新イベントとの同期の両方が保証されます。この場合、転送が進行中に通常の更新リクエストが起きた場合、それは破棄され、次に来る更新リクエスト時に有効な更新が起きます。

図 260 の時間経過図に、次の 3 つのケースに対するアクティブレジスタの内容を示します：PREEN=0、UPDGAT[3:0] = 0001 、および UPDGAT[3:0] = 0001（PREEN = 1 のとき）。

図 260. DMA バースト転送後のレジスタ更新



27.3.24 HRTIM の初期化

このセクションでは、MCU の他の関連ペリフェラルも含めて、HRTIM の推奨の初期化手順について説明します。

HRTIM クロックソースは、DLL ロックの f_{HRTIM} 範囲を考慮しながら、リセットおよびクロック制御ユニット（RCC）において有効にする必要があります。

DLL 較正は、HRTIM_DLLCR レジスタの CAL ビットをセットして開始しなければなりません。

HRTIM マスタおよびタイミングユニットは、高分解能ユニットの準備ができて初めて起動できます。これは DLLRDY フラグのセットで示されます。DLLRDY フラグは、初期化を再開する前にポーリングできます。または HRTIM や他の MCU ペリフェラルの他のレジスタが初期化されている間に、較正をバックグラウンドで実行できます。この場合、カウンタを開始する前に DLLRDY フラグを確認する必要があります（必要に応じて、HRTIM_IER の DLLRDYIE フラグで有効にして較正終了割込みを発行できます）。DLL 較正が完了したら、CALEN ビットをセットして較正を定期的に実行し、潜在

的な電圧および温度ドリフトを補正する必要があります。較正の周期性は、HRTIM_DLLCR レジスタの CALRTE[1:0] フィールドを使用して定義します。

HRTIM 制御レジスタは電力コンバータの接続形態やタイミングユニットの使用事例に従って初期化することができます。すべての入力を設定する必要があります（ソース、極性、エッジ対応）。

最終的には次の順序で HRTIM 出力を設定する必要があります。

- 極性は HRTIM_OUTxR の POLx ビットを使用して定義する必要があります。
- FAULT および IDLE 状態は HRTIM_OUTxR の FAULTx[1:0] および IDLESx ビットを使用して設定する必要があります。

HRTIM 出力が MCU の I/O 類に接続可能となります。GPIO コントローラにおいて、選択された HRTIM の I/O 類を製品データシートのオルタネート機能配置表によって設定する必要があります。

この時点から、HRTIM は IDLE 状態にある出力を制御します。

出力は、HRTIM_OENR レジスタの TxyOEN ビットをセットすることによって、RUN モードに設定されます。2 つの出力は、RUN モードでの最初の有効なセット／リセットイベントまで、非アクティブ状態です。TxCEN ビットがリセットされている限り、いかなる出力セット／リセットイベントも（SST、SRT を使用したソフトウェアリクエストを除き）無視されます。バーストモードリクエストも同じく無視されます（IDLEM ビットの値は無視されます）。同様に、バーストモードコントローラから来るカウンタリセットリクエストも無視されます（TxBM ビットがセットされている場合）。

注： **デッドタイム挿入が有効（DTEN ビットのセット）になっているとき、RUN モードに入るとすぐに出力が相補状態になるように、SST および RST ビットを使用して、ソフトウェアによって出力状態を強制する必要があります。**

HRTIM 動作は、最終的に HRTIM_MCR の TxCEN または MCEN ビットをセットすることによって開始することができます。

HRTIM ペリフェラルが、リセットおよびクロックコントローラによってリセットされると、出力制御は GPIO コントローラに解放され、出力はトライステートになります。

27.3.25 デバッグ

マイクロコントローラがデバッグモードになると（FPU 搭載 Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBG モジュールの下記 DBG_HRTIM_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

- DBG_HRTIM_STOP = 0：動作の変更はなく、HRTIM は動作を続けます。
- DBG_HRTIM_STOP = 1：マスタタイマを含め、すべての HRTIM タイマは停止します。RUN モードの出力は、FAULTx[1:0] = 01、10、11 の場合、FAULT 状態になるか、または FAULTx[1:0] = 00 の場合、現在の状態を保ちます。アイドル状態の出力はその状態を維持します。これは、MCU が停止モードを出た場合も、永続的に維持されます。これによって、ステップ実行中に安全な状態を維持することができます。出力は、TxyOEN ビットをセットすることによって（デバッガの使用を要求）、再び有効化できます。

DBG_HRTIM_STOP = 1 のとき、MCU 停止中のタイマ動作

セット／リセットクロスバー、デッドタイムおよびプッシュプルユニット、アイドル／バランス障害検出、および RUN モードで通常の出力を駆動しているすべてのロジックはデバッグの影響を受けません。出力は、（MCU の停止中またはその後）TxyOEN が再びセットされる時に出力の通常信号を回復できるように、内部的にトグルを維持します。出力が無効になっているとき、関連するトリガおよびフィルタも内部波形に従っています。

FAULT 入力およびイベント（どのソースも）は、MCU の停止中、有効になっています。

MCU の停止中に、フォールトがその時に発生した場合、フォールトステータスビットはセットでき、TxyOEN ビットはリセットできます（TxyOEN および TxyODS は DBG_HRTIM_ストップビット状態の影響を受けません）。

キャプチャイベントのみならず、同期、カウンタリセット、開始、およびリセット開始イベントは、デバッグモードでは破棄されます。これは、MCU が停止している限り、関連するすべてのレジスタを安定に保つためです。

ブレークポイントに達するとカウンタはカウントを停止します。しかし、カウンタイネーブル信号はリセットされません。したがって、デバッグから出た時、開始イベントは発行されません。外部イベントのみならず、すべてのカウンタリセットおよびキャプチャトリガは無効になります（MCU が停止している限り、無視されます）。強制ソフトウェアセット／リセットの場合を除き、出力 SET および RST フラグは凍結されます。レベル対応のイベントは、デバッグ中はマスクされますが、デバッグが終了するとすぐに、再びアクティブになります。エッジ対応のイベントに対しては、MCU が停止中に信号がアクティブに維持される場合は、デバッグを終了する時、新しいエッジは生成されません。

更新イベントは破棄されます。これによって、hrtim_upd_en[3:1] 入力への更新トリガが防止されます。DMA トリガは無効になります。バーストモード回路は停止します：トリガは無視され、バーストモードカウンタは停止します。

DLL 較正は、MCU が停止している間、ブロックされません（DLLRDY フラグはセットできます）。

27.4 アプリケーションの使用事例

27.4.1 降圧コンバータ

降圧コンバータはステップダウンコンバータとして一般的に使用されます。HRTIM は 7 つの独立なスイッチング周波数を使って最大 12 個の降圧コンバータを制御できます。

コンバータは通常、固定の周波数で動作し、Vin/Vout 比は次のように電源スイッチに適用されるデューティサイクル D に依存します。

$$V_{out} = D \times V_{in}$$

接続形態は図 261 に示すとおりで、電圧の読みのために ADC へ接続されています。

図 261. 降圧コンバータの接続形態

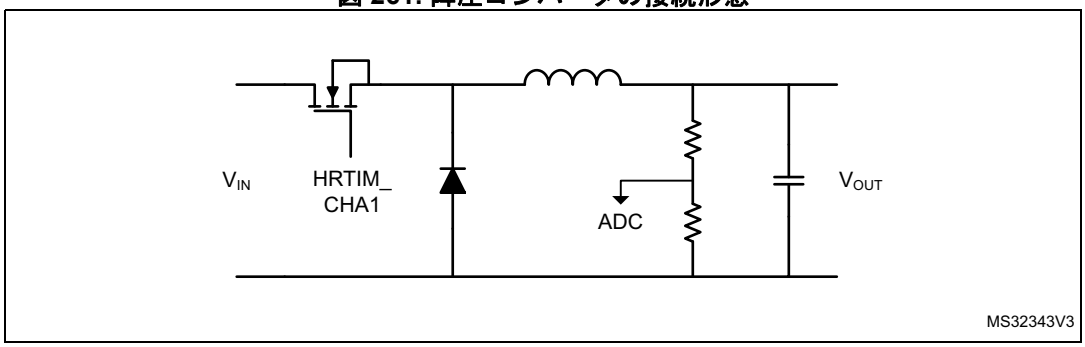
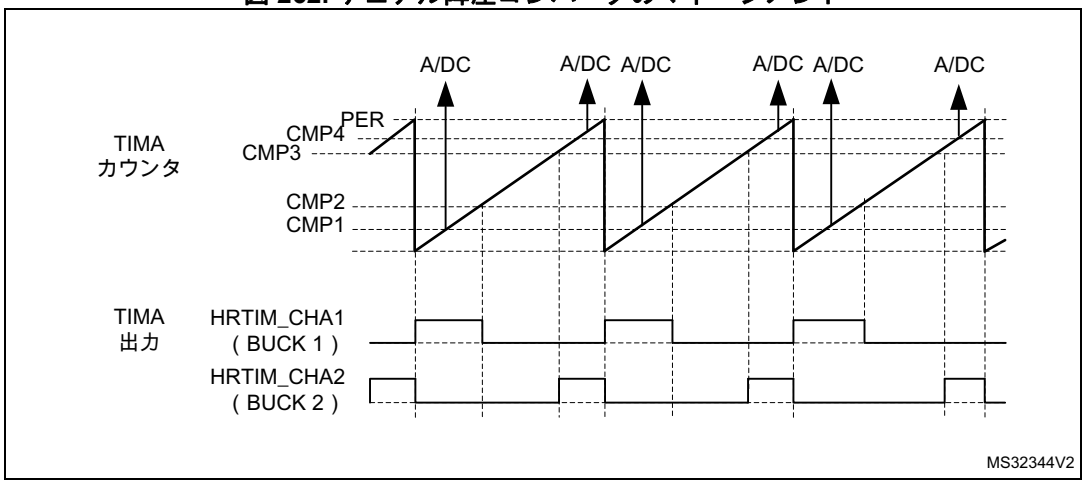


図 262 に、同じ周波数の PWM 信号を使用した 2 つのコンバータのマネージメントを示します。出力は、次のように定義されています。

- HRTIM_CHA1 は周期でセット、CMP1 でリセット
- HRTIM_CHA2 は CMP3 でセット、PER でリセット

CMP2 および CMP4 イベントを使用して、ADC は、正確に ON タイムの中央で周期あたり 2 回トリガされています。

図 262. デュアル降圧コンバータのマネージメント

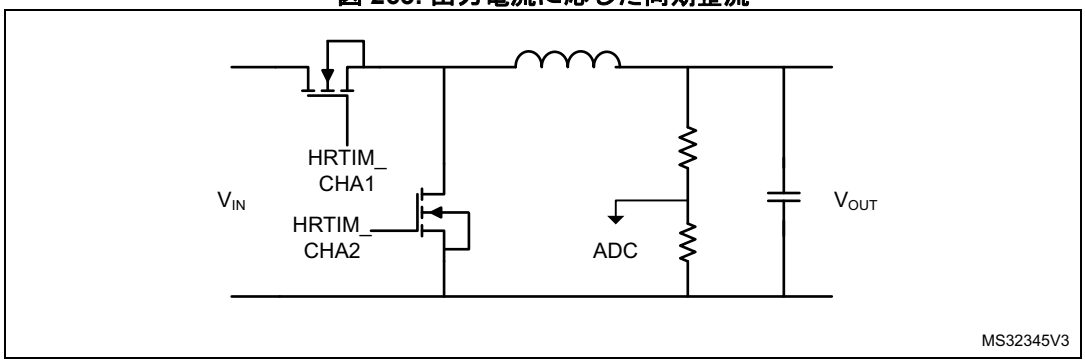


タイマ A～E は、ペア（両方とも同じスイッチング周波数）に結合された 12 個の降圧コンバータ、または、マスタタイマを 7 番目のタイムベースとして使用して 7 個の完全に独立したコンバータ（それぞれが異なるスイッチング周波数を持つ）を提供できます。

27.4.2 同期整流式の降圧コンバータ

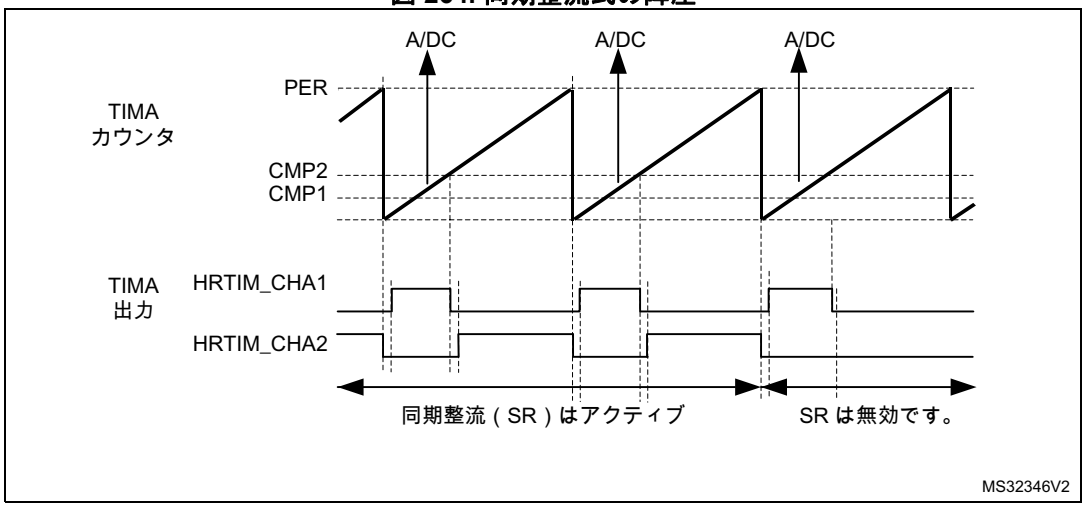
同期整流によって、フリーホイーリングダイオードに代わる FET を使って降圧コンバータの損失を最小にすることができます。同期整流は、図 263 に示すように、出力の電流レベルに応じて臨機応変にオンまたはオフにできます。

図 263. 出力電流に応じた同期整流



シングルスイッチの降圧コンバータに対する主な違いは、HRTIM_CHA1 の基準波形に基づいて HRTIM_CHA2 上にほとんど相補的な波形を生成するためにデッドタイムを付加することです（図 264 参照）。

図 264. 同期整流式の降圧



27.4.3 多相コンバータ

多相技術は複数の電力変換接続形態（降圧、フライバック）に適用できます。これらの主要な利点は以下のとおりです。

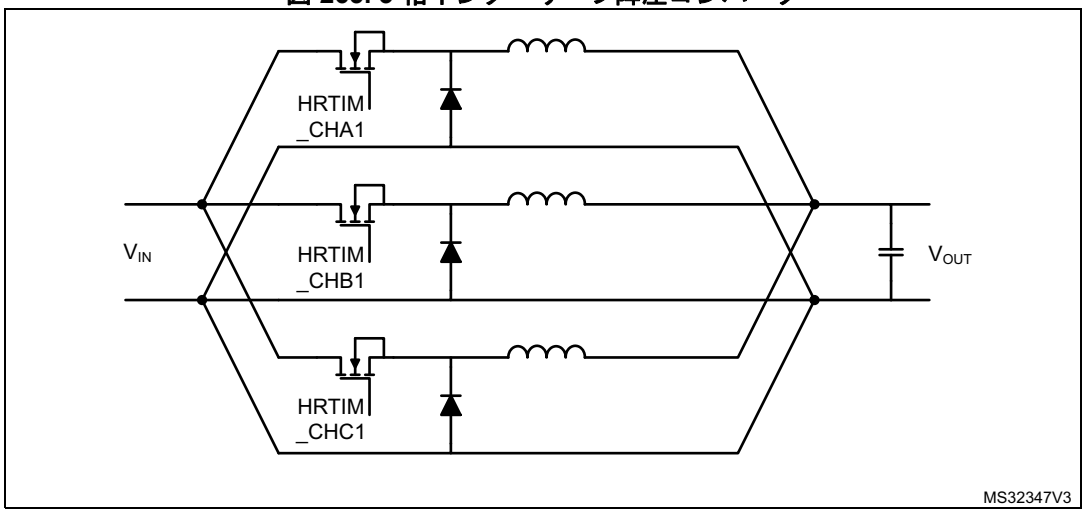
- 入力および出力コンデンサの電流リップルの削減
- EMI の低減
- 位相の数を動的に変えること（位相シェディング）による軽負荷で高効率

HRTIM は複数のコンバータを管理します。制御できるコンバータの数は、次のように、接続形態と使用するリソース（ADC トリガを含む）に依存します。

- マスタタイマと 5 つのタイマを使用する 5 つの同期整流（SR）式の降圧コンバータ
- マスタタイマと 2 つのタイマを使用する 4 つの降圧コンバータ（SR なし）

図 265 は、3 相インターリーブ降圧コンバータの接続形態を示します。

図 265. 3 相インターリーブ降圧コンバータ



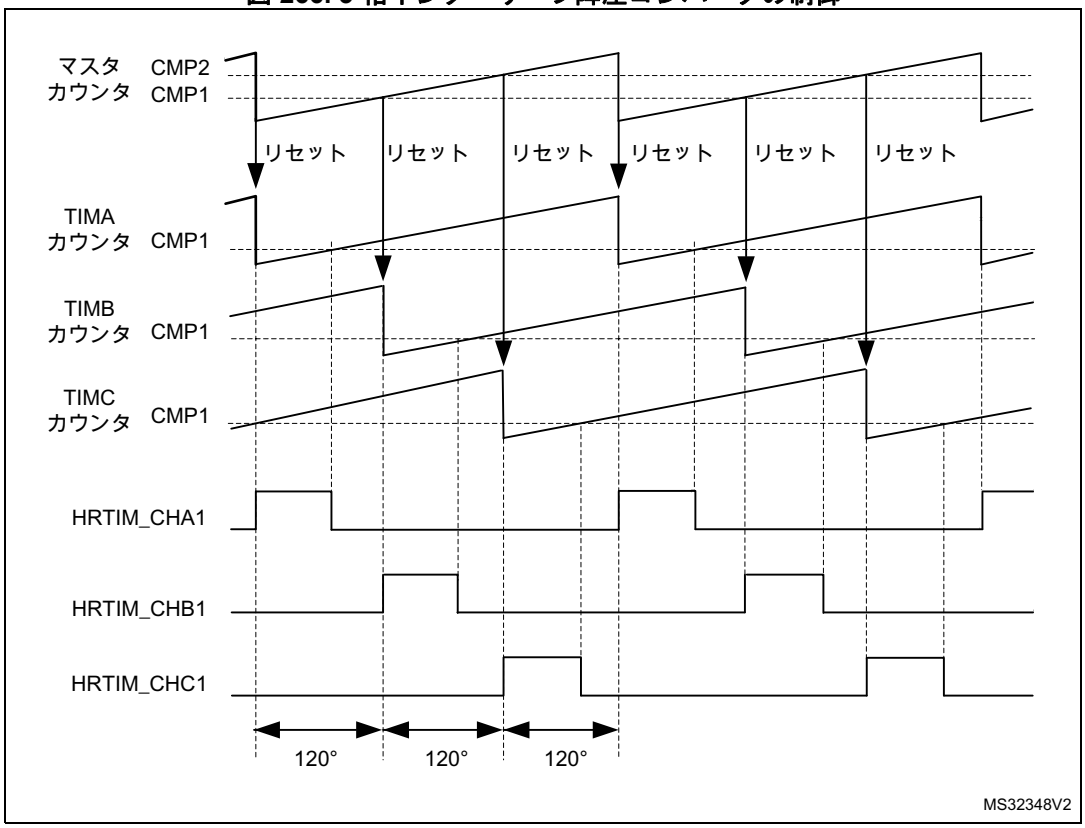
マスタタイマは位相管理を受け持ちます。タイマを周期的にリセットすることによってコンバータ間の位相関係を定義します。位相シフトは 360°を位相の数で割った値で、この例では 120°です。

デューティサイクルは各タイマにプログラムされます。出力は、次のように定義されています。

- HRTIM_CHA1 はマスタタイマの周期でセット、TACMP1 でリセット
- HRTIM_CHB1 はマスタタイマの MCMP1 でセット、TBCMP1 でリセット
- HRTIM_CHC1 はマスタタイマの MCMP2 でセット、TCCMP1 でリセット

ADC トリガは TxCMP2 比較イベントによって生成できます。コンバータの接続形態によりすべての ADC トリガソースは位相がずれているため、それらすべてを 1 つの ADC トリガに結合して ADC リソースを節約できます（たとえば、完全マルチフェーズコンバータに対して 1 つの ADC 通常チャネル）。

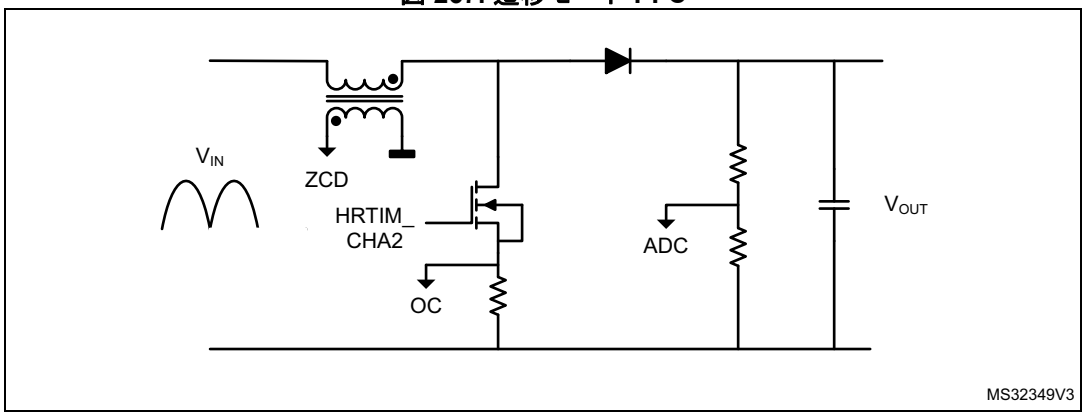
図 266. 3 相インターリーブ降圧コンバータの制御



27.4.4 遷移モード力率補正

基本的動作原理は、固定の Ton 時間中に電流をインダクタに蓄積することです。その後、この電流は Toff 時間中に減衰し、null になった時に周期が再開されます。これは、図 267 に示すように、ゼロ交差検出回路（ZCD）を使用して検出されます。一定の Ton 時間により、インダクタ内のピーク電流値は整流された AC 入力電圧に比例し、それによって力率が補正されます。

図 267. 遷移モード PFC



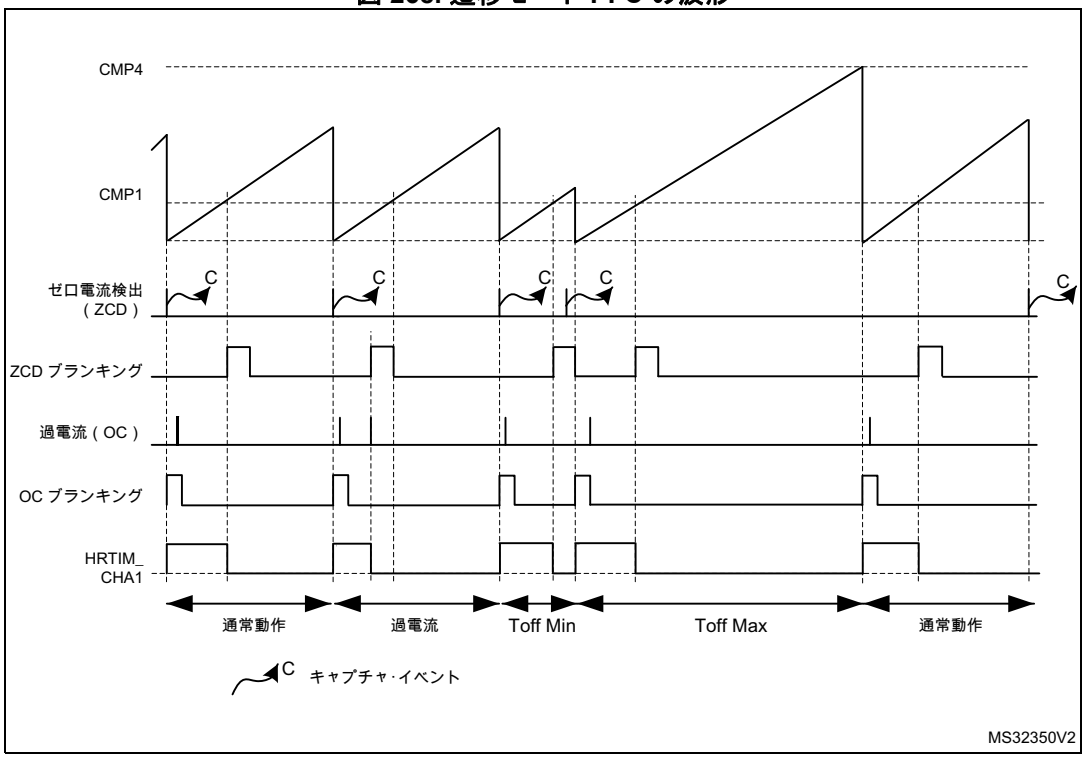
このコンバータは、一定の Ton 時間と、Toff 時間の変動（入力電圧に依存）による可変周波数によって動作します。また、ゼロ交差が検出されないとき動作し、または過電流（OC）の場合に Ton 時間を制限する機能も含まれている必要があります。OC フィードバックは通常、内蔵コンパレータで調整され、外部イベント入力に配送されます。

図 268 に、下記に定義されたパラメータによる、さまざまな動作モード中の波形を示します。

- Ton Min：誤った過電流（フリーホイーリングダイオードリカバリ電流）をマスクします。OC ブランキングとして表されます。
- Ton Max：実用上、コンバータの設定値。CMP1 によって定義されます。
- Toff Min：電流制限がゼロに近いとき（消磁は非常に速い）、周波数を制限します。CMP2 によって定義されます。
- Toff Max：ZCD が働かない場合に、システムが動かなくなることを防ぎます。自動遅延モードで CMP4 によって定義されます。

両方の Toff 値は、出力立ち下がりエッジに関連しなければならないため、自動遅延されます。

図 268. 遷移モード PFC の波形



27.5 HRTIM レジスタ

27.5.1 HRTIM マスタ･タイマ制御レジスタ（HRTIM_MCR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	38	27	26	25	24	23	22	21	20	19	18	17	16
BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	TFCEN	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN
rw	rw	rw		rw	rw	rw			rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		INTLVD[1:0]		HALF	RETRIG	CONT	CKPSC[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 **BRSTDMA[1:0]**：バースト DMA 更新

これらのビットは、バースト DMA トランザクションに関連して更新がどのように起きるかを指定します。

00：DMA バースト転送完了とは無関係に更新されます。

01：DMA バースト転送が完了したときに更新されます。

10：DMA バースト転送完了後にマスタ･タイマがロールオーバーするときに更新されます。このモードは、連続モードでのみ働きます。

11： 予約済みです。

ビット 29 **MREPU**：マスタ･タイマ繰り返し更新

このビットは、マスタタイマ繰り返し周期が（ロールオーバーまたはリセットイベントにより）完了した時、更新が起きるかどうかを決定します。MREPU は、BRSTDMA[1:0] = 00 または 01 の場合のみセットできます。

0：繰り返し時の更新は無効です。

1：繰り返し時の更新は有効です。

ビット 28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **PREEN**：プリロード･イネーブル

このビットは、レジスタプリロードメカニズムを有効にし、メモリに配置されたレジスタへの書込みアクセスが HRTIM のアクティブレジスタに行われるか、プリロードレジスタに行われるかを定義します。

0：プリロード無効：書込みアクセスは直接、アクティブ･レジスタに行われます。

1：プリロード有効：書込みアクセスはプリロード･レジスタに行われます。

ビット 26:25 **DACSYNC[1:0]** DAC 同期

DAC 同期イベントを有効化でき、マスタタイマ更新が起きた時に生成されます。これらのビットは、どの出力に DAC 同期が送られるかを定義します（接続の詳細はセクション 27.3.21：DAC トリガを参照）。

00：DAC トリガは生成されません。

01： トリガは hrtim_dac_trg1 に生成されます。

10： トリガは hrtim_dac_trg2 に生成されます。

11： トリガは hrtim_dac_trg3 に生成されます。

ビット 24:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **TFCEN**：タイマ F カウンタ･イネーブル

このビットはタイマ F カウンタを開始します。

0：タイマ F カウンタ無効化

1：タイマ F カウンタ有効化

注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 21 **TECEN**：タイマ E カウンタ･イネーブル
このビットはタイマ E カウンタを開始します。
0：タイマ E カウンタ無効化
1：タイマ E カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 20 **TDCEN**：タイマ D カウンタ･イネーブル
このビットはタイマ D カウンタを開始します。
0：タイマ D カウンタ無効化
1：タイマ D カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 19 **TCCEN**：タイマ C カウンタ･イネーブル
このビットはタイマ C カウンタを開始します。
0：タイマ C カウンタ無効化
1：タイマ C カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 18 **TBCEN**：タイマ B カウンタ･イネーブル
このビットはタイマ B カウンタを開始します。
0：タイマ B カウンタ無効化
1：タイマ B カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 17 **TACEN**：タイマ A カウンタ･イネーブル
このビットはタイマ A カウンタを開始します。
0：タイマ A カウンタ無効化
1：タイマ A カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 16 **MCEN**：マスタ･タイマ･カウンタ･イネーブル
このビットはマスタタイマのカウンタを開始します。
0：マスタ･カウンタ無効化
1：マスタ･カウンタ有効化
注： このビットは、少なくとも f_{HRTIM} クロックの 8 周期以内に変更してはなりません。

ビット 15:14 **SYNCSRC[1:0]**：同期ソース
これらのビットは、同期出力 SYNCOUT[2:1] に送るソースおよびイベントを定義します。
00：マスタ･タイマの開始
01：マスタ･タイマの比較 1 イベント
10：タイマ A の開始／リセット
11：タイマ A の比較 1 イベント

ビット 13:12 **SYNCOUT[1:0]**：同期出力
これらのビットによって、同期出力イベントのルーチンと調整を定義します。
00：無効
01：予約済み。
10：HRTIM_SCOUT 出力へ正のパルス（16x f_{HRTIM} クロック･サイクル）
11：HRTIM_SCOUT 出力へ負のパルス（16x f_{HRTIM} クロック･サイクル）
注： このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 11 **SYNCSTRTM**：同期マスタ開始
このビットは、同期入力イベントを受信した時、マスタタイマ開始を有効にします。
0：マスタ･タイマへの影響はありません。
1：同期入力イベントはマスタ･タイマを開始します。

ビット 10 **SYNCRSTM**：同期マスタリセット

このビットは、同期入力イベントを受信した時、マスタタイマリセットを有効にします。

0：マスタ･タイマへの影響はありません。

1：同期入力イベントはマスタ･タイマをリセットします。

ビット 9:8 **SYNCIN[1:0]** 同期入力

これらのビットは、同期入力ソースを定義します。

00：無効。HRTIM は同期されず、スタンドアローンモードで動作します。

01：予約済み。

10：hrtim_in_sync[2] への内部イベント：HRTIM はオンチップ･タイマに同期します（セクション ：同期入力 を参照）。

11：hrtim_in_sync[3] への外部イベント：HRTIM_SCIN 入力の正のパルスが HRTIM をトリガします。

注： このパラメータは、影響を受けたタイマが有効化された後は、変更できません。

ビット 7:6 **INTLVD[1:0]**：インタリーブ･モード

このビットフィールドは HALF ビットがリセットされている場合のみ有効です。これは、インタリーブモードを有効化します。

00：インタリーブ･モードは無効です。

01：トリプル･インタリーブ･モード：HRTIM_MPER レジスタが書き込まれているとき、HRTIM_MCMP1R アクティブ･レジスタは HRTIM_MPER/3 の値で自動的に更新され、HRTIM_MCMP2R アクティブ･レジスタは 2x（HRTIM_MPER/3）の値で自動的に更新されます。

10：クワッド･インタリーブ･モード：HRTIM_MPER レジスタが書き込まれているとき、HRTIM_MCMP1R アクティブ･レジスタは HRTIM_MPER/4 の値で自動的に更新され、HRTIM_MCMP2R アクティブ･レジスタは HRTIM_MPER/2 の値で自動的に更新され、HRTIM_MCMP3R アクティブ･レジスタは 3x（HRTIM_MPER/4）の値で自動的に更新されます。

11：インタリーブ･モードは無効です。

ビット 5 **HALF**：ハーフ･モード

このビットは、ハーフデューティサイクルモードを有効にします：HRTIM_MCMP1R アクティブレジスタは、HRTIM_MPER レジスタが書き込まれているとき、HRTIM_MPER/2 の値で自動的に更新されます。

0：ハーフ･モードは無効です。

1：ハーフ･モードは有効です。

ビット 4 **RETRIG**：再トリガ可能モード

このビットは、シングルショットモードでのマスタタイマのカウンタの動作を定義します。

0：タイマは再トリガ可能ではありません：カウンタ･リセットは、カウンタが停止（周期が経過）した場合のみ可能です。

1：タイマは再トリガ可能です：カウンタ･リセットは、カウンタの状態（動作中または停止中）にかかわらず可能です。

ビット 3 **CONT**：連続モード

0：タイマはシングルショット･モードで動作し、MPER 値に達した時、停止します。

1：タイマは連続（フリーランニング）モードで動作し、MPER 値に達した時、ゼロに戻ります。

ビット 2:0 **CKPSC[2:0]**：クロック･プリスケーラ

これらのビットは、マスタタイマの高分解能クロックプリスケーラ比を定義します。

カウンタクロックの等価周波数（$f_{COUNTER}$）は $f_{HRCK} / 2^{CKPSC[2:0]}$ に等しくなります。

プリスケーラ比は、タイマが有効化された後は、変更できません。

27.5.2 HRTIM マスタタイマ割込みステータスレジスタ（HRTIM_MISR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPD**：マスタ更新割込みフラグ
このビットは、マスタタイマレジスタが更新された時にハードウェアによってセットされます。
0：マスタ更新割込みは起きていません。
1：マスタ更新割込みが起きました。

ビット 5 **SYNC**：同期入力割込みフラグ
このビットは、同期入力イベントが受信された時に、ハードウェアによってセットされます。
0：同期入力割込みは起きていません。
1：同期入力割込みが起きました。

ビット 4 **MREP**：マスタ繰り返し割込みフラグ
このビットは、マスタタイマ繰り返し周期が経過した時に、ハードウェアによってセットされます。
0：マスタ繰り返し割込みは起きていません。
1：マスタ繰り返し割込みが起きました。

ビット 3 **MCMP4**：マスタ比較 4 割込みフラグ
MCMP1 の説明を参照してください。

ビット 2 **MCMP3**：マスタ比較 3 割込みフラグ
MCMP1 の説明を参照してください。

ビット 1 **MCMP2**：マスタ比較 2 割込みフラグ
MCMP1 の説明を参照してください。

ビット 0 **MCMP1**：マスタ比較 1 割込みフラグ
このビットは、マスタタイマカウンタが、マスタ比較 1 レジスタでプログラムされた値と一致した時に、ハードウェアによってセットされます。
0：マスタ比較 1 割込みは起きていません。
1：マスタ比較 1 割込みが起きました。

27.5.3 HRTIM マスタタイマ割込みクリアレジスタ（HRTIM_MICR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
									w	w	w	w	w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPDC**：マスタ更新割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MUPD フラグがクリアされます。

ビット 5 **SYNCC**：同期入力割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの SYNC フラグがクリアされます。

ビット 4 **MREPC**：繰り返し割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MREP フラグがクリアされます。

ビット 3 **MCMP4C**：マスタ比較 4 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP4 フラグがクリアされます。

ビット 2 **MCMP3C**：マスタ比較 3 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP3 フラグがクリアされます。

ビット 1 **MCMP2C**：マスタ比較 2 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP2 フラグがクリアされます。

ビット 0 **MCMP1C**：マスタ比較 1 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_MISR レジスタの MCMP1 フラグがクリアされます。

27.5.4 HRTIM マスタ･タイマ DMA／割込みイネーブル･レジスタ（HRTIM_MDIER）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **MUPDDE**：マスタ更新 DMA リクエスト･イネーブル

このビットは、マスタ更新 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：マスタ更新 DMA リクエストは無効です。

1：マスタ更新 DMA リクエストは有効です。

ビット 21 **SYNCDE**：同期入力 DMA リクエスト･イネーブル

このビットは、同期入力 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：同期入力 DMA リクエストは無効です。

1：同期入力 DMA リクエストは有効です。

ビット 20 **MREPDE**：マスタ繰り返し DMA リクエスト･イネーブル

このビットは、マスタタイマ繰り返し DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：繰り返し DMA リクエストは無効です。

1：繰り返し DMA リクエストは有効です。

ビット 19 **MCMP4DE**：マスタ比較 4 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 18 **MCMP3DE**：マスタ比較 3 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 17 **MCMP2DE**：マスタ比較 2 DMA リクエストイネーブル

MCMP1DE の説明を参照してください。

ビット 16 **MCMP1DE**：マスタ比較 1 DMA リクエストイネーブル

このビットは、マスタタイマ比較 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：比較 1 DMA リクエストは無効です。

1：比較 1 DMA リクエストは有効です。

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 6 **MUPDIE**：マスタ更新割込みイネーブル

このビットは、マスタタイマレジスタ更新割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：マスタ更新割込みは無効です。

1：マスタ更新割込みは有効です。

ビット 5 **SYNCIE**：同期入力割込みイネーブル
このビットは、同期入力割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：同期入力割込みは無効です。
1：同期入力割込みは有効です。

ビット 4 **MREPIE**：マスタ繰り返し割込みイネーブル
このビットは、マスタタイマ繰り返し割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：マスタ繰り返し割込みは無効です。
1：マスタ繰り返し割込みは有効です。

ビット 3 **MCMP4IE**：マスタ比較 4 割込みイネーブル
MCMP1IE の説明を参照してください。

ビット 2 **MCMP3IE**：マスタ比較 3 割込みイネーブル
MCMP1IE の説明を参照してください。

ビット 1 **MCMP2IE**：マスタ比較 2 割込みイネーブル
MCMP1IE の説明を参照してください。

ビット 0 **MCMP1IE**：マスタ比較 1 割込みイネーブル
このビットは、マスタタイマ比較 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：比較 1 割込みは無効です。
1：比較 1 割込みは有効です。

27.5.5 HRTIM マスタタイマカウンタレジスタ（HRTIM_MCNTR）

アドレスオフセット：0x010

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCNT[15:0]**：カウンタ値

マスタタイマカウンタ値を保持します。このレジスタは、マスタタイマが停止している（HRTIM_MCR で MCEN = 0）ときのみ書き込むことができます。

注： 32 を下回る（CKPSC[2:0] < 5）HR クロック･プリスケーラ比に対しては、カウンタの最下位ビットは意味がありません。これらは書込みできず、また、読み出すと 0 を返します。

注： カウンタ値が HRTIM_MPER レジスタ値を超えて設定された場合は、タイマの動作は保証されません。

27.5.6 HRTIM マスタタイマ周期レジスタ（HRTIM_MPER）

アドレスオフセット：0x014

リセット値：0x0000 FFDF

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MPER[15:0]**：マスタ･タイマ周期値

このレジスタは、カウンタオーバーフロー値を定義します。

周期値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

最大値は 0x0000 FFDF です。

27.5.7 HRTIM マスタタイマ繰り返しレジスタ（HRTIM_MREP）

アドレスオフセット：0x018

リセット値：0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **MREP[7:0]**：マスタ･タイマ繰り返し周期値

このレジスタはマスタカウンタの繰り返し周期値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

27.5.8 HRTIM マスタタイマ比較 1 レジスタ（HRTIM_MCMP1R）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP1[15:0]**：マスタタイマの比較 1 の値

このレジスタはマスタタイマ比較 1 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

27.5.9 HRTIM マスタタイマ比較 2 レジスタ（HRTIM_MCMP2R）

アドレス･オフセット：0x024

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP2[15:0]**：マスタタイマの比較 2 の値

このレジスタはマスタタイマ比較 2 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

27.5.10 HRTIM マスタタイマ比較 3 レジスタ（HRTIM_MCMP3R）

アドレス･オフセット：0x028

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP3[15:0]**：マスタタイマの比較 3 の値

このレジスタはマスタタイマ比較 3 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

27.5.11 HRTIM マスタタイマ比較 4 レジスタ（HRTIM_MCMP4R）

アドレス･オフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MCMP4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **MCMP4[15:0]**：マスタタイマの比較 4 の値

このレジスタはマスタタイマ比較 4 の値を保持します。これはプリロードレジスタか、またはプリロードが無効の場合はアクティブレジスタです。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

27.5.12 HRTIM タイマ x 制御レジスタ（HRTIM_TIMxCR）（x = A～F）

アドレスオフセット：ブロック A：0x080

アドレス･オフセット：ブロック B：0x100

アドレス･オフセット：ブロック C：0x180

アドレス･オフセット：ブロック D：0x200

アドレス･オフセット：ブロック E：0x280

アドレス･オフセット：ブロック F：0x300

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDGAT[3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	TAU	TxRSTU	TxREPU	TFU
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DELCMP4 [1:0]		DELCMP2 [1:0]		SYNCSTRTx	SYNCRSTx	RSYNCU	INTLVD[1:0]		PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **UPDGAT[3:0]**：更新に対するゲート操作

これらのビットは、バースト DMA トランザクションおよび、更新イネーブル入力 hrtim_upd_en[3:1]（表 210 参照）の外部更新リクエストに関連して更新がどのように起きるかを指定します。

以下に述べる更新イベントになり得るのは：MSTU、TFU、TEU、TDU、TCU、TBU、TAU、TxRSTU、TxREPU です。

0000：DMA バースト転送とは無関係に更新されます。

0001：DMA バースト転送が完了したときに更新されます。

0010：DMA バースト転送が完了した後の更新イベント時に更新されます。

0011：hrtim_upd_en1 の立ち上がりエッジで更新されます。

0100：hrtim_upd_en2 の立ち上がりエッジで更新されます。

0101：hrtim_upd_en3 の立ち上がりエッジで更新されます。

0110：hrtim_upd_en1 の立ち上がりエッジの後の更新イベントで更新されます。

0111：hrtim_upd_en2 の立ち上がりエッジの後の更新イベントで更新されます。

1000：hrtim_upd_en3 の立ち上がりエッジの後の更新イベントで更新されます。

その他：予約済みです。

注：　このビットフィールドは、新しい値をプログラムする前にリセットする必要があります。

0001、0011、0100、0101 に等しい UPDGAT[3:0] の値に対しては、複数の同時更新ソースを受けることが可能です（たとえば、RSTU と DMA バースト）。

ビット 27 **PREEN**：プリロード･イネーブル

このビットは、レジスタプリロードメカニズムを有効にし、プリロード可能なレジスタへの書込みアクセスがアクティブレジスタに行われるか、プリロードレジスタに行われるかを決定します。

0：プリロード無効：書込みアクセスは直接、アクティブ･レジスタに行われます。

1：プリロード有効：書込みアクセスはプリロード･レジスタに行われます。

ビット 26:25 **DACSYNC[1:0]** DAC 同期

タイマ更新が起きた時、DAC 同期イベントが生成されます。これらのビットは、どの出力に DAC 同期が送られるかを定義します（接続の詳細はセクション 27.3.21：DAC トリガを参照）。

00：DAC トリガは生成されません。

01：トリガは hrtim_dac_trg1 に生成されます。

10：トリガは hrtim_dac_trg2 に生成されます。

11：トリガは hrtim_dac_trg3 に生成されます。

ビット 24 **MSTU**：マスタ･タイマ更新

レジスタ更新はマスタタイマの更新によってトリガされます。

0：マスタ･タイマによる更新は無効です。

1：マスタ･タイマによる更新は有効です。

ビット 23 **TEU**：タイマ E 更新

レジスタ更新はタイマ E の更新によってトリガされます。

0：タイマ E による更新は無効です。

1：タイマ E による更新は有効です。

注： このビットは HRTIM_TIMECR では予約済みです。これは HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMFCR でのみ使用できます。

ビット 22 **TDU**：タイマ D 更新

レジスタ更新はタイマ D の更新によってトリガされます。

0：タイマ D による更新は無効です。

1：タイマ D による更新は有効です。

注： このビットは HRTIM_TIMDCR では予約済みです。これは HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMECR、HRTIM_TIMFCR でのみ使用できます。

ビット 21 **TCU**：タイマ C 更新

レジスタ更新はタイマ C の更新によってトリガされます。

0：タイマ C による更新は無効です。

1：タイマ C による更新は有効です。

注： このビットは HRTIM_TIMCCR では予約済みです。これは HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMDCR、HRTIM_TIMECR、HRTIM_TIMFCR でのみ使用できます。

ビット 20 **TBU**：タイマ B 更新

レジスタ更新はタイマ B の更新によってトリガされます。

0：タイマ B による更新は無効です。

1：タイマ B による更新は有効です。

注： このビットは HRTIM_TIMBCR では予約済みです。これは HRTIM_TIMACR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMECR、HRTIM_TIMFCR でのみ使用できます。

ビット 19 **TAU**：タイマ A 更新

レジスタ更新はタイマ A の更新によってトリガされます。

0：タイマ A による更新は無効です。

1：タイマ A による更新は有効です。

注： このビットは HRTIM_TIMBCR では予約済みです。これは HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMECR、HRTIM_TIMFCR でのみ使用できます。

ビット 18 **TxRSTU**：タイマ x リセット更新

レジスタ更新は、タイマ x カウンタのリセットによって、または連続モードで周期値に達した後の 0 へのロールオーバーによってトリガされます。

0：タイマ x のリセット／ロールオーバーによる更新は無効です。

1：タイマ x のリセット／ロールオーバーによる更新は有効です。

ビット 17 **TxREPU**：タイマ x 繰り返し更新

レジスタ更新はカウンタがロールオーバーし、HRTIM_REPx = 0 になった時、トリガされます。

0：繰り返し時の更新は無効です。

1：繰り返し時の更新は有効です。

ビット 16 **TFU**：タイマ F 更新

レジスタ更新はタイマ F の更新によってトリガされます。

0：タイマ F による更新は無効です。

1：タイマ F による更新は有効です。

注： このビットは HRTIM_TIMFCR では予約済みです。これは HRTIM_TIMACR、HRTIM_TIMBCR、HRTIM_TIMCCR、HRTIM_TIMDCR、HRTIM_TIMECR でのみ使用できます。

ビット 15:14 **DELCMP4[1:0]**：CMP4 自動遅延モード

このビットフィールドは、比較レジスタが標準モード（カウンタが比較値に等しくなるとすぐに比較一致が発行される）で動作するか、自動遅延モード（セクション ：自動遅延モードを参照）で動作するかを定義します。

00：CMP4 レジスタは常にアクティブです（標準比較モード）。

01：CMP4 の値はキャプチャ 2 イベント後に再計算されてアクティブになります。

10：CMP4 の値は、キャプチャ 2 イベントの後に再計算されてアクティブになるか、または比較 1 の一致後に再計算されてアクティブになります（キャプチャ 2 イベントが見つからない場合のタイムアウト機能）。

11：CMP4 の値は、キャプチャ 2 イベントの後に再計算されてアクティブになるか、または比較 3 の一致後に再計算されてアクティブになります（キャプチャ･イベントが見つからない場合のタイムアウト機能）。

注： このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 13:12 **DELCMP2[1:0]**：CMP2 自動遅延モード

このビットフィールドは、比較レジスタが標準モード（カウンタが比較値に等しくなるとすぐに比較一致が発行される）で動作するか、自動遅延モード（セクション ：自動遅延モードを参照）で動作するかを定義します。

00：CMP2 レジスタは常にアクティブです（標準比較モード）。

01：CMP2 の値はキャプチャ 1 イベント後に再計算されてアクティブになります。

10：CMP2 の値はキャプチャ 1 イベント後に再計算されてアクティブになるか、または比較 1 の一致後に再計算されてアクティブになります（キャプチャ･イベントが見つからない場合のタイムアウト機能）。

11：CMP2 の値はキャプチャ 1 イベント後に再計算されてアクティブになるか、または比較 3 の一致後に再計算されてアクティブになります（キャプチャ･イベントが見つからない場合のタイムアウト機能）。

注： このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 11 **SYNCSTRTx**：同期タイマ x 開始

このビットは、同期イベント後のタイマ x の動作を定義します。

0：タイマ x に影響しません。

1：同期入力イベントはタイマ x を開始します。

ビット 10 **SYNCRSTx**：同期タイマ x リセット

このビットは、同期イベント後のタイマ x の動作を定義します。

0：タイマ x に影響しません。

1：同期入力イベントはタイマ x をリセットします。

ビット 9 **RSYNCU**：再同期更新

このビットは、タイミングユニットの外部から来る更新ソースを同期する必要があるかどうかを指定します。

0： 隣接するタイマに起因する更新（MSTU、TAU、TBU、TCU、TDU、TEU、TFU ビットがセットされている場合）またはソフトウェア更新（TxSWU ビット）に起因する更新は直ちに考慮されます。

1： 隣接するタイマに起因する更新（MSTU、TAU、TBU、TCU、TDU、TEU、TFU ビットがセットされている場合）またはソフトウェア更新（TxSWU ビット）に起因する更新は、次のリセット／ロールオーバー･イベント時に考慮されます。

注： このビットは、UPDGAT[3:0] = 0000 の場合にのみ、意味があります。そうでない場合は、無視されます。

ビット 8:7 **INTLVD[1:0]**：インタリーブ･モード

このビットフィールドは HALF ビットがリセットされている場合のみ有効です。これは、インタリーブモードを有効化します。

00：インタリーブ･モードは無効です。

01：トリプル･インタリーブ･モード：HRTIM_PERxR レジスタが書き込まれているとき、HRTIM_CMP1xR アクティブ･レジスタは HRTIM_PERxR/3 の値で自動的に更新され、HRTIM_CMP2xR アクティブ･レジスタは 2x（HRTIM_PERxR/3）の値で自動的に更新されます。

10：クワッド･インタリーブ･モード：HRTIM_PERxR レジスタが書き込まれているとき、HRTIM_CMP1xR アクティブ･レジスタは HRTIM_PERxR/4 の値で自動的に更新され、HRTIM_CMP2xR アクティブ･レジスタは HRTIM_PERxR/2 の値で自動的に更新され、HRTIM_CMP3xR アクティブ･レジスタは 3x（HRTIM_PERxR/4）の値で自動的に更新されます。

11：インタリーブ･モードは無効です。

ビット 6 **PSHPLL**：プッシュプル･モード･イネーブル

このビットはプッシュプルモードを有効にします。

0：プッシュプル･モードは無効です。

1：プッシュプル･モードは有効です。

注： このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 5 **HALF**：ハーフ･モード･イネーブル

このビットは、ハーフデューティサイクルモードを有効にします：HRTIM_CMP1xR アクティブレジスタは、HRTIM_PERxR レジスタが書き込まれているとき、HRTIM_PERxR/2 の値で自動的に更新されます。

0：ハーフ･モードは無効です。

1：ハーフ･モードは有効です。

ビット 4 **RETRIG**：再トリガ可能モード

このビットは、シングルショットモードでのカウンタの動作を定義します。

0：タイマは再トリガ可能ではありません：カウンタ･リセットは、カウンタが停止（シングルショット･モードで周期が経過、または連続モードでカウンタが停止）した場合に行われます。

1：タイマは再トリガ可能です：カウンタ･リセットは、カウンタの状態にかかわらず行われます。

ビット 3 **CONT**：連続モード

このビットはタイマの動作モードを定義します。

0：タイマはシングルショット･モードで動作し、TIMxPER 値に達した時、停止します。

1：タイマは連続モードで動作し、TIMxPER 値に達した時、ゼロに戻ります。

ビット 2:0 **CKPSCx[2:0]**：HRTIM タイマ x クロックプリスケーラ

これらのビットは、マスタタイマの高分解能クロックプリスケーラ比を定義します。

カウンタクロックの等価周波数（$f_{COUNTER}$）は $f_{HRCK} / 2^{CKPSC[2:0]}$ に等しくなりす。

プリスケーラ比は、タイマが有効化された後は、変更できません。

27.5.13 HRTIM タイマ x 割込みステータス･レジスタ（HRTIM_TIMxISR）（x = A ~ F）

アドレスオフセット：ブロック A：0x084

アドレス･オフセット：ブロック B：0x104

アドレス･オフセット：ブロック C：0x184

アドレス･オフセット：ブロック D：0x204

アドレス･オフセット：ブロック E：0x284

アドレス･オフセット：ブロック F：0x304

リセット値：0x000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2 STAT	O1 STAT	IPP STAT	CPP STAT
										r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	r	r	r	r	r	r	r	r	r		r	r	r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **O2CPY**：出力 2 コピー

このステータスビットは、出力ステージ（チョッパ、極性）前の、出力 2 状態の生のコピーです。これによって、遅延保護後に出力を再有効化する前に現在の出力状態を確認できます。

0：出力 2 は非アクティブです。

1：出力 2 はアクティブです。

ビット 20 **O1CPY**：出力 1 コピー

このステータスビットは、出力ステージ（チョッパ、極性）前の、出力 1 状態の生のコピーです。これによって、遅延保護後に出力を再有効化する前に現在の出力状態を確認できます。

0：出力 1 は非アクティブです。

1：出力 1 はアクティブです。

ビット 19 **O2STAT**：出力 2 ステータス

このステータスビットは、遅延アイドル保護がトリガされた時、出力 2 の状態を示します。このビットは、新たに遅延保護に移行した時に更新されます。このビットはバランスアイドルでは更新されません。

0：出力 2 は非アクティブでした。

1：出力 2 はアクティブでした。

ビット 18 **O1STAT**：出力 1 ステータス

このステータスビットは、遅延アイドル保護がトリガされた時、出力 1 の状態を示します。このビットは、新たに遅延保護に移行した時に更新されます。このビットはバランスアイドルでは更新されません。

0：出力 1 は非アクティブでした。

1：出力 1 はアクティブでした。

ビット 17 **IPPSTAT**：アイドルプッシュプルステータス

このステータスビットは、プッシュプルモード、バランスフォールトモード、または遅延アイドルモードで、保護がトリガされた時（出力状態がアクティブであれ、非アクティブであれ）、どちらの出力に信号が適用されていたかを示します。

0：出力 1 がアクティブで、出力 2 が非アクティブに強制されていた時に、保護が発生しました。

1：出力 2 がアクティブで、出力 1 が非アクティブに強制されていた時に、保護が発生しました。

ビット 16 **CPPSTAT**：現在のプッシュプルステータス

このステータスビットは、プッシュプルモードで信号が現在、どちらの出力に適用されているかを示します。これは、この設定でのみ意味を持ちます。

0：信号は出力 1 に適用されており、出力 2 は非アクティブに強制されています。

1：信号は出力 2 に適用されており、出力 1 は非アクティブに強制されています。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DLYPRT**：遅延保護フラグ

0：遅延保護割込みは起きていません。

1：遅延アイドルまたはバランス･アイドル･モードへの移行が起きました。

ビット 13 **RST**：リセットまたはロールオーバー割込みフラグ（あるいはその両方）

このビットは、タイマ x カウンタがリセットされた、または連続モードでロールオーバーした時に、ハードウェアによってセットされます。

0：TIMx カウンタのリセット／ロールオーバー割込みは起きていません。

1：TIMX カウンタのリセット／ロールオーバー割込みが起きました。

ビット 12 **RSTx2**：出力 2 リセット割込みフラグ

RSTx1 の説明を参照してください。

ビット 11 **SETx2**：出力 2 セット割込みフラグ

SETx1 の説明を参照してください。

ビット 10 **RSTx1**：出力 1 リセット割込みフラグ

このビットは、Tx1 出力がリセット（アクティブから非アクティブモードに移行）された時に、ハードウェアによってセットされます。

0：Tx1 出力リセット割込みは起きていません。

1：Tx1 出力リセット割込みが起きました。

ビット 9 **SETx1**：出力 1 セット割込みフラグ

このビットは、Tx1 出力がセット（非アクティブからアクティブモードに移行）された時に、ハードウェアによってセットされます。

0：Tx1 出力セット割込みは起きていません。

1：Tx1 出力セット割込みが起きました。

ビット 8 **CPT2**：キャプチャ 2 割込みフラグ

CPT1 の説明を参照してください。

ビット 7 **CPT1**：キャプチャ 1 割込みフラグ

このビットは、タイマ x キャプチャ 1 イベントが発生した時に、ハードウェアによってセットされます。

0：タイマ x キャプチャ 1 割込みは起きていません。

1：タイマ x キャプチャ 1 割込みが起きました。

ビット 6 **UPD**：更新割込みフラグ

このビットは、タイマ x 更新イベントが発生した時に、ハードウェアによってセットされます。

0：タイマ x 更新割込みは起きていません。

1：タイマ x 更新割込みが起きました。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REP**：繰り返し割込みフラグ

このビットは、タイマ x 繰り返し周期が経過した時に、ハードウェアによってセットされます。

0：タイマ x 繰り返し割込みは起きていません。

1：タイマ x 繰り返し割込みが起きました。

ビット 3 **CMP4**：比較 4 割込みフラグ

CMP1 の説明を参照してください。

ビット 2 **CMP3**：比較 3 割込みフラグ
CMP1 の説明を参照してください。

ビット 1 **CMP2**：比較 2 割込みフラグ
CMP1 の説明を参照してください。

ビット 0 **CMP1**：比較 1 割込みフラグ
このビットは、タイマ x カウンタが、比較 1 レジスタでプログラムされた値に一致した時に、ハードウェアによってセットされます。
0：比較 1 割込みは起きていません。
1：比較 1 割込みが起きました。

27.5.14 HRTIM タイマ x 割込みクリア･レジスタ（HRTIM_TIMxICR）（x = A ~ F）

アドレスオフセット：ブロック A：0x088

アドレス･オフセット：ブロック B：0x108

アドレス･オフセット：ブロック C：0x188

アドレス･オフセット：ブロック D：0x208

アドレス･オフセット：ブロック E：0x288

アドレス･オフセット：ブロック F：0x308

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DLYPRTC**：遅延保護フラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの DLYPRT フラグがクリアされます。

ビット 13 **RSTC**：リセット割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RST フラグがクリアされます。

ビット 12 **RSTx2C**：出力 2 リセットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RSTx2 フラグがクリアされます。

ビット 11 **SETx2C**：出力 2 セットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの SETx2 フラグがクリアされます。

ビット 10 **RSTx1C**：出力 1 リセットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの RSTx1 フラグがクリアされます。

ビット 9 **SETx1C**：出力 1 セットフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの SETx1 フラグがクリアされます。

ビット 8 **CPT2C**：キャプチャ 2 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CPT2 フラグがクリアされます。

ビット 7 **CPT1C**：キャプチャ 1 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CPT1 フラグがクリアされます。

ビット 6 **UPDC**：更新割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの UPD フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REPC**：繰り返し割込みフラグ･クリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの REP フラグがクリアされます。

ビット 3 **CMP4C**：比較 4 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP4 フラグがクリアされます。

ビット 2 **CMP3C**：比較 3 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP3 フラグがクリアされます。

ビット 1 **CMP2C**：比較 2 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP2 フラグがクリアされます。

ビット 0 **CMP1C**：比較 1 割込みフラグクリア
このビットに 1 を書き込むと、HRTIM_TIMxISR レジスタの CMP1 フラグがクリアされます。

27.5.15 HRTIM タイマ x DMA 割込みイネーブル･レジスタ（HRTIM_TIMxDIER）（x = A ~ F）

アドレスオフセット：ブロック A：0x08C

アドレス･オフセット：ブロック B：0x10C

アドレス･オフセット：ブロック C：0x18C

アドレス･オフセット：ブロック D：0x20C

アドレス･オフセット：ブロック E：0x28C

アドレス･オフセット：ブロック F：0x30C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SETx1DE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **DLYPRTDE**：遅延保護 DMA リクエストイネーブル
このビットは、遅延保護の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：遅延保護 DMA リクエストは無効です。
1：遅延保護 DMA リクエストは有効です。

ビット 29 **RSTDE**：リセット／ロールオーバー DMA リクエスト･イネーブル

このビットは、タイマ x カウンタのリセットまたは連続モードでのロールオーバー時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：タイマ x カウンタのリセット／ロールオーバー DMA リクエストは無効です。

1：タイマ x カウンタのリセット／ロールオーバー DMA リクエストは有効です。

ビット 28 **RSTx2DE**：出力 2 リセット DMA リクエストイネーブル

RSTx1DE の説明を参照してください。

ビット 27 **SETx2DE**：出力 2 セット DMA リクエストイネーブル

SETx1DE の説明を参照してください。

ビット 26 **RSTx1DE**：出力 1 リセット DMA リクエストイネーブル

このビットは、Tx1 出力リセット DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx1 出力リセット DMA リクエストは無効です。

1：Tx1 出力リセット DMA リクエストは有効です。

ビット 25 **SETx1DE**：出力 1 セット DMA リクエストイネーブル

このビットは、Tx1 出力セット DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx1 出力セット DMA リクエストは無効です。

1：Tx1 出力セット DMA リクエストは有効です。

ビット 24 **CPT2DE**：キャプチャ 2 DMA リクエストイネーブル

CPT1DE の説明を参照してください。

ビット 23 **CPT1DE**：キャプチャ 1 DMA リクエストイネーブル

このビットは、キャプチャ 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：キャプチャ 1 DMA リクエストは無効です。

1：キャプチャ 1 DMA リクエストは有効です。

ビット 22 **UPDDE**：更新 DMA リクエスト･イネーブル

このビットは、更新イベント時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **REPDE**：繰り返し DMA リクエスト･イネーブル

このビットは、繰り返しイベント時の DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：繰り返し DMA リクエストは無効です。

1：繰り返し DMA リクエストは有効です。

ビット 19 **CMP4DE**：比較 4 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 18 **CMP3DE**：比較 3 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 17 **CMP2DE**：比較 2 DMA リクエストイネーブル

CMP1DE の説明を参照してください。

ビット 16 **CMP1DE**：比較 1 DMA リクエストイネーブル

このビットは、比較 1 DMA リクエストを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：比較 1 DMA リクエストは無効です。

1：比較 1 DMA リクエストは有効です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **DLYPRTIE**：遅延保護割込みイネーブル

このビットは、遅延保護の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：遅延保護割込みは無効です。

1：遅延保護割込みは有効です。

ビット 13 **RSTIE**：リセット／ロールオーバー割込みイネーブル

このビットは、タイマ x カウンタのリセットまたは連続モードでのロールオーバー時の割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：タイマ x カウンタのリセット／ロールオーバー割込みは無効です。

1：タイマ x カウンタのリセット／ロールオーバー割込みは有効です。

ビット 12 **RSTx2IE**：出力 2 リセット割込みイネーブル

RSTx1IE の説明を参照してください。

ビット 11 **SETx2IE**：出力 2 セット割込みイネーブル

SETx1IE の説明を参照してください。

ビット 10 **RSTx1IE**：出力 1 リセット割込みイネーブル

このビットは、Tx1 出力リセット割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx1 出力リセット割込みは無効です。

1：Tx1 出力リセット割込みは有効です。

ビット 9 **SETx1IE**：出力 1 セット割込みイネーブル

このビットは、Tx1 出力セット割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：Tx1 出力セット割込みは無効です。

1：Tx1 出力セット割込みは有効です。

ビット 8 **CPT2IE**：キャプチャ割込みイネーブル

CPT1IE の説明を参照してください。

ビット 7 **CPT1IE**：キャプチャ割込みイネーブル

このビットは、キャプチャ 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：キャプチャ 1 割込みは無効です。

1：キャプチャ 1 割込みは有効です。

ビット 6 **UPDIE**：更新割込みイネーブル

このビットは、更新イベント割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新割込みは無効です。

1：更新割込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **REPIE**：繰り返し割込みイネーブル

このビットは、繰り返しイベント割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：繰り返し割込みは無効です。

1：繰り返し割込みは有効です。

ビット 3 **CMP4IE**：比較 4 割込みイネーブル

CMP1IE の説明を参照してください。

ビット 2 **CMP3IE**：比較 3 割込みイネーブル
CMP1IE の説明を参照してください。

ビット 1 **CMP2IE**：比較 2 割込みイネーブル
CMP1IE の説明を参照してください。

ビット 0 **CMP1IE**：比較 1 割込みイネーブル
このビットは、比較 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：比較 1 割込みは無効です。
1：比較 1 割込みは有効です。

27.5.16 HRTIM タイマ x カウンタレジスタ（HRTIM_CNTxR）（x = A～F）

アドレスオフセット：ブロック A：0x090

アドレス･オフセット：ブロック B：0x110

アドレス･オフセット：ブロック C：0x190

アドレス･オフセット：ブロック D：0x210

アドレス･オフセット：ブロック E：0x290

アドレス･オフセット：ブロック F：0x310

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNTx[15:0]**：タイマ x カウンタ値
このレジスタは、タイマ x カウンタ値を保持します。これは、タイマが停止している（HRTIM_TIMxCR で TxCEN = 0）ときのみ書き込むことができます。

注： 32 を下回る（CKPSC[2:0] < 5）HR クロックプリスケーラ比に対しては、カウンタの最下位ビットは意味がありません。これらは書込みできず、また、読み出すと 0 を返します。

注： カウンタ値が HRTIM_PERxR レジスタ値を超えている場合は、タイマの動作は保証されません。

27.5.17 HRTIM タイマ x 周期レジスタ（HRTIM_PERxR）（x = A～F）

アドレスオフセット：ブロック A：0x094

アドレス･オフセット：ブロック B：0x114

アドレス･オフセット：ブロック C：0x194

アドレス･オフセット：ブロック D：0x214

アドレス･オフセット：ブロック E：0x294

アドレス･オフセット：ブロック F：0x314

リセット値：0x0000 FFDF

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **PERx[15:0]**：タイマ x 周期値

このレジスタは、タイマ x 周期値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

周期値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

最大値は 0x0000 FFDF です。

27.5.18 HRTIM タイマ x 繰り返しレジスタ（HRTIM_REPxR）（x = A～F）

アドレスオフセット：ブロック A：0x098

アドレス･オフセット：ブロック B：0x118

アドレス･オフセット：ブロック C：0x198

アドレス･オフセット：ブロック D：0x218

アドレス･オフセット：ブロック E：0x298

アドレス･オフセット：ブロック F：0x318

リセット値：0x0000 0000

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

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REPx[7:0]**：タイマ x 繰り返し周期値

このレジスタは、繰り返し周期値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

27.5.19 HRTIM タイマ x 比較 1 レジスタ（HRTIM_CMP1xR）（x = A～F）

アドレス･オフセット：ブロック A：0x09C

アドレス･オフセット：ブロック B：0x11C

アドレス･オフセット：ブロック C：0x19C

アドレス･オフセット：ブロック D：0x21C

アドレス･オフセット：ブロック E：0x29C

アドレス･オフセット：ブロック F：0x31C

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP1x [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP1x[15:0]**：タイマ x の比較 1 値

このレジスタは比較 1 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は null か、または f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

null 値は、セクション ：ヌルデューティサイクルの例外ケースで説明されている使用例に従ってプログラムされます。

27.5.20 HRTIM タイマ x 比較 1 合成レジスタ（HRTIM_CMP1CxR）（x = A～F）

アドレス･オフセット：ブロック A：0x0A0

アドレス･オフセット：ブロック B：0x120

アドレス･オフセット：ブロック C：0x1A0

アドレス･オフセット：ブロック D：0x220

アドレス･オフセット：ブロック E：0x2A0

アドレス･オフセット：ブロック F：0x320

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP1x [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **REPx[7:0]**：タイマ x 繰り返し値（HRTIM_REPx レジスタからのエイリアス）
このビットフィールドは、HRTIMx_REPxR レジスタの REPx[7:0] ビットフィールドからのエイリアスです。

ビット 15:0 **CMP1x[15:0]**：タイマ x の比較 1 値
このビットフィールドは、HRTIMx_CMP1xR レジスタの CMP1x[15:0] ビットフィールドからのエイリアスです。

27.5.21 HRTIM タイマ x 比較 2 レジスタ（HRTIM_CMP2xR）（x = A～F）

アドレス･オフセット：ブロック A：0x0A4

アドレス･オフセット：ブロック B：0x124

アドレス･オフセット：ブロック C：0x1A4

アドレス･オフセット：ブロック D：0x224

アドレス･オフセット：ブロック E：0x2A4

アドレス･オフセット：ブロック F：0x324

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP2x [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP2x[15:0]**：タイマ x の比較 2 値

このレジスタは比較 2 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

このレジスタは、HRTIM_TIMxCR の DELCMP2[1:0] ビットで有効にされている場合、自動遅延比較レジスタとして動作します。

27.5.22 HRTIM タイマ x 比較 3 レジスタ（HRTIM_CMP3xR）（x = A～F）

アドレス･オフセット：ブロック A：0x0A8

アドレス･オフセット：ブロック B：0x128

アドレス･オフセット：ブロック C：0x1A8

アドレス･オフセット：ブロック D：0x228

アドレス･オフセット：ブロック E：0x2A8

アドレス･オフセット：ブロック F：0x328

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP3x [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP3x[15:0]**：タイマ x の比較 3 値

このレジスタは比較 3 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は null か、または fHRTIM クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

null 値は、セクション ：ヌルデューティサイクルの例外ケースで説明されている使用例に従ってプログラムされます。

27.5.23 HRTIM タイマ x 比較 4 レジスタ（HRTIM_CMP4xR）（x = A～F）

アドレス･オフセット：ブロック A：0x0AC

アドレス･オフセット：ブロック B：0x12C

アドレス･オフセット：ブロック C：0x1AC

アドレス･オフセット：ブロック D：0x22C

アドレス･オフセット：ブロック E：0x2AC

アドレス･オフセット：ブロック F：0x32C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP4x [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP4x[15:0]**：タイマ x の比較 4 値

このレジスタは比較 4 の値を保持します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

比較値は f_{HRTIM} クロックの 3 周期（CKPSC[2:0] = 0 の場合は 0x60、CKPSC[2:0] = 1 の場合は 0x30、CKPSC[2:0] = 2 の場合は 0x18、・・・）以上でなければなりません。

このレジスタは、HRTIM_TIMxCR の DELCMP4[1:0] ビットで有効にされている場合、自動遅延比較レジスタとして動作できます。

27.5.24 HRTIM タイマ x キャプチャ 1 レジスタ（HRTIM_CPT1xR）（x = A～F）

アドレス･オフセット：ブロック A：0x0B0

アドレス･オフセット：ブロック B：0x130

アドレス･オフセット：ブロック C：0x1B0

アドレス･オフセット：ブロック D：0x230

アドレス･オフセット：ブロック E：0x2B0

アドレス･オフセット：ブロック F：0x330

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT1x [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **DIR**：タイマ x キャプチャ 1 方向ステータス
このレジスタは、キャプチャ 1 イベントが発生した時のカウント方向の値を保持します。
0：タイマはアップカウントです。
1：タイマはダウンカウントです。
アップカウントモード（UDM ビットはリセット）では、DIR ビットは常に 0 として読み出されます。

ビット 15:0 **CPT1x[15:0]**：タイマ x のキャプチャ 1 の値
このレジスタは、キャプチャ 1 イベントが発生した時のカウンタ値を保持します。

注： **アップ／ダウンモード（UDM ビットを 1 に設定）では、キャプチャ値は以下から参照されます。**
- アップカウント時のカウントリセット
- ダウンカウント時の PER イベント

DIR ビットを使用すると、キャプチャされた値を読み取るときに、アップ・ダウンフェーズを区別できます。

注： **これは通常分解能のレジスタです。32 を下回る（CKPSC[2:0] < 5）HR クロックプリスケーラ比に対しては、カウンタの最下位ビットは意味がありません。これらは書込みできず、また、読み出すと 0 を返します。**

27.5.25 HRTIM タイマ x キャプチャ 2 レジスタ（HRTIM_CPT2xR）（x = A～F）

アドレス･オフセット：ブロック A：0x0B4

アドレス･オフセット：ブロック B：0x134

アドレス･オフセット：ブロック C：0x1B4

アドレス･オフセット：ブロック D：0x234

アドレス･オフセット：ブロック E：0x2B4

アドレス･オフセット：ブロック F：0x334

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CPT2x [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **DIR**：タイマ x キャプチャ 1 方向ステータス
このレジスタは、キャプチャ 1 イベントが発生した時のカウント方向の値を保持します。
0：タイマはアップカウントです。
1：タイマはダウンカウントです。
アップカウントモード（UDM ビットはリセット）では、DIR ビットは常に 0 として読み出されます。

ビット 15:0 **CPT2x[15:0]**：タイマ x のキャプチャ 2 の値
このレジスタは、キャプチャ 2 イベントが発生した時のカウンタ値を保持します。

注： **アップ／ダウンモード（UDM ビットを 1 に設定）では、キャプチャ値は以下から参照されます。**
- アップカウント時のカウントリセット
- ダウンカウント時の PER イベント

DIR ビットを使用すると、キャプチャされた値を読み取るときに、アップ・ダウンフェーズを区別できます。

注： **これは通常分解能のレジスタです。32 を下回る（CKPSC[2:0] < 5）HR クロックプリスケーラ比に対しては、カウンタの最下位ビットは意味がありません。これらは書込みできず、また、読み出すと 0 を返します。**

27.5.26 HRTIM タイマ x デッドタイム･レジスタ（HRTIM_DTxR）（x = A ~ F）

アドレスオフセット：ブロック A：0x0B8

アドレス･オフセット：ブロック B：0x138

アドレス･オフセット：ブロック C：0x1B8

アドレス･オフセット：ブロック D：0x238

アドレス･オフセット：ブロック E：0x2B8

アドレス･オフセット：ブロック F：0x338

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]								
rwo	rwo					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DTRLKx	DTRSLKx	Res.	DTPRSC[1:0]			SDTRx	DTRx[8:0]								
rwo	rwo		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **DTFLKx**：デッドタイム立ち下がりロック

この 1 回のみ書込み可能のビットは、デッドタイム（正負符号および値）が変更されることを防ぎます（有効な場合）。

0：デッドタイム立ち下がりの値と正負符号は書込み可能です。

1：デッドタイム立ち下がりの値と正負符号は読出し専用です。

注： このビットはプリロードされません。

ビット 30 **DTFSLKx**：デッドタイム立ち下がり正負符号ロック

この 1 回のみ書込み可能のビットは、立ち下がりデッドタイムの正負符号が変更されることを防ぎます（有効な場合）。

0：デッドタイム立ち下がりの正負符号は書込み可能です。

1：デッドタイム立ち下がりの正負符号は読出し専用です。

注： このビットはプリロードされません。

ビット 29:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **SDTFx**：デッドタイム立ち下がり値の正負符号

このレジスタは、デッドタイムが正か（信号はオーバーラップしない）、または負か（信号はオーバーラップする）を決定します。

0：立ち下がりエッジ時の正のデッドタイム

1：立ち下がりエッジ時の負のデッドタイム

ビット 24:16 **DTFx[8:0]**：デッドタイム立ち下がり値

このレジスタは、リファレンス PWM 信号の立ち下がりエッジの後のデッドタイムの値を保持します。

t_{DTF} = DTFx[8:0] x t_{DTG}

ビット 15 **DTRLKx**：デッドタイム立ち上がりロック

この 1 回のみ書込み可能のビットは、デッドタイム（正負符号および値）が変更されることを防ぎます（有効な場合）。

0：デッドタイム立ち上がりの値と正負符号は書込み可能です。

1：デッドタイム立ち上がりの値と正負符号は読出し専用です。

注： このビットはプリロードされません。

ビット 14 **DTRSLKx**：デッドタイム立ち上がり正負符号ロック

この 1 回のみ書込み可能のビットは、デッドタイムの正負符号が変更されることを防ぎます（有効な場合）。

0：デッドタイム立ち上がりの正負符号は書込み可能です。

1：デッドタイム立ち上がりの正負符号は読出し専用です。

注： このビットはプリロードされません。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:10 **DTPRSC[2:0]**：デッドタイム･プリスケーラ

このレジスタは、デッドタイムのクロックプリスケーラの値を保持します。

t_{DTG} = （$2^{(DTPRSC[2:0])}$）x （t_{HRTIM} / 8）

（すなわち、000：735 ps、001 = 1.471 ns、・・・）

このビットフィールドは、ロックビット（DTFLKx、DTFSLKx、DTRLKx、DTRSLKx）のいずれかが有効になるとすぐに読出し専用になります。

ビット 9 **SDTRx**：デッドタイム立ち上がり値の正負符号

このレジスタは、デッドタイムが正か負か（信号はオーバーラップする）を決定します。

0：立ち上がりエッジ時の正のデッドタイム

1：立ち上がりエッジ時の負のデッドタイム

ビット 8:0 **DTRx[8:0]**：デッドタイム立ち上がり値

このレジスタは、リファレンス PWM 信号の立ち上がりエッジの後のデッドタイムの値を保持します。

t_{DTR} = DTRx[8:0] x t_{DTG}

27.5.27 HRTIM タイマ x 出力 1 設定レジスタ（HRTIM_SETx1R）（x = A～F）

アドレス･オフセット：ブロック A：0x0BC

アドレス･オフセット：ブロック B：0x13C

アドレス･オフセット：ブロック C：0x1BC

アドレス･オフセット：ブロック D：0x23C

アドレス･オフセット：ブロック E：0x2BC

アドレス･オフセット：ブロック F：0x33C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UPDATE**：レジスタ更新（プリロードからアクティブへ遷移）
レジスタ更新イベントは出力をアクティブ状態に強制します。

ビット 30 **EXTEVNT10**：外部イベント 10
EXTEVNT1 の説明を参照してください。

ビット 29 **EXTEVNT9**：外部イベント 9
EXTEVNT1 の説明を参照してください。

ビット 28 **EXTEVNT8**：外部イベント 8
EXTEVNT1 の説明を参照してください。

ビット 27 **EXTEVNT7**：外部イベント 7
EXTEVNT1 の説明を参照してください。

ビット 26 **EXTEVNT6**：外部イベント 6
EXTEVNT1 の説明を参照してください。

ビット 25 **EXTEVNT5**：外部イベント 5
EXTEVNT1 の説明を参照してください。

ビット 24 **EXTEVNT4**：外部イベント 4
EXTEVNT1 の説明を参照してください。

ビット 23 **EXTEVNT3**：外部イベント 3
EXTEVNT1 の説明を参照してください。

ビット 22 **EXTEVNT2**：外部イベント 2
EXTEVNT1 の説明を参照してください。

ビット 21 **EXTEVNT1**：外部イベント 1
外部イベント 1 は出力をアクティブ状態に強制します。

ビット 20 **TIMEVNT9**：タイマイベント 9
TIMEVNT1 の説明を参照してください。

ビット 19 **TIMEVNT8**：タイマイベント 8
TIMEVNT1 の説明を参照してください。

ビット 18 **TIMEVNT7**：タイマイベント 7
TIMEVNT1 の説明を参照してください。

ビット 17 **TIMEVNT6**：タイマイベント 6
TIMEVNT1 の説明を参照してください。

ビット 16 **TIMEVNT5**：タイマイベント 5
TIMEVNT1 の説明を参照してください。

ビット 15 **TIMEVNT4**：タイマイベント 4
TIMEVNT1 の説明を参照してください。

ビット 14 **TIMEVNT3**：タイマイベント 3
TIMEVNT1 の説明を参照してください。

ビット 13 **TIMEVNT2**：タイマイベント 2
TIMEVNT1 の説明を参照してください。

ビット 12 **TIMEVNT1**：タイマイベント 1
タイマイベント 1 は出力をアクティブ状態に強制します（タイマイベントの割り当てについては表 217 を参照）。

ビット 11 **MSTCMP4**：マスタ比較 4
マスタタイマ比較 4 イベントは出力をアクティブ状態に強制します。

ビット 10 **MSTCMP3**：マスタ比較 3
マスタタイマ比較 3 イベントは出力をアクティブ状態に強制します。

ビット 9 **MSTCMP2**：マスタ比較 2
マスタタイマ比較 2 イベントは出力をアクティブ状態に強制します。

ビット 8 **MSTCMP1**：マスタ比較 1
マスタタイマ比較 1 イベントは出力をアクティブ状態に強制します。

ビット 7 **MSTPER**：マスタ周期
連続モードでのマスタタイマのカウンタのロールオーバー、またはシングルショットモードでのマスタタイマのリセットは、出力をアクティブ状態に強制します。

ビット 6 **CMP4**：タイマ x の比較 4
タイマ A 比較 4 イベントは出力をアクティブ状態に強制します。

ビット 5 **CMP3**：タイマ x の比較 3
タイマ A 比較 3 イベントは出力をアクティブ状態に強制します。

ビット 4 **CMP2**：タイマ x の比較 2
タイマ A 比較 2 イベントは出力をアクティブ状態に強制します。

ビット 3 **CMP1**：タイマ x の比較 1
タイマ A 比較 1 イベントは出力をアクティブ状態に強制します。

ビット 2 **PER**：タイマ x 周期
タイマ A 周期イベントは出力をアクティブ状態に強制します。
注： アップ／ダウンモード（UDM ビットを 1 に設定）では、カウンタ周期イベントは OUTROM [1:0] ビットの設定に従って定義されます。

ビット 1 **RESYNC**：タイマ A 再同期
ソフトウェアまたは SYNC 入力からのみのタイマ A リセットイベントは出力をアクティブ状態に強制します。
注： RESYNC=1 のとき、その他のタイマリセットは出力に影響しません。

ビット 0 **SST**：ソフトウェア･セット･トリガ
このビットは出力をアクティブ状態に強制します。このビットは、ソフトウェアによってのみセットでき、ハードウェアによってリセットされます。
注： このビットはプリロードされません。

27.5.28 HRTIM タイマ x 出力 1 リセット･レジスタ（HRTIM_RSTx1R）（x = A ~ F）

アドレス･オフセット：ブロック A：0x0C0

アドレス･オフセット：ブロック B：0x140

アドレス･オフセット：ブロック C：0x1C0

アドレス･オフセット：ブロック D：0x240

アドレス･オフセット：ブロック E：0x2C0

アドレス･オフセット：ブロック F：0x340

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx1 出力を非アクティブ状態に強制できるソースを定義しています。

27.5.29 HRTIM タイマ x 出力 2 設定レジスタ（HRTIM_SETx2R）（x = A～F）

アドレス･オフセット：ブロック A：0x0C4

アドレス･オフセット：ブロック B：0x144

アドレス･オフセット：ブロック C：0x1C4

アドレス･オフセット：ブロック D：0x244

アドレス･オフセット：ブロック E：0x2C4

アドレス･オフセット：ブロック F：0x344

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx2 出力をアクティブ状態に強制するソースを定義しています。

27.5.30 HRTIM タイマ x 出力 2 リセットレジスタ（HRTIM_RSTx2R）（x = A～F）

アドレス･オフセット：ブロック A：0x0C8

アドレス･オフセット：ブロック B：0x148

アドレス･オフセット：ブロック C：0x1C8

アドレス･オフセット：ブロック D：0x248

アドレス･オフセット：ブロック E：0x2C8

アドレス･オフセット：ブロック F：0x348

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_SETx1R ビットの説明を参照してください。

これらのビットは、Tx2 出力を非アクティブ状態に強制するソースを定義しています。

27.5.31 HRTIM タイマ x 外部イベントフィルタリングレジスタ 1（HRTIM_EEFxR1）（x = A～F）

アドレス･オフセット：ブロック A：0x0CC

アドレス･オフセット：ブロック B：0x14C

アドレス･オフセット：ブロック C：0x1CC

アドレス･オフセット：ブロック D：0x24C

アドレス･オフセット：ブロック E：0x2CC

アドレス･オフセット：ブロック F：0x34C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE5FLTR [3:0]				EE5LTCH	Res.	EE4FLTR [3:0]				EE4LTCH	Res.	EE3FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE3FLTR [2:0]			EE3LTCH	Res.	EE2FLTR [3:0]				EE2LTCH	Res.	EE1FLTR [3:0]				EE1LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **EE5FLTR[3:0]**：外部イベント 5 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 24 **EE5LTCH**：外部イベント 5 ラッチ
EE1LTCH の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **EE4FLTR[3:0]**：外部イベント 4 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 18 **EE4LTCH**：外部イベント 4 ラッチ
EE1LTCH の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:13 **EE3FLTR[3:0]**：外部イベント 3 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 12 **EE3LTCH**：外部イベント 3 ラッチ
EE1LTCH の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **EE2FLTR[3:0]**：外部イベント 2 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 6 **EE2LTCH**：外部イベント 2 ラッチ
EE1LTCH の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:1 **EE1FLTR[3:0]**：外部イベント 1 フィルタ

0000： フィルタリングなし

0001： カウンタのリセット／ロールオーバーから比較 1 までブランキング

0010： アップカウント･モード（UDM ビットはリセット）で、カウンタのリセット／ロールオーバーから比較 2 までブランキング
アップダウンカウントモード（UDM ビットのセット）で比較 1 から比較 2 までブランキング（アップカウントフェーズ中のみ）

0011： カウンタのリセット／ロールオーバーから比較 3 までブランキング

0100： カウンタのリセット／ロールオーバーから比較 4 までブランキング

0100： アップカウント･モード（UDM ビットはリセット）で、カウンタのリセット／ロールオーバーから比較 4 までブランキング
アップダウンカウントモード（UDM ビットのセット）で比較 3 から比較 4 までブランキング（アップカウントフェーズ中のみ）

0101： 別のタイミング･ユニットからブランキング：TIMFLTR1 ソース（詳細は表 225 を参照）

0110： 別のタイミング･ユニットからブランキング：TIMFLTR2 ソース（詳細は表 225 を参照）

0111： 別のタイミング･ユニットからブランキング：TIMFLTR3 ソース（詳細は表 225 を参照）

1000： 別のタイミング･ユニットからブランキング：TIMFLTR4 ソース（詳細は表 225 を参照）

1001： 別のタイミング･ユニットからブランキング：TIMFLTR5 ソース（詳細は表 225 を参照）

1010： 別のタイミング･ユニットからブランキング：TIMFLTR6 ソース（詳細は表 225 を参照）

1011： 別のタイミング･ユニットからブランキング：TIMFLTR7 ソース（詳細は表 225 を参照）

1100： 別のタイミング･ユニットからブランキング：TIMFLTR8 ソース（詳細は表 225 を参照）

1101： アップカウント･モード（UDM ビットはリセット）で、カウンタのリセット／ロールオーバーから比較 2 までウィンドウイング
アップダウンカウントモード（UDM ビットのセット）で、比較 2 から比較 3 までウィンドウイング（アップカウントフェーズ中のみ）

1110： アップカウント･モード（UDM ビットはリセット）で、カウンタのリセット／ロールオーバーから比較 3 までウィンドウイング
アップダウンカウントモード（UDM ビットのセット）で、比較 2 から比較 3 までウィンドウイング（ダウンカウントフェーズ中のみ）

1111： 別のタイミング･ユニットからウィンドウイング：アップカウントモード（UDM ビットはリセット）での TIMWIN ソース（詳細は表 226 を参照）
アップダウンカウントモード（UDM ビットのセット）で、アップカウントフェーズ中の比較 2 からダウンカウントフェーズ中の比較 3 までウィンドウイング

注： 比較レジスタがフィルタリングに使用される時は、その値は必ず 0 より大きくなければなりません。
このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 0 **EE1LTCH**：外部イベント 1 ラッチ

0：イベント 1 は、ブランク中に発生した場合は無視され、ウィンドウ中に発生した場合はパススルーされます。

1：イベント 1 はラッチされ、ブランキングまたはウィンドウイング期間の終りまで遅延されます。

注： ウィンドウモード（EE1FLTR[3:0]=1101、1110、1111）では、EE1LTCH = 0 の場合、タイムアウトイベントが生成されます。ただし、外部イベントが高速モードでプログラムされている（EExFAST = 1）場合は除きます。
このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

27.5.32 HRTIM タイマ x 外部イベントフィルタリングレジスタ 2（HRTIM_EEFxR2）（x = A～F）

アドレス･オフセット：ブロック A：0x0D0

アドレス･オフセット：ブロック B：0x150

アドレス･オフセット：ブロック C：0x1D0

アドレス･オフセット：ブロック D：0x250

アドレス･オフセット：ブロック E：0x2D0

アドレス･オフセット：ブロック F：0x350

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10FLTR [3:0]				EE10LTCH	Res.	EE9FLTR [3:0]				EE9LTCH	Res.	EE8FLTR[3]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8FLTR [2:0]			EE8LTCH	Res.	EE7FLTR [3:0]				EE7LTCH	Res.	EE6FLTR [3:0]				EE6LTCH
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 **EE10FLTR[3:0]**：外部イベント 10 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 24 **EE10LTCH**：外部イベント 10 ラッチ
EE1LTCH の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **EE9FLTR[3:0]**：外部イベント 9 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 18 **EE9LTCH**：外部イベント 9 ラッチ
EE1LTCH の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:13 **EE8FLTR[3:0]**：外部イベント 8 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 12 **EE8LTCH**：外部イベント 8 ラッチ
EE1LTCH の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **EE7FLTR[3:0]**：外部イベント 7 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 6 **EE7LTCH**：外部イベント 7 ラッチ
EE1LTCH の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:1 **EE6FLTR[3:0]**：外部イベント 6 フィルタ
EE1FLTR[3:0] の説明を参照してください。

ビット 0 **EE6LTCH**：外部イベント 6 ラッチ
EE1LTCH の説明を参照してください。

27.5.33 HRTIM タイマ A リセットレジスタ（HRTIM_RSTAR）

アドレスオフセット：0x0D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMF CMP2	TIME CMP4	TIME CMP2	TIME CMP1	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	EXT EVNT 10	EXT EVNT9	EXT EVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP2	UPDT	TIMF CMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMFCPM2**：タイマ F の比較 2
タイマ F の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 30 TIME**CPM4**：タイマ E の比較 4
タイマ E の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 29 **T**IM**ECMP2**：タイマ E の比較 2
タイマ E の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 28 **T**IM**ECMP1**：タイマ E の比較 1
タイマ E の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 27 **T**IM**DCMP4**：タイマ D の比較 4
タイマ D の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 26 **T**IM**DCMP2**：タイマ D の比較 2
タイマ D の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 25 **T**IM**DCMP1**：タイマ D の比較 1
タイマ D の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 24 **T**IM**CCMP4**：タイマ C の比較 4
タイマ C の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 23 **T**IM**CCMP2**：タイマ C の比較 2
タイマ C の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 22 **T**IM**CCMP1**：タイマ C の比較 1
タイマ C の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 21 **T**IM**BCMP4**：タイマ B の比較 4
タイマ B の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 20 **T**IM**BCMP2**：タイマ B の比較 2
タイマ B の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 19 **T**IM**BCMP1**：タイマ B の比較 1
タイマ B の比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ A カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ A カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ A カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ A カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ A カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ A カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ A カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ A カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ A カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ A カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ A カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ A カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ A カウンタがリセットされます。

ビット 3 **CMP4**：タイマ A の比較 4 リセット
タイマ A の比較 4 イベント時にタイマ A カウンタがリセットされます。

ビット 2 **CMP2**：タイマ A の比較 2 リセット
タイマ A の比較 2 イベント時にタイマ A カウンタがリセットされます。

ビット 1 **UPDT**：タイマ A 更新リセット
タイマ A カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMFCMP1**：タイマ F の比較 1
タイマ F の比較 1 イベント時にタイマ A カウンタがリセットされます。

27.5.34 HRTIM タイマ B リセットレジスタ（HRTIM_RSTBR）

アドレスオフセット：0x154

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMF CMP2	TIME CMP4	TIME CMP2	TIME CMP1	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXT EVNT 10	EXT EVNT9	EXT EVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP2	UPDT	TIMF CMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMFCPM2**：タイマ F の比較 2
タイマ F の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 30 TIME**CPM4**：タイマ E の比較 4
タイマ E の比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 29 **T**IM**ECMP2**：タイマ E の比較 2
タイマ E の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 28 **T**IM**ECMP1**：タイマ E の比較 1
タイマ E の比較 1 イベント時にタイマ B カウンタがリセットされます。

ビット 27 **T**IM**DCMP4**：タイマ D の比較 4
タイマ D の比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 26 **T**IM**DCMP2**：タイマ D の比較 2
タイマ D の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 25 **T**IM**DCMP1**：タイマ D の比較 1
タイマ D の比較 1 イベント時にタイマ B カウンタがリセットされます。

ビット 24 **T**IM**CCMP4**：タイマ C の比較 4
タイマ C の比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 23 **T**IM**CCMP2**：タイマ C の比較 2
タイマ C の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 22 **T**IM**CCMP1**：タイマ C の比較 1
タイマ C の比較 1 イベント時にタイマ B カウンタがリセットされます。

ビット 21 **T**IMA**CMP4**：タイマ A の比較 4
タイマ A の比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 20 **T**IMA**CMP2**：タイマ A の比較 2
タイマ A の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 19 **T**IMA**CMP1**：タイマ A の比較 1
タイマ A の比較 1 イベント時にタイマ B カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ B カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ B カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ B カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ B カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ B カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ B カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ B カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ B カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ B カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ B カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ B カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ B カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ B カウンタがリセットされます。

ビット 3 **CMP4**：タイマ B の比較 4 リセット
タイマ B の比較 4 イベント時にタイマ B カウンタがリセットされます。

ビット 2 **CMP2**：タイマ B の比較 2 リセット
タイマ B の比較 2 イベント時にタイマ B カウンタがリセットされます。

ビット 1 **UPDT**：タイマ B 更新リセット
タイマ B カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMFCMP1**：タイマ F の比較 1
タイマ F の比較 1 イベント時にタイマ B カウンタがリセットされます。

27.5.35 HRTIM タイマ C リセットレジスタ（HRTIM_RSTCR）

アドレスオフセット：0x2D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMF CMP2	TIME CMP4	TIME CMP2	TIME CMP1	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXT EVNT 10	EXT EVNT9	EXT EVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP2	UPDT	TIMF CMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMFCPM2**：タイマ F の比較 2
タイマ F の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 30 TIME**CPM4**：タイマ E の比較 4
タイマ E の比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 29 **T**IM**ECMP2**：タイマ E の比較 2
タイマ E の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 28 **T**IM**ECMP1**：タイマ E の比較 1
タイマ E の比較 1 イベント時にタイマ C カウンタがリセットされます。

ビット 27 **T**IM**DCMP4**：タイマ D の比較 4
タイマ D の比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 26 **T**IM**DCMP2**：タイマ D の比較 2
タイマ D の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 25 **T**IM**DCMP1**：タイマ D の比較 1
タイマ D の比較 1 イベント時にタイマ C カウンタがリセットされます。

ビット 24 **T**IM**BCMP4**：タイマ B の比較 4
タイマ B の比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 23 **T**IM**BCMP2**：タイマ B の比較 2
タイマ B の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 22 **T**IM**BCMP1**：タイマ B の比較 1
タイマ B の比較 1 イベント時にタイマ C カウンタがリセットされます。

ビット 21 **T**IMA**CMP4**：タイマ A の比較 4
タイマ A の比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 20 **T**IMA**CMP2**：タイマ A の比較 2
タイマ A の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 19 **T**IMA**CMP1**：タイマ A の比較 1
タイマ A の比較 1 イベント時にタイマ C カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ C カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ C カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ C カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ C カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ C カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ C カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ C カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ C カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ C カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ C カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ C カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ C カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ C カウンタがリセットされます。

ビット 3 **CMP4**：タイマ C の比較 4 リセット
タイマ C の比較 4 イベント時にタイマ C カウンタがリセットされます。

ビット 2 **CMP2**：タイマ C の比較 2 リセット
タイマ C の比較 2 イベント時にタイマ C カウンタがリセットされます。

ビット 1 **UPDT**：タイマ C 更新リセット
タイマ C カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMFCMP1**：タイマ F の比較 1
タイマ F の比較 1 イベント時にタイマ C カウンタがリセットされます。

27.5.36 HRTIM タイマ D リセットレジスタ（HRTIM_RSTDR）

アドレスオフセット：0x254

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMF CMP2	TIME CMP4	TIME CMP2	TIME CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXT EVNT 10	EXT EVNT9	EXT EVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP2	UPDT	TIMF CMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMFCPM2**：タイマ F の比較 2
タイマ F の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 30 TIME**CPM4**：タイマ E の比較 4
タイマ E の比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 29 **T**IM**ECMP2**：タイマ E の比較 2
タイマ E の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 28 **T**IM**ECMP1**：タイマ E の比較 1
タイマ E の比較 1 イベント時にタイマ D カウンタがリセットされます。

ビット 27 **T**IM**CCMP4**：タイマ C の比較 4
タイマ C の比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 26 **T**IM**CCMP2**：タイマ C の比較 2
タイマ C の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 25 **T**IM**CCMP1**：タイマ C の比較 1
タイマ C の比較 1 イベント時にタイマ D カウンタがリセットされます。

ビット 24 **T**IM**BCMP4**：タイマ B の比較 4
タイマ B の比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 23 **T**IM**BCMP2**：タイマ B の比較 2
タイマ B の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 22 **T**IM**BCMP1**：タイマ B の比較 1
タイマ B の比較 1 イベント時にタイマ D カウンタがリセットされます。

ビット 21 **T**IMA**CMP4**：タイマ A の比較 4
タイマ A の比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 20 **T**IMA**CMP2**：タイマ A の比較 2
タイマ A の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 19 **T**IMA**CMP1**：タイマ A の比較 1
タイマ A の比較 1 イベント時にタイマ D カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ D カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ D カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ D カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ D カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ D カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ D カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ D カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ D カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ D カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ D カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ D カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ D カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ D カウンタがリセットされます。

ビット 3 **CMP4**：タイマ D の比較 4 リセット
タイマ D の比較 4 イベント時にタイマ D カウンタがリセットされます。

ビット 2 **CMP2**：タイマ D の比較 2 リセット
タイマ D の比較 2 イベント時にタイマ D カウンタがリセットされます。

ビット 1 **UPDT**：タイマ D 更新リセット
タイマ D カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMFCMP1**：タイマ F の比較 1
タイマ F の比較 1 イベント時にタイマ D カウンタがリセットされます。

27.5.37 HRTIM タイマ E リセットレジスタ（HRTIM_RSTER）

アドレスオフセット：0x2D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMF CMP2	TIMD CMP4	TIMD CMP2	TIMD CMP1	TIMC CMP4	TIMC CMP2	TIMC CMP1	TIMB CMP4	TIMB CMP2	TIMB CMP1	TIMA CMP4	TIMA CMP2	TIMA CMP1	EXT EVNT 10	EXT EVNT9	EXT EVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXT EVNT7	EXT EVNT6	EXT EVNT5	EXT EVNT4	EXT EVNT3	EXT EVNT2	EXT EVNT1	MST CMP4	MST CMP3	MST CMP2	MST CMP1	MST PER	CMP4	CMP2	UPDT	TIMF CMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMFCPM2**：タイマ F の比較 2
タイマ F の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 30 TIMD**CPM4**：タイマ D の比較 4
タイマ D の比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 29 **T**IM**DCMP2**：タイマ D の比較 2
タイマ D の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 28 **T**IM**DCMP1**：タイマ D の比較 1
タイマ D の比較 1 イベント時にタイマ E カウンタがリセットされます。

ビット 27 **T**IM**CCMP4**：タイマ C の比較 4
タイマ C の比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 26 **T**IM**CCMP2**：タイマ C の比較 2
タイマ C の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 25 **T**IM**CCMP1**：タイマ C の比較 1
タイマ C の比較 1 イベント時にタイマ E カウンタがリセットされます。

ビット 24 **T**IM**BCMP4**：タイマ B の比較 4
タイマ B の比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 23 **T**IM**BCMP2**：タイマ B の比較 2
タイマ B の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 22 **T**IMB**CMP1**：タイマ B の比較 1
タイマ B の比較 1 イベント時にタイマ E カウンタがリセットされます。

ビット 21 **T**IMA**CMP4**：タイマ A の比較 4
タイマ A の比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 20 **T**IMA**CMP2**：タイマ A の比較 2
タイマ A の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 19 **T**IMA**CMP1**：タイマ A の比較 1
タイマ A の比較 1 イベント時にタイマ E カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ E カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ E カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ E カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ E カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ E カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ E カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ E カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ E カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ E カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ E カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ E カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ E カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ E カウンタがリセットされます。

ビット 3 **CMP4**：タイマ E の比較 4 リセット
タイマ E の比較 4 イベント時にタイマ E カウンタがリセットされます。

ビット 2 **CMP2**：タイマ E の比較 2 リセット
タイマ E の比較 2 イベント時にタイマ E カウンタがリセットされます。

ビット 1 **UPDT**：タイマ E 更新リセット
タイマ E カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMFCMP1**：タイマ F の比較 1
タイマ F の比較 1 イベント時にタイマ E カウンタがリセットされます。

27.5.38 HRTIM タイマ F リセットレジスタ（HRTIM_RSTFR）

アドレスオフセット：0x354

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TIMECMP2	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMECMP1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TIMECPM2**：タイマ E の比較 2
タイマ E の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 30 TIMDC**PM4**：タイマ D の比較 4
タイマ D の比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 29 **T**IM**DCMP2**：タイマ D の比較 2
タイマ D の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 28 **T**IMD**CMP1**：タイマ D の比較 1
タイマ D の比較 1 イベント時にタイマ F カウンタがリセットされます。

ビット 27 **T**IM**CCMP4**：タイマ C の比較 4
タイマ C の比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 26 **T**IMC**CMP2**：タイマ C の比較 2
タイマ C の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 25 **T**IMC**CMP1**：タイマ C の比較 1
タイマ C の比較 1 イベント時にタイマ F カウンタがリセットされます。

ビット 24 **T**IMB**CMP4**：タイマ B の比較 4
タイマ B の比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 23 **T**IMB**CMP2**：タイマ B の比較 2
タイマ B の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 22 **T**IMB**CMP1**：タイマ B の比較 1
タイマ B の比較 1 イベント時にタイマ F カウンタがリセットされます。

ビット 21 **T**IMA**CMP4**：タイマ A の比較 4
タイマ A の比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 20 **T**IMA**CMP2**：タイマ A の比較 2
タイマ A の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 19 **T**IMA**CMP1**：タイマ A の比較 1
タイマ A の比較 1 イベント時にタイマ F カウンタがリセットされます。

ビット 18 **EXTEVNT10**：外部イベント
外部イベント 10 の時にタイマ F カウンタがリセットされます。

ビット 17 **EXTEVNT9**：外部イベント 9
外部イベント 9 の時にタイマ F カウンタがリセットされます。

ビット 16 **EXTEVNT8**：外部イベント 8
外部イベント 8 の時にタイマ F カウンタがリセットされます。

ビット 15 **EXTEVNT7**：外部イベント 7
外部イベント 7 の時にタイマ F カウンタがリセットされます。

ビット 14 **EXTEVNT6**：外部イベント 6
外部イベント 6 の時にタイマ F カウンタがリセットされます。

ビット 13 **EXTEVNT5**：外部イベント 5
外部イベント 5 の時にタイマ F カウンタがリセットされます。

ビット 12 **EXTEVNT4**：外部イベント 4
外部イベント 4 の時にタイマ F カウンタがリセットされます。

ビット 11 **EXTEVNT3**：外部イベント 3
外部イベント 3 の時にタイマ F カウンタがリセットされます。

ビット 10 **EXTEVNT2**：外部イベント 2
外部イベント 2 の時にタイマ F カウンタがリセットされます。

ビット 9 **EXTEVNT1**：外部イベント 1
外部イベント 1 の時にタイマ F カウンタがリセットされます。

ビット 8 **MSTCMP4**：マスタ比較 4
マスタタイマの比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 7 **MSTCMP3**：マスタ比較 3
マスタタイマの比較 3 イベント時にタイマ F カウンタがリセットされます。

ビット 6 **MSTCMP2**：マスタ比較 2
マスタタイマの比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 5 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベント時にタイマ F カウンタがリセットされます。

ビット 4 **MSTPER** マスタタイマ周期
マスタタイマの周期イベント時にタイマ F カウンタがリセットされます。

ビット 3 **CMP4**：タイマ F の比較 4 リセット
タイマ F の比較 4 イベント時にタイマ F カウンタがリセットされます。

ビット 2 **CMP2**：タイマ F の比較 2 リセット
タイマ F の比較 2 イベント時にタイマ F カウンタがリセットされます。

ビット 1 **UPDT**：タイマ F 更新リセット
タイマ F カウンタは、更新イベント時にリセットされます。

ビット 0 **TIMECMP1**：タイマ E の比較 1
タイマ E の比較 1 イベント時にタイマ F カウンタがリセットされます。

27.5.39 HRTIM タイマ x チョッパレジスタ（HRTIM_CHPxR）（x = A～F）

アドレスオフセット：ブロック A：0x0D8

アドレス･オフセット：ブロック B：0x158

アドレス･オフセット：ブロック C：0x1D8

アドレス･オフセット：ブロック D：0x258

アドレス･オフセット：ブロック E：0x2D8

アドレス･オフセット：ブロック F：0x358

リセット値：0x0000 0000

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

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

ビット 31:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:7 **STRPW[3:0]**：タイマ x 開始パルス幅

このレジスタは、出力信号の立ち上がりエッジ後の最初のパルス幅を定義します。
このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
t_{1STPW} = （STRPW[3:0]+1）x 16 x t_{HRTIM}

0000：94.1 ns（1/10.625 MHz）
.....
1111：1.51 µs（16/10.625 MHz）

ビット 6:4 **CARDTY[2:0]**：タイマ x チョッパ･デューティ･サイクル値

このレジスタは、キャリア信号のデューティサイクルを定義します。このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
000 :0/8（最初のパルスのみが存在します）
.....
111 :7/8

ビット 3:0 **CARFRQ[3:0]**：タイマ x キャリア周波数の値

このレジスタは、キャリア周波数 F_{CHPFRQ} = f_{HRTIM} /（16 x （CARFRQ[3:0]+1））を定義します。
このビットフィールドは、CHPx ビットの 1 つがセットされているとき、変更できません。
0000 :10.625 MHz（f_{HRTIM}/ 16）
.....
1111 :664.1 kHz（f_{HRTIM} / 256）

27.5.40 HRTIM タイマ A キャプチャ 1 制御レジスタ（HRTIM_CPT1ACR）

アドレス･オフセット：0x0DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TE CMP2	TE CMP1	TE1 RST	TE1 SET	TD CMP2	TD CMP1	TD1 RST	TD1 SET	TC CMP2	TC CMP1	TC1 RST	TC1 SET	TB CMP2	TB CMP1	TB1 RST	TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TF CMP2	TF CMP1	TF1 RST	TF1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.41 HRTIM タイマ B キャプチャ 1 制御レジスタ（HRTIM_CPT1BCR）

アドレス･オフセット：0x15C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TE CMP2	TE CMP1	TE1 RST	TE1 SET	TD CMP2	TD CMP1	TD1 RST	TD1 SET	TC CMP2	TC CMP1	TC1 RST	TC1 SET	TF CMP2	TF CMP1	TF1 RST	TF1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA CMP2	TA CMP1	TA1 RST	TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.42 HRTIM タイマ C キャプチャ 1 制御レジスタ（HRTIM_CPT1CCR）

アドレス･オフセット：0x1DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TE CMP2	TE CMP1	TE1 RST	TE1 SET	TD CMP2	TD CMP1	TD1 RST	TD1 SET	TF CMP2	TF CMP1	TF1 RST	TF1 SET	TB CMP2	TB CMP1	TB1 RST	TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA CMP2	TA CMP1	TA1 RST	TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.43 HRTIM タイマ D キャプチャ 1 制御レジスタ（HRTIM_CPT1DCR）

アドレス･オフセット：0x25C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TE CMP2	TE CMP1	TE1 RST	TE1 SET	TF CMP2	TF CMP1	TF1 RST	TF1 SET	TC CMP2	TC CMP1	TC1 RST	TC1 SET	TB CMP2	TB CMP1	TB1 RST	TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA CMP2	TA CMP1	TA1 RST	TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.44 HRTIM タイマ E キャプチャ 1 制御レジスタ（HRTIM_CPT1ECR）

アドレス･オフセット：0x2DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TF CMP2	TF CMP1	TF1 RST	TF1 SET	TD CMP2	TD CMP1	TD1 RST	TD1 SET	TC CMP2	TC CMP1	TC1 RST	TC1 SET	TB CMP2	TB CMP1	TB1 RST	TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA CMP2	TA CMP1	TA1 RST	TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.45 HRTIM タイマ F キャプチャ 1 制御レジスタ（HRTIM_CPT1FCR）

アドレス･オフセット：0x35C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TE CMP2	TE CMP1	TE1 RST	TE1 SET	TD CMP2	TD CMP1	TD1 RST	TD1 SET	TC CMP2	TC CMP1	TC1 RST	TC1 SET	TB CMP2	TB CMP1	TB1 RST	TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TA CMP2	TA CMP1	TA1 RST	TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_CPT2xCR ビットの説明を参照してください。

27.5.46 HRTIM タイマ x キャプチャ 2 制御レジスタ（HRTIM_CPT2xCR）（x = A～F）

アドレス･オフセット：ブロック A：0x0E0

アドレス･オフセット：ブロック B：0x160

アドレス･オフセット：ブロック C：0x1E0

アドレス･オフセット：ブロック D：0x260

アドレス･オフセット：ブロック E：0x2E0

アドレス･オフセット：ブロック F：0x360

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TF CMP2 / TE CMP2	TF CMP1 / TE CMP1	TF1 RST / TE1 RST	TF1 SET / TE1 SET	TF CMP2 / TD CMP2	TF CMP1 / TD CMP1	TF1 RST / TD1 RST	TF1 SET / TD1 SET	TF CMP2 / TC CMP2	TF CMP1 / TC CMP1	TF1 RST / TC1 RST	TF1 SET / TC1 SET	TF CMP2 / TB CMP2	TF CMP1 / TB CMP1	TF1 RST / TB1 RST	TF1 SET / TB1 SET
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TF CMP2 / TA CMP2	TF CMP1 / TA CMP1	TF1 RST / TA1 RST	TF1 SET / TA1 SET	EXEV10CPT	EXEV9 CPT	EXEV8 CPT	EXEV7 CPT	EXEV6 CPT	EXEV5 CPT	EXEV4 CPT	EXEV3 CPT	EXEV2 CPT	EXEV1 CPT	UPD CPT	SW CPT
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2FCR において：

TECMP2：タイマ E の比較 2

TACMP2 の説明を参照してください。

HRTIM_CPT2ECR：

TFCMP2：タイマ F の比較 2

TACMP2 の説明を参照してください。

ビット 30　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2FCR において：

TECMP1：タイマ E の比較 1

TACMP1 の説明を参照してください。

HRTIM_CPT2ECR：

TFCMP1：タイマ F の比較 1

TACMP1 の説明を参照してください。

ビット 29　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2FCR において：

TE1RST：タイマ E の出力 1 リセット

TA1RST の説明を参照してください。

HRTIM_CPT2ECR：

TF1RST：タイマ F の出力 1 リセット

TA1RST の説明を参照してください。

ビット 28 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2FCR において：
TE1SET：タイマ E の出力 1 セット
TA1SET の説明を参照してください。

HRTIM_CPT2ECR：
TF1SET：タイマ F の出力 1 セット
TA1SET の説明を参照してください。

ビット 27 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TDCMP2：タイマ D の比較 2
TACMP2 の説明を参照してください。

HRTIM_CPT2DCR：
TFCMP2：タイマ F の比較 2
TACMP2 の説明を参照してください。

ビット 26 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TDCMP1：タイマ D の比較 1
TACMP1 の説明を参照してください。

HRTIM_CPT2DCR：
TFCMP1：タイマ F の比較 1
TACMP1 の説明を参照してください。

ビット 25 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TD1RST：タイマ D の出力 1 リセット
TA1RST の説明を参照してください。

HRTIM_CPT2DCR：
TF1RST：タイマ F の出力 1 リセット
TA1RST の説明を参照してください。

ビット 24 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TD1SET：タイマ D の出力 1 セット
TA1SET の説明を参照してください。

HRTIM_CPT2DCR：
TF1SET：タイマ F の出力 1 セット
TA1SET の説明を参照してください。

ビット 23 HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TCCMP2：タイマ C の比較 2
TACMP2 の説明を参照してください。

HRTIM_CPT2CCR：
TFCMP2：タイマ F の比較 2
TACMP2 の説明を参照してください。

ビット 22　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TCCMP1：タイマ C の比較 1

TACMP1 の説明を参照してください。

HRTIM_CPT2CCR：

TFCMP1：タイマ F の比較 1

TACMP1 の説明を参照してください。

ビット 21　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TC1RST：タイマ C の出力 1 リセット

TA1RST の説明を参照してください。

HRTIM_CPT2CCR：

TF1RST：タイマ F の出力 1 リセット

TA1RST の説明を参照してください。

ビット 20　HRTIM_CPT2ACR、HRTIM_CPT2BCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TC1SET：タイマ C の出力 1 セット

TA1SET の説明を参照してください。

HRTIM_CPT2CCR：

TF1SET：タイマ F の出力 1 セット

TA1SET の説明を参照してください。

ビット 19　HRTIM_CPT2ACR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TBCMP2：タイマ B の比較 2

TACMP2 の説明を参照してください。

HRTIM_CPT2BCR：

TFCMP2：タイマ F の比較 2

TACMP2 の説明を参照してください。

ビット 18　HRTIM_CPT2ACR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TBCMP1：タイマ B の比較 1

TACMP1 の説明を参照してください。

HRTIM_CPT2BCR：

TFCMP1：タイマ F の比較 1

TACMP1 の説明を参照してください。

ビット 17　HRTIM_CPT2ACR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：

TB1RST：タイマ B の出力 1 リセット

TA1RST の説明を参照してください。

HRTIM_CPT2BCR：

TF1RST：タイマ F の出力 1 リセット

TA1RST の説明を参照してください。

ビット 16 HRTIM_CPT2ACR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TB1SET：タイマ B の出力 1 セット
TA1SET の説明を参照してください。

HRTIM_CPT2BCR：
TF1SET：タイマ F の出力 1 セット
TA1SET の説明を参照してください。

ビット 15 HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TACMP2：タイマ A の比較 2
0：影響なし。
1：タイマ A 比較 2 はキャプチャ 2 をトリガします。

HRTIM_CPT2ACR：
TFCMP2：タイマ E の比較 2
TACMP2 の説明を参照してください。

ビット 14 HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TACMP1：タイマ A の比較 1
0：影響なし。
1：タイマ A 比較 1 はキャプチャ 2 をトリガします。

HRTIM_CPT2ACR：
TFCMP1：タイマ F の比較 1
TACMP1 の説明を参照してください。

ビット 13 HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TA1RST：タイマ B の出力 1 リセット
0：影響なし。
1：キャプチャ 2 は TA1 出力のアクティブから非アクティブへの遷移によってトリガされます。

HRTIM_CPT2ACR：
TF1RST：タイマ F の出力 1 リセット
TA1RST の説明を参照してください。

ビット 12 HRTIM_CPT2BCR、HRTIM_CPT2CCR、HRTIM_CPT2DCR、HRTIM_CPT2ECR、HRTIM_CPT2FCR において：
TA1SET：タイマ B の出力 1 セット
0：影響なし。
1：キャプチャ 2 は TA1 出力の非アクティブからアクティブへの遷移によってトリガされます。

HRTIM_CPT2ACR：
TF1SET：タイマ F の出力 1 セット
TA1SET の説明を参照してください。

ビット 11 **EXEV10CPT**：外部イベント 10 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 10 **EXEV9CPT**：外部イベント 9 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 9 **EXEV8CPT**：外部イベント 8 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 8 **EXEV7CPT**：外部イベント 7 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 7 **EXEV6CPT**：外部イベント 6 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 6 **EXEV5CPT**：外部イベント 5 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 5 **EXEV4CPT**：外部イベント 4 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 4 **EXEV3CPT**：外部イベント 3 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 3 **EXEV2CPT**：外部イベント 2 キャプチャ
EXEV1CPT の説明を参照してください。

ビット 2 **EXEV1CPT**：外部イベント 1 キャプチャ
0：影響なし。
1：外部イベント 1 はキャプチャ 2 をトリガします。

ビット 1 **UPDCPT**：更新キャプチャ
0：影響なし。
1：更新イベントはキャプチャ 2 をトリガします。

ビット 0 **SWCPT**：ソフトウェア･キャプチャ
0：影響なし。
1：このビットはソフトウェアによるキャプチャ 2 を強制します。このビットは、セットのみで、ハードウェアによってリセットされます。

27.5.47 HRTIM タイマ x 出力レジスタ（HRTIM_OUTxR）（x = A～F）

アドレスオフセット：ブロック A：0x0E4

アドレス･オフセット：ブロック B：0x164

アドレス･オフセット：ブロック C：0x1E4

アドレス･オフセット：ブロック D：0x264

アドレス･オフセット：ブロック E：0x2E4

アドレス･オフセット：ブロック F：0x364

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM 2	POL2	Res.
								rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	BIAR	Res.	DLYPRT[2:0]			DLYPR TEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM 1	POL1	Res.
	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **DIDL2**：バーストモードアイドル移行時の出力 2 デッドタイム

このビットは、出力をアイドル状態に切り替える前にデッドタイム挿入を強制することによって、アイドルモードへの移行を遅らせます。この設定は、バーストモード動作中にアイドル状態に移行する時にのみ適用されます。

0：プログラムされたアイドル状態は直ちに出力 2 に適用されます。

1：アイドル･モードに移行する前に、デッドタイム（非アクティブ･レベル）が出力 2 に挿入されます。デッドタイム値は DTFx[8:0] によってセットされます。

注： このパラメータは、タイマ x が有効化された後は、変更できません。

バーストモード中に出力の 1 つがアクティブで（IDLES=1）、正のデッドタイム（SDTR/SDTF が 0 にセットされている）の場合のみ、DIDL=1 をセットします。

ビット 22 **CHP2**：出力 2 チョッパイネーブル

このビットは、出力 2 にチョッパを有効にします。

0：出力信号は変更されません。

1：出力信号はキャリア信号でチョップ処理されます。

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 21:20 **FAULT2[1:0]**：出力 2 フォールト状態

これらのビットは、フォールトイベント後の出力 2 の状態を選択します。

00：作用なし：出力はフォールト入力に影響されず、動作モードに留まります。

01：アクティブ

10：非アクティブ

11：ハイ･インピーダンス

注： FLTENx ビットがセットされている場合、または出力が FAULT 状態の場合は、このパラメータは、いったんタイマ x が有効になると（TxCEN ビットのセット）、変更できません。

ビット 19 **IDLES2**：出力 2 アイドル状態

このビットは出力 2 のアイドル状態を選択します。

0：非アクティブ

1：アクティブ

注： このパラメータは、HRTIM が出力を制御する前にセットする必要があります。

ビット 18 **IDLEM2**：出力 2 アイドルモード

このビットは出力 2 のアイドルモードを選択します。

0：作用なし：出力はバースト･モード動作の影響を受けません。

1：バースト･モード･コントローラによってリクエストされると、出力はアイドル状態になります。

注： このビットはプリロードされ、実行中に変更されますが、バーストモードがアクティブである間は、変更してはなりません。

ビット 17 **POL2**：出力 2 極性

このビットは出力 2 の極性を選択します。

0：正の極性（出力はアクティブ･ハイ）

1：負の極性（出力はアクティブ･ロー）

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 16:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **BIAR**：バランス･アイドルの自動再開

このビットは、バランスアイドルイベント後に出力が自動的に再有効化されることを選択します。

このビットは、DLYPRT[2:0] = 011 または 111 の場合にのみ、意味があります。そうでない場合は、無視されます。

0：無効

1：有効

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12:10 **DLYPRT[2:0]**：遅延保護

これらのビットは、遅延保護スキームを適用するソースおよび出力を定義します。

HRTIM_OUTAR、HRTIM_OUTBR、HRTIM_OUTCR において：

000 :外部イベント 6 による出力 1 遅延アイドル

001 :外部イベント 6 による出力 2 遅延アイドル

010 :外部イベント 6 による出力 1 および出力 2 遅延アイドル

011 :外部イベント 6 によるバランス･アイドル

100 :外部イベント 7 による出力 1 遅延アイドル

101 :外部イベント 7 による出力 2 遅延アイドル

110 :外部イベント 7 による出力 1 および出力 2 遅延アイドル

111：外部イベント 7 によるバランス･アイドル

HRTIM_OUTDR、HRTIM_OUTER、HRTIM_OUTFR において：

000 :外部イベント 8 による出力 1 遅延アイドル

001 :外部イベント 8 による出力 2 遅延アイドル

010 :外部イベント 8 による出力 1 および出力 2 遅延アイドル

011 :外部イベント 8 によるバランス･アイドル

100 :外部イベント 9 による出力 1 遅延アイドル

101 :外部イベント 9 による出力 2 遅延アイドル

110 :外部イベント 9 による出力 1 および出力 2 遅延アイドル

111 :外部イベント 9 によるバランス･アイドル

注： このビットフィールドは、遅延保護が有効化（DLYPRTEN ビットのセット）された後には変更してはなりません。

ビット 9 **DLYPRTEN**：遅延保護イネーブル

このビットは、遅延保護スキームを有効にします。

0：影響なし。

1：DLYPRT[2:0] ビットに従い遅延保護が有効化されます。

注： このパラメータは、タイマ x が有効化（TxEN ビットのセット）された後は、変更できません。

ビット 8 **DTEN**：デッドタイム･イネーブル

このビットは、出力 1 と出力 2 へのデッドタイム挿入を有効化します。

0：出力 1 と出力 2 の信号は独立です。

1：デッドタイムが出力 1 と出力 2 の間に挿入されます（基準信号は出力 1 の信号ジェネレータです）。

注： このパラメータは、いったんタイマが動作すると（TxEN ビットのセット）、またはその出力が有効化され、別のタイマによってセット／リセットされる場合は、変更できません。

ビット 7 **DIDL1**：バーストモードアイドル移行時の出力 1 デッドタイム

このビットは、出力をアイドル状態に切り替える前にデッドタイム挿入を強制することによって、アイドルモードへの移行を遅らせます。この設定は、バーストモード動作中にアイドル状態に移行する時にのみ適用されます。

0：プログラムされたアイドル状態は直ちに出力 1 に適用されます。

1：アイドル･モードに移行する前に、デッドタイム（非アクティブ･レベル）が出力 1 に挿入されます。デッドタイム値は DTRx[8:0] によってセットされます。

注： このパラメータは、タイマ x が有効化された後は、変更できません。

バーストモード中に出力の 1 つがアクティブで（IDLES=1）、正のデッドタイム（SDTR/SDTF が 0 にセットされている）の場合のみ、DIDL=1 をセットします。

ビット 6 **CHP1**：出力 1 チョッパイネーブル

このビットは、出力 1 にチョッパを有効にします。

0：出力信号は変更されません。

1：出力信号はキャリア信号でチョップ処理されます。

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 5:4 **FAULT1[1:0]**：出力 1 フォールト状態

これらのビットは、フォールトイベント後の出力 1 の状態を選択します。

00：作用なし：出力はフォールト入力に影響されず、動作モードに留まります。

01：アクティブ

10：非アクティブ

11：ハイ･インピーダンス

注： FLTENx ビットがセットされている場合、または出力が FAULT 状態の場合は、このパラメータは、いったんタイマ x が有効になると（TxCEN ビットのセット）、変更できません。

ビット 3 **IDLES1**：出力 1 アイドル状態

このビットは出力 1 のアイドル状態を選択します。

0：非アクティブ

1：アクティブ

注： このパラメータは、HRTIM が出力を制御する前にセットする必要があります。

ビット 2 **IDLEM1**：出力 1 アイドルモード

このビットは出力 1 のアイドルモードを選択します。

0：作用なし：出力はバースト･モード動作の影響を受けません。

1：バースト･モード･コントローラによってリクエストされると、出力はアイドル状態になります。

注： このビットはプリロードされ、実行中に変更されますが、バーストモードがアクティブである間は、変更してはなりません。

ビット 1 **POL1**：出力 1 極性

このビットは出力 1 の極性を選択します。

0：正の極性（出力はアクティブ･ハイ）

1：負の極性（出力はアクティブ･ロー）

注： このパラメータは、タイマ x が有効化された後は、変更できません。

ビット 0 予約済みであり、リセット値に保持する必要があります。

27.5.48 HRTIM タイマ x フォールトレジスタ（HRTIM_FLTxR）（x = A～F）

アドレスオフセット：ブロック A：0x0E8

アドレス･オフセット：ブロック B：0x168

アドレス･オフセット：ブロック C：0x1E8

アドレス･オフセット：ブロック D：0x268

アドレス･オフセット：ブロック E：0x2E8

アドレス･オフセット：ブロック F：0x368

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6 EN	FLT5 EN	FLT4 EN	FLT3 EN	FLT2 EN	FLT1 EN
										rw	rw	rw	rw	rw	rw

ビット 31 **FLTLCK**：フォールト･ソース･ロック

0：FLT1EN ~ FLT6EN ビットは読出し／書込み用です。

1：FLT1EN ~ FLT6EN ビットは読出し専用です。

FLTLCK ビットは 1 回のみ書込み可能です。いったんセットされると、次のシステム･リセットまで変更できません。

ビット 30:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **FLT6EN**：フォールト 6 イネーブル

0：フォールト 6 入力は無視されます。

1：フォールト 6 入力はアクティブで、HRTIM 出力を無効にします。

ビット 4 **FLT5EN**：フォールト 5 イネーブル

0：フォールト 5 入力は無視されます。

1：フォールト 5 入力はアクティブで、HRTIM 出力を無効にします。

ビット 3 **FLT4EN**：フォールト 4 イネーブル

0：フォールト 4 入力は無視されます。

1：フォールト 4 入力はアクティブで、HRTIM 出力を無効にします。

ビット 2 **FLT3EN**：フォールト 3 イネーブル

0：フォールト 3 入力は無視されます。

1：フォールト 3 入力はアクティブで、HRTIM 出力を無効にします。

ビット 1 **FLT2EN**：フォールト 2 イネーブル

0：フォールト 2 入力は無視されます。

1：フォールト 2 入力はアクティブで、HRTIM 出力を無効にします。

ビット 0 **FLT1EN**：フォールト 1 イネーブル

0：フォールト 1 入力は無視されます。

1：フォールト 1 入力はアクティブで、HRTIM 出力を無効にします。

27.5.49 HRTIM タイマ x 制御レジスタ 2（HRTIM_TIMxCR2）（x = A～F）

アドレス･オフセット：ブロック A：0x0EC

アドレス･オフセット：ブロック B：0x16C

アドレス･オフセット：ブロック C：0x1EC

アドレス･オフセット：ブロック D：0x26C

アドレス･オフセット：ブロック E：0x2EC

アドレス･オフセット：ブロック F：0x36C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRG HLF	Res.	Res.	GT CMP3	GT CMP1
											rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FEROM[1:0]		BMROM[1:0]		ADROM[1:0]		OUTROM[1:0]		ROM[1:0]		Res.	UDM	Res.	DCDR	DCDS	DCDE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw		rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 20 **TRGHLF**：トリガハーフ･モード

このビットフィールドは、比較 2 レジスタが標準モード（カウンタが比較値に等しくなるとすぐに比較一致が発行される）で動作するか、トリガハーフモード（セクション ：トリガハーフモードを参照）で動作するかを定義します。

0：CMP2 レジスタはユーザによってのみ書き込まれます（標準比較モード）。

1：CMP2 値は、キャプチャ 1 イベントが発生するとすぐに、ハードウェアによってセットされます。キャプチャ 1 値を 2 で割った値がロードされます。初期値はユーザが書き込むことができますが（TRGHLF がリセットされているかぎり）、最初のキャプチャがトリガされると無視されます（プリロード･メカニズムは、TRGHLF ビットがセットされているとき、CMP2 に対して無効になります）。

注： このビットフィールドは、カウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **GTCMP3**：「より大きい」比較 3 の PWM モード

このビットは比較 3 の動作モードを定義します。

0：比較 3 イベントは、カウンタが比較値と等しいときに生成されます（比較一致モード）。

1：比較 3 イベントは、カウンタが比較値より大きいときに生成されます。比較値が動作中に変更された場合、新しい比較値は現在のカウンタ値と比較され、SET または RESET の出力が生成されます。

ビット 16 **GTCMP1**：「より大きい」比較 1 の PWM モード

このビットは比較 1 の動作モードを定義します。

0：比較 1 イベントは、カウンタが比較値と等しいときに生成されます（比較一致モード）。

1：比較 1 イベントは、カウンタが比較値より大きいときに生成されます。比較値が動作中に変更された場合、新しい比較値は現在のカウンタ値と比較され、SET または RESET の出力が生成されます。

ビット 15:14 **FEROM[1:0]**：フォールトおよびイベントのロールオーバー･モード

このビットは、アップダウンカウントモードでいつロールオーバーが生成されるかを定義します。これは、フォールトおよびイベントカウンタで使用されるロールオーバーイベントのみに関係します。

00：カウンタが 0 または HRTIM_PERxR の値に等しい時にイベントが生成されます。

01：カウンタが 0 に等しい時にイベントが生成されます。

10：カウンタが HRTIM_PERxR に等しい時にイベントが生成されます。

11：予約済みです。

注： この設定は、UDM ビットがセットされたときにのみ適用されます。そうでない場合は、意味を持ちません。

注： このビットフィールドは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 13:12 **BMROM[1:0]**：バースト･モードのロールオーバー･モード

このビットは、アップダウンカウントモードでいつロールオーバーが生成されるかを定義します。これは、バーストモードのトリガとしてのクロックとして、バーストモードコントローラで使用されるロールオーバーイベントのみに関係します。

00：カウンタが 0 または HRTIM_PERxR の値に等しい時にイベントが生成されます。

01：カウンタが 0 に等しい時にイベントが生成されます。

10：カウンタが HRTIM_PERxR に等しい時にイベントが生成されます。

11：予約済みです。

注： この設定は、UDM ビットがセットされたときにのみ適用されます。そうでない場合は、意味を持ちません。

注： このパラメータは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 11:10 **ADROM[1:0]**：ADC のロールオーバー･モード

このビットは、アップダウンカウントモードでいつロールオーバーが生成されるかを定義します。これは、ADC をトリガするロールオーバーイベントのみに関係します。

00：カウンタが 0 または HRTIM_PERxR の値に等しい時にイベントが生成されます。

01：カウンタが 0 に等しい時にイベントが生成されます。

10：カウンタが HRTIM_PERxR に等しい時にイベントが生成されます。

11：予約済みです。

注： この設定は、UDM ビットがセットされたときにのみ適用されます。そうでない場合は、意味を持ちません。

注： このビットフィールドは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 9:8 **OUTROM[1:0]**：出力のロールオーバー･モード

このビットは、アップダウンカウントモードでいつロールオーバーが生成されるかを定義します。これは、HRTIM_SETxyR および HRTIM_RSTxyR の設定に従って、出力をセットおよび／またはリセットするロールオーバーイベントのみに関係します。

00：カウンタが 0 または HRTIM_PERxR の値に等しい時にイベントが生成されます。

01：カウンタが 0 に等しい時にイベントが生成されます。

10：カウンタが HRTIM_PERxR に等しい時にイベントが生成されます。

11：予約済みです。

注： この設定は、UDM ビットがセットされたときにのみ適用されます。そうでない場合は、意味を持ちません。

注： このビットフィールドは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 7:6 **ROM[1:0]**：ロールオーバー･モード

このビットは、アップダウンカウントモードでいつロールオーバーが生成されるかを定義します。これは、更新トリガ（プリロードからアクティブレジスタに内容を転送するための）、IRQ および DMA リクエスト、繰り返しカウンタのデクリメント、および外部イベントフィルタを転送先とするロールオーバーイベントのみに関係します。

00：カウンタが 0 または HRTIM_PERxR の値に等しい時にイベントが生成されます。

01：カウンタが 0 に等しい時にイベントが生成されます。

10：カウンタが HRTIM_PERxR に等しい時にイベントが生成されます。

11：予約済みです。

注： この設定は、UDM ビットがセットされたときにのみ適用されます。そうでない場合は、意味を持ちません。

注： このビットフィールドは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **UDM**：アップダウン･モード

このビットは、カウンタがアップカウントモードで動作するか、またはアップダウンカウントモードで動作するかを定義します。

0：カウンタはアップカウント･モードで動作します。

1：カウンタはアップダウン･カウント･モードで動作します。

注： このビットは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DCDR**：デュアル･チャネル DAC リセット･トリガ

このビットは、hrtim_dac_reset_trgx トリガが生成されるタイミングを定義します。

0：トリガは、カウンタのリセットまたはロールオーバー･イベントによって生成されます。

1：トリガは、出力 1 のセット･イベントによって生成されます。

注： DCDR ビットは、DCDE ビットがリセットされている（デュアルチャネル DAC トリガが無効の）ときは、意味がありません。

ビット 1 **DCDS** デュアルチャネル DAC ステップトリガ

このビットは、hrtim_dac_step_trgx トリガが生成されるタイミングを定義します。

0：トリガは、比較 2 のイベントによって生成されます。

1：トリガは、出力 1 のリセット･イベントによって生成されます。

注： DCDR ビットは、DCDE ビットがリセットされている（デュアルチャネル DAC トリガが無効の）ときは、意味がありません。

ビット 0 **DCDE**：デュアル･チャネル DAC トリガ･イネーブル

このビットはデュアルチャネル DAC トリガメカニズムを有効にします。

0：デュアル･チャネル DAC トリガは無効です。

1：デュアル･チャネル DAC トリガは有効です。

注： このビットは、タイマが動作（TxEN ビットのセット）された後は、変更できません。

27.5.50 HRTIM タイマ x 外部イベントフィルタリングレジスタ 3（HRTIM_TIMxEEFR3）（x = A～F）

アドレス･オフセット：ブロック A：0x0F0

アドレス･オフセット：ブロック B：0x170

アドレス･オフセット：ブロック C：0x1F0

アドレス･オフセット：ブロック D：0x270

アドレス･オフセット：ブロック E：0x2F0

アドレス･オフセット：ブロック F：0x370

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	EEVACNT[5:0]						EEVASEL[3:0]				Res.	EEVA RSTM	EEVA CRES	EEVA CE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:8 **EEVACNT[5:0]** 外部イベント A カウンタ

このビットフィールドは、外部イベント A カウンタの閾値を選択します。イベントの数が（EEVACNT[5:0]+1）値に等しいとき、イベントは有効と見なされます。

ビット 7:4 **EEVASEL[3:0]**：外部イベント A 選択

このビットは、外部イベント A のソースを選択します。

0：外部イベント A のソースとして外部イベント 1 が使用されます。

1：外部イベント A のソースとして外部イベント 2 が使用されます。

.....

9：外部イベント A のソースとして外部イベント 10 が使用されます。

その他：予約済みです。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **EEVARSTM**：外部イベント A のリセットモード

このビットは、外部イベント x カウンタのリセットモードを選択します。

0：外部イベント･カウンタ A は、リセット／ロールオーバー･イベントごとにリセットされます。

1：最後のカウント周期中にイベントが発生しない場合にのみ、外部イベント･カウンタ A は、リセット／ロールオーバー･イベントごとにリセットされます。

ビット 1 **EEVACRES**：外部イベント A カウンタのリセット

このビットは、外部イベント A カウンタをリセットします。これは、ソフトウェアによってセットされ、ハードウェアによってリセットされます。

0：影響なし。

1：外部イベント･カウンタ A はリセットされます。

ビット 0 **EEVACE**：外部イベント A カウンタ･イネーブル

このビットは、外部イベントA カウンタを有効にします。

0：外部イベント A カウンタは無効です。

1：外部イベント A カウンタは有効です。

27.5.51 HRTIM 制御レジスタ 1（HRTIM_CR1）

アドレス･オフセット：0x380

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	AD4USRC [2:0]			AD3USRC [2:0]			AD2USRC [2:0]			AD1USRC [2:0]		
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TF UDIS	TE UDIS	TD UDIS	TC UDIS	TB UDIS	TA UDIS	MUDIS
									rw	rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:25 **AD4USRC[2:0]**：ADC トリガ 4 更新ソース
AD1USRC[2:0] の説明を参照してください。

ビット 24:22 **AD3USRC[2:0]**：ADC トリガ 3 更新ソース
AD1USRC[2:0] の説明を参照してください。

ビット 21:19 **AD2USRC[2:0]**：ADC トリガ 2 更新ソース
AD1USRC[2:0] の説明を参照してください。

ビット 18:16 **AD1USRC[2:0]**：ADC トリガ 1 更新ソース
これらのビットは、HRTIM_ADC1R レジスタの更新（プリロードからアクティブレジスタへ転送）をトリガするソースを定義します。ソースタイマのみを定義します。正確な条件はタイマ自身の中（HRTIM_MCR または HRTIM_TIMxCR）で定義されます。
000 : マスタ･タイマ
001 : タイマ A
010 : タイマ B
011 : タイマ C
100 : タイマ D
101 : タイマ E
110 : タイマ F
111 : 予約済みです。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TFUDIS**：タイマ F 更新無効
TAUDIS の説明を参照してください。

ビット 5 **TEUDIS**：タイマ E 更新無効
TAUDIS の説明を参照してください

ビット 4 **TDUDIS**：タイマ D 更新無効
TAUDIS の説明を参照してください。

ビット 3 **TCUDIS**：タイマ C 更新無効
TAUDIS の説明を参照してください。

ビット 2 **TBUDIS**：タイマ B 更新無効
TAUDIS の説明を参照してください。

ビット 1 **TAUDIS**：タイマ A 更新無効

このビットは、タイマ A で一時的に更新イベントの生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新は有効です。更新は、選択されたソースの生成時に起きます。

1：更新は無効です。同時に考慮する必要のある複数のレジスタにソフトウェアが書き込めるように、更新は一時的に無効にされます。

ビット 0 **MUDIS**：マスタ更新無効

このビットは、一時的に更新イベントの生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：更新は有効です。

1：更新は無効です。同時に考慮する必要のある複数のレジスタにソフトウェアが書き込めるように、更新は一時的に無効にされます。

27.5.52 HRTIM 制御レジスタ 2（HRTIM_CR2）

アドレス･オフセット：0x384

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWPF	SWPE	SWPD	SWPC	SWPB	SWPA
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TFRST	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	TF SWU	TE SWU	TD SWU	TC SWU	TB SWU	TA SWU	MSWU
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **SWPF**：タイマ F 出力のスワップ

SWPA の説明を参照してください。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 20 **SWPE**：タイマ E 出力のスワップ

SWPA の説明を参照してください。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 19 **SWPD**：タイマ D 出力のスワップ

SWPA の説明を参照してください。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 18 **SWPC**：タイマ C 出力のスワップ

SWPA の説明を参照してください。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 17 **SWPB**：タイマ B 出力のスワップ

SWPA の説明を参照してください。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 16 **SWPA**：タイマ A 出力のスワップ

このビットによって、タイマ A 出力をスワップできます。

0 : HRTIM_SETA1R と HRTIM_RSTA1R は出力 A1 用のコードであり、HRTIM_SETA2R と HRTIM_RSTA2R は出力 A2 用のコードです。

1 : HRTIM_SETA1R と HRTIM_RSTA1R は出力 A2 用のコードであり、HRTIM_SETA2R と HRTIM_RSTA2R は出力 A1 用のコードです。

注： プッシュプルモードが有効な（PSHPLL = 1）ときは、このビットは意味がありません。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TFRST**：タイマ F カウンタのソフトウェア･リセット

TARST の説明を参照してください。

ビット 13 **TERST**：タイマ E カウンタのソフトウェア･リセット

TARST の説明を参照してください。

ビット 12 **TDRST**：タイマ D カウンタのソフトウェア･リセット

TARST の説明を参照してください。

ビット 11 **TCRST**：タイマ C カウンタのソフトウェア･リセット

TARST の説明を参照してください。

ビット 10 **TBRST**：タイマ B カウンタのソフトウェア･リセット

TARST の説明を参照してください。

ビット 9 **TARST**：タイマ A カウンタのソフトウェア･リセット

このビットをセットすると、タイマ A のカウンタがリセットされます。

このビットは、ハードウェアによって自動的にリセットされます。

ビット 8 **MRST**：マスタ･カウンタのソフトウェア･リセット

このビットをセットすると、マスタタイマのカウンタがリセットされます。

このビットは、ハードウェアによって自動的にリセットされます。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TFSWU**：タイマ F ソフトウェア更新

TASWU の説明を参照してください。

ビット 5 **TESWU**：タイマ E ソフトウェア更新

TASWU の説明を参照してください。

ビット 4 **TDSWU**：タイマ D ソフトウェア更新

TASWU の説明を参照してください。

ビット 3 **TCSWU**：タイマ C ソフトウェア更新

TASWU の説明を参照してください。

ビット 2 **TBSWU**：タイマ B ソフトウェア更新

TASWU の説明を参照してください。

ビット 1 **TASWU**：タイマ A ソフトウェア更新

このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。これはプリロードからアクティブレジスタへの転送を直ちに強制し、保留中の更新リクエストはキャンセルされます。

ビット 0 **MSWU**：マスタ･タイマ･ソフトウェア更新

このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。これはマスタタイマのプリロードからアクティブレジスタへの転送を直ちに強制し、保留中の更新リクエストはキャンセルされます。

27.5.53 HRTIM 割込みステータスレジスタ（HRTIM_ISR）

アドレスオフセット：0x388

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
									r		r	r	r	r	r

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPER**：バースト･モード周期割込みフラグ

このビットは、シングルショットバーストモード動作が完了した時、または連続モードでのバーストモード周期の終りに、ハードウェアによってセットされます。1 のとき、ソフトウェアで書き込むことによってクリアされます。

0：バースト･モード周期割込みは起きていません。

1：バースト･モード周期割込みが起きました。

ビット 16 **DLLRDY**：DLL レディ割込みフラグ

このビットは、DLL 較正が完了したときに、ハードウェアによってセットされます。1 のとき、ソフトウェアで書き込むことによってクリアされます。

0：DLL 較正レディ割込みは起きていません。

1：DLL 較正レディ割込みが起きました。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FLT6**：フォールト 6 割込みフラグ

FLT1 の説明を参照してください。

ビット 5 **SYSFLT**：システム･フォールト割込みフラグ

FLT1 の説明を参照してください。

ビット 4 **FLT5**：フォールト 5 割込みフラグ

FLT1 の説明を参照してください。

ビット 3 **FLT4**：フォールト 4 割込みフラグ

FLT1 の説明を参照してください。

ビット 2 **FLT3**：フォールト 3 割込みフラグ

FLT1 の説明を参照してください。

ビット 1 **FLT2**：フォールト 2 割込みフラグ

FLT1 の説明を参照してください。

ビット 0 **FLT1**：フォールト 1 割込みフラグ

このビットは、フォールト 1 イベントが発生したときに、ハードウェアによってセットされます。1 のとき、ソフトウェアで書き込むことによってクリアされます。

0：フォールト 1 割込みは起きていません。

1：フォールト 1 割込みが起きました。

27.5.54 HRTIM 割込みクリアスレジスタ（HRTIM_ICR）

アドレスオフセット：0x38C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	DLLRDYC
														w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6C	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
									w	w	w	w	w	w	w

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPERC**：バースト･モード周期フラグ･クリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの BMPER フラグがクリアされます。

ビット 16 **DLLRDYC**：DLL レディ割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの DLLRDY フラグがクリアされます。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FLT6C**：フォールト 6 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT6 フラグがクリアされます。

ビット 5 **SYSFLTC**：システムフォールト割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの SYSFLT フラグがクリアされます。

ビット 4 **FLT5C**：フォールト 5 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT5 フラグがクリアされます。

ビット 3 **FLT4C**：フォールト 4 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT4 フラグがクリアされます。

ビット 2 **FLT3C**：フォールト 3 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT3 フラグがクリアされます。

ビット 1 **FLT2C**：フォールト 2 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT2 フラグがクリアされます。

ビット 0 **FLT1C**：フォールト 1 割込みフラグクリア

このビットに 1 を書き込むと、HRTIM_ISR レジスタの FLT1 フラグがクリアされます。

27.5.55 HRTIM 割込み有効レジスタ（HRTIM_IER）

アドレスオフセット：0x390

リセット値：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6IE	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **BMPERIE**：バーストモード周期割込みイネーブル
このビットは、バーストモード周期割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：バースト･モード周期割込みは無効です。
1：バースト･モード周期割込みは有効です。

ビット 16 **DLLRDYIE**：DLL レディ割込みイネーブル
このビットは、DLL レディ割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：DLL レディ割込みは無効です。
1：DLL レディ割込みは有効です。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FLT6IE**：フォールト 6 割込みイネーブル
FLT1IE の説明を参照してください。

ビット 5 **SYSFLTIE**：システムフォールト割込みイネーブル
FLT1IE の説明を参照してください。

ビット 4 **FLT5IE**：フォールト 5 割込みイネーブル
FLT1IE の説明を参照してください。

ビット 3 **FLT4IE**：フォールト 4 割込みイネーブル
FLT1IE の説明を参照してください。

ビット 2 **FLT3IE**：フォールト 3 割込みイネーブル
FLT1IE の説明を参照してください。

ビット 1 **FLT2IE**：フォールト 2 割込みイネーブル
FLT1IE の説明を参照してください。

ビット 0 **FLT1IE**：フォールト 1 割込みイネーブル
このビットは、フォールト 1 割込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0：フォールト 1 割込みは無効です。
1：フォールト 1 割込みは有効です。

27.5.56 HRTIM 出力有効レジスタ（HRTIM_OENR）

アドレスオフセット：0x394

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TF2OEN	TF1OEN	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
				rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TF2OEN**：タイマ F の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 10 **TF1OEN**：タイマ F の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 9 **TE2OEN**：タイマ E の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 8 **TE1OEN**：タイマ E の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 7 **TD2OEN**：タイマ D の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 6 **TD1OEN**：タイマ D の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 5 **TC2OEN**：タイマ C の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 4 **TC1OEN**：タイマ C の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 3 **TB2OEN**：タイマ B の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 2 **TB1OEN**：タイマ B の出力 1 イネーブル
TA1OEN の説明を参照してください。

ビット 1 **TA2OEN**：タイマ A の出力 2 イネーブル
TA1OEN の説明を参照してください。

ビット 0 **TA1OEN**：タイマ A の出力 1 イネーブル
このビットをセットすると、タイマ A 出力 1 が有効になります。“0”を書き込んでも、ビットの値は変化しません。ビットを読み出すと、出力の有効／無効ステータスが返されます。
このビットは、タイマ関連のフォールト入力がアクティブになるとすぐに、ハードウェアによって非同期にクリアされます。
0：出力 A1 は無効です。出力はフォールトまたはアイドル状態です。
1：出力 A1 は有効です。

注： 無効ステータスはアイドルとフォールト状態の両方に対応します。出力無効ステータスは HRTIM_ODSR レジスタの TA1ODS ビットによって得られます。

27.5.57 HRTIM 出力無効レジスタ（HRTIM_ODISR）

アドレスオフセット：0x398

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TF2 ODIS	TF1 ODIS	TE2 ODIS	TE1 ODIS	TD2 ODIS	TD1 ODIS	TC2 ODIS	TC1 ODIS	TB2 ODIS	TB1 ODIS	TA2 ODIS	TA1 ODIS
				w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TF2ODIS**：タイマ F の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 10 **TF1ODIS**：タイマ F の出力 1 無効
TA1ODIS の説明を参照してください。

ビット 9 **TE2ODIS**：タイマ E の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 8 **TE1ODIS**：タイマ E の出力 1 無効
TA1ODIS の説明を参照してください。

ビット 7 **TD2ODIS**：タイマ D の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 6 **TD1ODIS**：タイマ D の出力 1 無効
TA1ODIS の説明を参照してください。

ビット 5 **TC2ODIS**：タイマ C の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 4 **TC1ODIS**：タイマ C の出力 1 無効
TA1ODIS の説明を参照してください。

ビット 3 **TB2ODIS**：タイマ B の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 2 **TB1ODIS**：タイマ B の出力 1 無効
TA1ODIS の説明を参照してください。

ビット 1 **TA2ODIS**：タイマ A の出力 2 無効
TA1ODIS の説明を参照してください。

ビット 0 **TA1ODIS**：タイマ A の出力 1 無効
このビットをセットすると、タイマ A 出力 1 が無効になります。出力は、実行状態またはフォールト状態から、アイドル状態に移行します。
“0”を書き込んでも、ビットの値は変化しません。

27.5.58 HRTIM 出力無効ステータスレジスタ（HRTIM_ODSR）

アドレスオフセット：0x39C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TF2ODS	TF1ODS	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **TF2ODS**：タイマ F の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 10 **TF1ODS**：タイマ F の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 9 **TE2ODS**：タイマ E の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 8 **TE1ODS**：タイマ E の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 7 **TD2ODS**：タイマ D の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 6 **TD1ODS**：タイマ D の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 5 **TC2ODS**：タイマ C の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 4 **TC1ODS**：タイマ C の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 3 **TB2ODS**：タイマ B の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 2 **TB1ODS**：タイマ B の出力 1 無効ステータス
TA1ODS の説明を参照してください。

ビット 1 **TA2ODS**：タイマ A の出力 2 無効ステータス
TA1ODS の説明を参照してください。

ビット 0 **TA1ODS**：タイマ A の出力 1 無効ステータス
ビットを読み出すと、出力の無効ステータスが返されます。出力がアクティブ（Tx1OEN または Tx2OEN = 1）のとき、これは意味がありません。
0：出力 A1 はアイドル状態で、無効です。
1：出力 A1 はフォールト状態で、無効です。

27.5.59 HRTIM バーストモード制御レジスタ（HRTIM_BMCR）

アドレスオフセット：0x3A0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFBM	TEBM	TDBM	TCBM	TBBM	TABM	MTBM
rc_w0									rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **BMSTAT**：バースト･モード･ステータス

このビットは、現在の動作状態を示します。

0：通常動作

1：バースト動作が進行中です。このビットに 0 を書き込むと、バーストモードが早期に終了します。

ビット 30:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **TFBM**：タイマ F バースト･モード

TABM の説明を参照してください。

ビット 21 **TEBM**：タイマ E バースト･モード

TABM の説明を参照してください。

ビット 20 **TDBM**：タイマ D バースト･モード

TABM の説明を参照してください。

ビット 19 **TCBM**：タイマ C バースト･モード

TABM の説明を参照してください。

ビット 18 **TBBM**：タイマ B バースト･モード

TABM の説明を参照してください。

ビット 17 **TABM**：タイマ A バースト･モード

このビットは、バーストモード動作中にタイマがどのように動作するかを指定します。このビットは、バーストモードが有効である間は、変更できません。

0：TA のカウンタ･クロックは維持され、タイマは通常動作します。

1：TA のカウンタ･クロックは停止され、カウンタはリセットされます。

注： このビットは、バランスアイドルモードがアクティブなときは（DLYPRT[2:0] = 0x11）、セットしてはいけません。

ビット 16 **MTBM**：マスタ･タイマ･バースト･モード

このビットは、バーストモード動作中にタイマがどのように動作するかを指定します。このビットは、バーストモードが有効である間は、変更できません。

0：マスタ･タイマのカウンタ･クロックは維持され、タイマは通常動作します。

1：マスタ･タイマのカウンタ･クロックは停止され、カウンタはリセットされます。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **BMPREN**：バースト･モード･プリロード･イネーブル

このビットは、レジスタプリロードメカニズムを有効にし、プリロード可能なレジスタ（HRTIM_BMCMPR、HRTIM_BMPER）への書込みアクセスがアクティブレジスタに行われるか、プリロードレジスタに行われるかを定義します。

0：プリロード無効：書込みアクセスは直接、アクティブ･レジスタに行われます。

1：プリロード有効：書込みアクセスはプリロード･レジスタに行われます。

ビット 9:6 **BMPRSC[3:0]**：バースト･モード･プリスケーラ

バーストモードコントローラ用の、f_{HRTIM} クロックのプリスケーリング比を定義します。このビットは、バーストモードが有効である間は、変更できません。

0000 : クロックは分周されません。
0001 : 分周比 2
0010 : 分周比 4
0011 : 分周比 8
0100 : 分周比 16
0101 : 分周比 32
0110 : 分周比 64
0111 : 分周比 128
1000 : 分周比 256
1001 : 分周比 512
1010 : 分周比 1024
1011 : 分周比 2048
1100 : 分周比 4096
1101 : 分周比 8192
1110 : 分周比 16384
1111 : 分周比 32768

ビット 5:2 **BMCLK[3:0]**：バースト･モード･クロック･ソース

このビットフィールドは、バーストモードカウンタ用のクロックソースを定義します。これは、バーストモードが有効である間は、変更できません（オンチップイベント 1～4 の接続の詳細は表 211 を参照）。

0000 : マスタ･タイマのカウンタ･リセット／ロールオーバー
0001 : タイマ A のカウンタ･リセット／ロールオーバー
0010 : タイマ B のカウンタ･リセット／ロールオーバー
0011 : タイマ C のカウンタ･リセット／ロールオーバー
0100 : タイマ D のカウンタ･リセット／ロールオーバー
0101 : タイマ E のカウンタ･リセット／ロールオーバー
0110 : オンチップ･イベント 1（hrtim_bm_ck1）をバースト･モード･カウンタ･クロック･ソースとして使用します。
0111 : オンチップ･イベント 2（hrtim_bm_ck2）をバースト･モード･カウンタ･クロック･ソースとして使用します。
1000 : オンチップ･イベント 3（hrtim_bm_ck3）をバースト･モード･カウンタ･クロック･ソースとして使用します。
1001 : オンチップ･イベント 4（hrtim_bm_ck4）をバースト･モード･カウンタ･クロック･ソースとして使用します。
1010 : プリスケールされた f_{HRTIM} クロック（BMPRSC[3:0] の設定による）
1011 : タイマ F のカウンタ･リセット／ロールオーバー
その他：予約済みです。

ビット 1 **BMOM**：バースト･モード動作モード

このビットは、バーストモードに一度だけ移行するのか、連続的に動作するのかを決定します。

0：シングルショット･モード
1：連続動作

ビット 0 **BME**：バースト･モード･イネーブル

このビットは、開始トリガを受信できる準備ができたバーストモードコントローラを開始します。
このビットに 0 を書き込むと、バーストモードが早期に終了します。

0：バースト･モードは無効です。
1：バースト･モードは有効です。

27.5.60 HRTIM バーストモードトリガレジスタ（HRTIM_BMTRGR）

アドレスオフセット：0x3A4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OCHPEV	EEV8	EEV7	TDEEV8	TAEEV7	TECMP2	TECMP1	TEREP	TFCMP1	TDCMP2	TFREP	TDREP	TDRST	TFRST	TCCMP1	TCREP
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **OCHPEV**：オンチップ·イベント

hrtim_bm_trg 入力（汎用タイマ TRGO 出力に接続されている）の立ち上がりエッジによって、バーストモードへの移行がトリガされます（詳細は表 208 を参照 ）。

ビット 30 **EEV8**：外部イベント 8（TIMD フィルタ適用）

TIMD フィルタによって調整された外部イベント 8 がバーストモード動作を開始します。

ビット 29 **EEV7**：外部イベント 7（TIMA フィルタ適用）

TIMA フィルタによって調整された外部イベント 7 がバーストモード動作を開始します。

ビット 28 **TDEEV8**：外部イベント 8 の後のタイマ D 周期

（TIMD フィルタによって調整された）外部イベント 8 の後のタイマ D 周期がバーストモード動作を開始します。

ビット 27 **TAEEV7**：外部イベント 7 の後のタイマ A 周期

（TIMA フィルタによって調整された）外部イベント 7 の後のタイマ A 周期がバーストモード動作を開始します。

ビット 26 **TECMP2**：タイマ E の比較 2 イベント

TACMP1 の説明を参照してください。

ビット 25 **TECMP1**：タイマ E の比較 1 イベント

TACMP1 の説明を参照してください。

ビット 24 **TEREP**：タイマ E 繰り返し

TAREP の説明を参照してください。

ビット 23 **TFCMP1**：タイマ F の比較 1 イベント

TACMP1 の説明を参照してください。

ビット 22 **TDCMP2**：タイマ D の比較 2 イベント

TACMP1 の説明を参照してください。

ビット 21 **TFREP**：タイマ F 繰り返し

TAREP の説明を参照してください。

ビット 20 **TDREP**：タイマ D 繰り返し

TAREP の説明を参照してください。

ビット 19 **TDRST**：タイマ D リセットまたはロールオーバー

TARST の説明を参照してください。

ビット 18 **TFRST**：タイマ F のリセット

TARST の説明を参照してください。

ビット 17 **TCCMP1**：タイマ C の比較 1 イベント

TACMP1 の説明を参照してください。

ビット 16 **TCREP**：タイマ C 繰り返し
TAREP の説明を参照してください。

ビット 15 **TCRST**：タイマ C リセットまたはロールオーバー
TARST の説明を参照してください。

ビット 14 **TBCMP2**：タイマ B の比較 2 イベント
TACMP1 の説明を参照してください。

ビット 13 **TBCMP1**：タイマ B の比較 1 イベント
TACMP1 の説明を参照してください。

ビット 12 **TBREP**：タイマ B 繰り返し
TAREP の説明を参照してください。

ビット 11 **TBRST**：タイマ B リセットまたはロールオーバー
TARST の説明を参照してください。

ビット 10 **TACMP2**：タイマ A の比較 2 イベント
TACMP1 の説明を参照してください。

ビット 9 **TACMP1**：タイマ A の比較 1 イベント
タイマ A の比較 1 イベントがバーストモード動作を開始します。

ビット 8 **TAREP**：タイマ A 繰り返し
タイマ A の繰り返しイベントがバーストモード動作を開始します。

ビット 7 **TARST**：タイマ A リセットまたはロールオーバー
タイマ A のリセットまたはロールオーバーがバーストモード動作を開始します。

ビット 6 **MSTCMP4**：マスタ比較 4
MSTCMP1 の説明を参照してください。

ビット 5 **MSTCMP3**：マスタ比較 3
MSTCMP1 の説明を参照してください。

ビット 4 **MSTCMP2**：マスタ比較 2
MSTCMP1 の説明を参照してください。

ビット 3 **MSTCMP1**：マスタ比較 1
マスタタイマの比較 1 イベントがバーストモード動作を開始します。

ビット 2 **MSTREP**：マスタ繰り返し
マスタタイマの繰り返しイベントがバーストモード動作を開始します。

ビット 1 **MSTRST**：マスタのリセットまたはロールオーバー
マスタタイマのリセットおよびロールオーバーイベントがバーストモード動作を開始します。

ビット 0 **SW**：ソフトウェア開始
このビットは、ソフトウェアによってセットされ、ハードウェアによって自動的にリセットされます。
セットされると、直ちにバーストモード動作を開始します。
このビットは、バーストモードが有効でない場合は（BME ビットがリセット）、アクティブではありません。

27.5.61 HRTIM バーストモード比較レジスタ（HRTIM_BMCMPR）

アドレスオフセット：0x3A8

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BMCMP[15:0]**：バースト･モード比較値

選択されたタイマがアイドル状態にある周期の数を定義します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

注： バーストモードクロックのソースとしてプリスケーラなしの f_{HRTIM} クロックを使用している場合（BMCLK[3:0] = 1010 および BMPRESC[3:0] = 0000）は、BMCMP[15:0] は 0x0000 に設定することはできません。

27.5.62 HRTIM バーストモード周期レジスタ（HRTIM_BMPER）

アドレスオフセット：0x3AC

リセット値：0x0000 0000

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

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BMPER[15:0]**：バースト･モード周期

バーストモード繰り返し周期を定義します。

このレジスタは、プリロードレジスタの内容か、またはプリロードが無効の場合はアクティブレジスタの内容を保持します。

注： BMPER[15:0] は、バーストモードが有効であるときは、null ではいけません。

27.5.63 HRTIM タイマ外部イベント制御レジスタ 1（HRTIM_EECR1）

アドレス･オフセット：0x3B0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1]
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE3SNS[0]	EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **EE5FAST**：外部イベント 5 高速モード
EE1FAST の説明を参照してください。

ビット 28:27 **EE5SNS[1:0]**：外部イベント 5 感度
EE1SNS[1:0] の説明を参照してください。

ビット 26 **EE5POL**：外部イベント 5 極性
EE1POL の説明を参照してください。

ビット 25:24 **EE5SRC[1:0]**：外部イベント 5 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 23 **EE4FAST**：外部イベント 4 高速モード
EE1FAST の説明を参照してください。

ビット 22:21 **EE4SNS[1:0]**：外部イベント 4 感度
EE1SNS[1:0] の説明を参照してください。

ビット 20 **EE4POL**：外部イベント 4 極性
EE1POL の説明を参照してください。

ビット 19:18 **EE4SRC[1:0]**：外部イベント 4 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 17 **EE3FAST**：外部イベント 3 高速モード
EE1FAST の説明を参照してください。

ビット 16:15 **EE3SNS[1:0]**：外部イベント 3 感度
EE1SNS[1:0] の説明を参照してください。

ビット 14 **EE3POL**：外部イベント 3 極性
EE1POL の説明を参照してください。

ビット 13:12 **EE3SRC[1:0]**：外部イベント 3 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 11 **EE2FAST**：外部イベント 2 高速モード
EE1FAST の説明を参照してください。

ビット 10:9 **EE2SNS[1:0]**：外部イベント 2 感度
EE1SNS[1:0] の説明を参照してください。

ビット 8 **EE2POL**：外部イベント 2 極性
EE1POL の説明を参照してください。

ビット 7:6 **EE2SRC[1:0]**：外部イベント 2 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 5 **EE1FAST**：外部イベント 1 高速モード
0：外部イベント 1 は、出力に作用する前に HRTIM のロジックによって再同期され、f_{HRTIM} クロックに関連する遅延時間が追加されます。
1：外部イベント 1 は非同期で出力に作用します（低遅延時間モード）
注： このビットは、イベントが使用されるカウンタが有効化（TxCEN ビットのセット）された後には変更してはなりません。

ビット 4:3 **EE1SNS[1:0]**：外部イベント 1 感度
00：EE1POL ビットで定義されたアクティブ･レベルを検出
01：EE1POL ビット値が何であれ、立ち上がりエッジを検出
10：EE1POL ビット値が何であれ、立ち下がりエッジを検出
11：EE1POL ビット値が何であれ、両エッジを検出

ビット 2 **EE1POL**：外部イベント 1 極性
このビットは、EE1SNS[1:0] = 00 の場合にのみ、意味があります。
0：外部イベントはアクティブ･ハイです。
1：外部イベントはアクティブ･ローです。
注： このパラメータは、タイマ x が有効化された後は、変更できません。EE1FAST ビットをセットする前に設定する必要があります。

ビット 1:0 **EE1SRC[1:0]**：外部イベント 1 ソース
このビットフィールドは、外部イベント 1 のソースを選択します。詳細については、表 209 を参照してください。
00：hrtim_eev1_1
01：hrtim_eev1_2
10：hrtim_eev1_3
11：hrtim_eev1_4
注： このパラメータは、タイマ x が有効化された後は、変更できません。EE1FAST ビットをセットする前に設定する必要があります。

27.5.64 HRTIM タイマ外部イベント制御レジスタ 2（HRTIM_EECR2）

アドレス･オフセット：0x3B4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EE10SNS [1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS [1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS[1]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8SNS[0]	EE8POL	EE8SRC[1:0]		Res.	EE7SNS [1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS [1:0]		EE6POL	EE6SRC[1:0]	
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:27 **EE10SNS[1:0]**：外部イベント 10 感度
EE1SNS[1:0] の説明を参照してください。

ビット 26 **EE10POL**：外部イベント 10 極性
EE1POL の説明を参照してください。

ビット 25:24 **EE10SRC[1:0]**：外部イベント 10 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:21 **EE9SNS[1:0]**：外部イベント 9 感度
EE1SNS[1:0] の説明を参照してください。

ビット 20 **EE9POL**：外部イベント 9 極性
EE1POL の説明を参照してください。

ビット 19:18 **EE9SRC[1:0]**：外部イベント 9 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:15 **EE8SNS[1:0]**：外部イベント 8 感度
EE1SNS[1:0] の説明を参照してください。

ビット 14 **EE8POL**：外部イベント 8 極性
EE1POL の説明を参照してください。

ビット 13:12 **EE8SRC[1:0]**：外部イベント 8 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:9 **EE7SNS[1:0]**：外部イベント 7 感度
EE1SNS[1:0] の説明を参照してください。

ビット 8 **EE7POL**：外部イベント 7 極性
EE1POL の説明を参照してください。

ビット 7:6 **EE7SRC[1:0]**：外部イベント 7 ソース
EE1SRC[1:0] の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **EE6SNS[1:0]**：外部イベント 6 感度
EE1SNS[1:0] の説明を参照してください。

ビット 2 **EE6POL**：外部イベント 6 極性
EE1POL の説明を参照してください。

ビット 1:0 **EE6SRC[1:0]**：外部イベント 6 ソース
EE1SRC[1:0] の説明を参照してください。

27.5.65 HRTIM タイマ外部イベント制御レジスタ 3（HRTIM_EECR3）

アドレス･オフセット：0x3B8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EEVSD[1:0]		Res.	Res.	EE10F [3:0]				Res.	Res.	EE9F [3:0]				Res.	Res.
rw	rw			rw	rw	rw	rw			rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EE8F [3:0]				Res.	Res.	EE7F [3:0]				Res.	Res.	EE6F [3:0]			
rw	rw	rw	rw			rw	rw	rw	rw			rw	rw	rw	rw

ビット 31:30 **EEVSD[1:0]**：外部イベント･サンプリング･クロック分周比
このビットフィールドは、タイマクロック周波数（f_{HRTIM}）と、デジタルフィルタによって使用される外部イベント信号サンプリングクロック（f_{EEVS}）との間の分周比を示します。
00：$f_{EEVS}=f_{HRTIM}$
01：$f_{EEVS}=f_{HRTIM}$ / 2
10：$f_{EEVS}=f_{HRTIM}$ / 4
11：$f_{EEVS}=f_{HRTIM}$ / 8

ビット 29:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **EE10F[3:0]**：外部イベント 10 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:18 **EE9F[3:0]**：外部イベント 9 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 17:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **EE8F[3:0]**：外部イベント 8 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:6 **EE7F[3:0]**：外部イベント 7 フィルタ
EE6F[3:0] の説明を参照してください。

ビット 4:5 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **EE6F[3:0]**：外部イベント 6 フィルタ

このビットフィールドは、外部イベント 6 入力をサンプリングするのに使用される周波数と、EEV6 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはカウンタでできていて、出力に有効な変化をもたらすには N 個の有効なサンプルが必要です。

0000：フィルタ無効
0001：$f_{SAMPLING}= f_{HRTIM}$、N = 2
0010：$f_{SAMPLING}= f_{HRTIM}$、N = 4
0011：$f_{SAMPLING}= f_{HRTIM}$、N = 8
0100：$f_{SAMPLING}= f_{EEVS}/2$、N=6
0101：$f_{SAMPLING}= f_{EEVS}/2$、N=8
0110：$f_{SAMPLING}= f_{EEVS}/4$、N=6
0111：$f_{SAMPLING}= f_{EEVS}/4$、N=8
1000：$f_{SAMPLING}= f_{EEVS}/8$、N=6
1001：$f_{SAMPLING}= f_{EEVS}/8$、N=8
1010：$f_{SAMPLING}= f_{EEVS}/16$、N=5
1011：$f_{SAMPLING}= f_{EEVS}/16$、N=6
1100：$f_{SAMPLING}= f_{EEVS}/16$、N=8
1101：$f_{SAMPLING}= f_{EEVS}/32$、N=5
1110：$f_{SAMPLING}= f_{EEVS}/32$、N=6
1111：$f_{SAMPLING}= f_{EEVS}/32$、N=8

27.5.66 HRTIM ADC トリガ 1 レジスタ（HRTIM_ADC1R）

アドレス･オフセット：0x3BC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC1 TEPER	ADC1 TEC4	ADC1 TEC3	ADC1 TFRST	ADC1 TDPER	ADC1 TDC4	ADC1 TDC3	ADC1 TFPER	ADC1 TCPER	ADC1 TCC4	ADC1 TCC3	ADC1 TFC4	ADC1 TBRST	ADC1 TBPER	ADC1 TBC4	ADC1 TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC1 TFC3	ADC1 TARST	ADC1 TAPER	ADC1 TAC4	ADC1 TAC3	ADC1 TFC2	ADC1 EEV5	ADC1 EEV4	ADC1 EEV3	ADC1 EEV2	ADC1 EEV1	ADC1 MPER	ADC1 MC4	ADC1 MC3	ADC1 MC2	ADC1 MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_ADC3R ビットの説明を参照してください。

27.5.67 HRTIM ADC トリガ 2 レジスタ（HRTIM_ADC2R）

アドレス･オフセット：0x3C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC2 TERST	ADC2 TEC4	ADC2 TEC3	ADC2 TEC2	ADC2 TDRST	ADC2 TDPER	ADC2 TDC4	ADC2 TFPER	ADC2T DC2	ADC2T CRST	ADC2T CPER	ADC2T CC4	ADC2 TFC4	ADC2T CC2	ADC2T BPER	ADC2T BC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC2 TFC3	ADC2 TBC2	ADC2 TAPER	ADC2 TAC4	ADC2 TFC2	ADC2 TAC2	ADC2 EEV10	ADC2 EEV9	ADC2 EEV8	ADC2 EEV7	ADC2 EEV6	ADC2 MPER	ADC2 MC4	ADC2 MC3	ADC2 MC2	ADC2 MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 HRTIM_ADC4R ビットの説明を参照してください。

27.5.68 HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R）

アドレス･オフセット：0x3C4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC3 TEPER	ADC3 TEC4	ADC3 TEC3	ADC3 TFRST	ADC3 TDPER	ADC3 TDC4	ADC3 TDC3	ADC3 TFPER	ADC3 TCPER	ADC3 TCC4	ADC3 TCC3	ADC3 TFC4	ADC3 TBRST	ADC3 TBPER	ADC3 TBC4	ADC3 TBC3
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3 TFC3	ADC3 TARST	ADC3 TAPER	ADC3 TAC4	ADC3 TAC3	ADC3 TFC2	ADC3 EEV5	ADC3 EEV4	ADC3 EEV3	ADC3 EEV2	ADC3 EEV1	ADC3 MPER	ADC3 MC4	ADC3 MC3	ADC3 MC2	ADC3 MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ADC3TEPER**：タイマ E 周期による ADC トリガ 3
ADC3TAPER の説明を参照してください。

ビット 30 **ADC3TEC4**：タイマ E の比較 4 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 29 **ADC3TEC3**：タイマ E の比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 28 **ADC3TFRST**：タイマ F のリセットおよびカウンタロールオーバーによる ADC トリガ 3(1)
ADC3TARST の説明を参照してください。

ビット 27 **ADC3TDPER**：タイマ D 周期による ADC トリガ 3
ADC3TAPER の説明を参照してください。

ビット 26 **ADC3TDC4**：タイマ D の比較 4 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 25 **ADC3TDC3**：タイマ D の比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 24 **ADC3TFPER**：タイマ F 周期による ADC トリガ 3
ADC3TAPER の説明を参照してください。

ビット 23 **ADC3TCPER**：タイマ C 周期による ADC トリガ 3
ADC3TAPER の説明を参照してください。

ビット 22 **ADC3TCC4**：タイマ C の比較 4 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 21 **ADC3TCC3**：タイマ C の比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 20 **ADC3TFC4**：タイマ F の比較 4 による ADC トリガ 3
ADC3TAC2 の説明を参照してください。

ビット 19 **ADC3TBRST**：タイマ B のリセットおよびカウンタのロールオーバーによる A/Dコンバータ トリガ 3 (1)
このビットは、タイマ B のリセットおよびロールオーバー･イベント発生時にA/Dコンバータ・トリガ 1 出力にA/Dコンバータ・トリガを生成する機能を有効にします。

ビット 18 **ADC3TBPER**：タイマ B 周期による ADC トリガ 3
ADC3TAPER の説明を参照してください。

ビット 17 **ADC3TBC4**：タイマ B の比較 4 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 16 **ADC3TBC3**：タイマ B の比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 15 **ADC3TFC3**：タイマ F の比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 14 **ADC3TARST**：タイマ A のリセットおよびカウンタ･ロールオーバーによる A/Dコンバータ トリガ 3 (1)
このビットは、タイマ A のリセットおよびロールオーバー･イベント発生時にADC トリガ 1 出力に ADC トリガを生成する機能を有効にします。

ビット 13 **ADC3TAPER**：タイマ A 周期による ADC トリガ 3
このビットは、ADC トリガ 1 出力に、タイマ A 周期イベント時に ADC トリガを生成することを有効にします。

ビット 12 **ADC3TAC4**：タイマ A 比較 4 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 11 **ADC3TAC3**：タイマ A 比較 3 による ADC トリガ 3
ADC3TFC2 の説明を参照してください。

ビット 10 **ADC3TFC2**：タイマ F の比較 2 による ADC トリガ 3
このビットは、ADC トリガ 3 出力に、タイマ F 比較 2 イベント時に ADC トリガを生成することを有効にします。

ビット 9 **ADC3EEV5**：外部イベント 5 による ADC トリガ 3 (1)
ADC3EEV1 の説明を参照してください。

ビット 8 **ADC3EEV4**：外部イベント 4 による ADC トリガ 3 (1)
ADC3EEV1 の説明を参照してください。

ビット 7 **ADC3EEV3**：外部イベント 3 による ADC トリガ 3 (1)
ADC3EEV1 の説明を参照してください。

ビット 6 **ADC3EEV2**：外部イベント 2 による ADC トリガ 3 (1)
ADC3EEV1 の説明を参照してください。

ビット 5 **ADC3EEV1**：外部イベント 1 による ADC トリガ 3 (1)
このビットは、ADC トリガ 3 出力に、外部イベント 1 の時に ADC トリガの生成を有効にします。

ビット 4 **ADC3MPER**：マスタ周期による ADC トリガ 3
このビットは、ADC トリガ 3 出力に、マスタタイマ周期イベント時に ADC トリガの生成を有効にします。

ビット 3 **ADC3MC4**：マスタの比較 4 による ADC トリガ 3
ADC3MC1 の説明を参照してください。

ビット 2 **ADC3MC3**：マスタの比較 3 による ADC トリガ 3
ADC3MC1 の説明を参照してください。

ビット 1 **ADC3MC2**：マスタの比較 2 による ADC トリガ 3
ADC3MC1 の説明を参照してください。

ビット 0 **ADC3MC1**：マスタの比較 1 による ADC トリガ 3
このビットは、ADC トリガ 3 出力に、マスタ比較 1 イベント時に ADC トリガの生成を有効にします。

1. これらのトリガは、HRTIM_ADC2R/HRTIM_ADC4R と HRTIM_ADC1R/HRTIM_ADC3R とで異なります。

27.5.69 HRTIM ADC トリガ 4 レジスタ（HRTIM_ADC4R）

アドレス･オフセット：0x3C8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADC4 TERST	ADC4 TEC4	ADC4 TEC3	ADC4 TEC2	ADC4 TDRST	ADC4 TDPER	ADC4 TDC4	ADC4 TFPER	ADC4T DC2	ADC4T CRST	ADC4T CPER	ADC4T CC4	ADC4 TFC4	ADC4T CC2	ADC4T BPER	ADC4T BC4
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC4 TFC3	ADC4 TBC2	ADC4 TAPER	ADC4 TAC4	ADC4 TFC2	ADC4 TAC2	ADC4 EEV10	ADC4 EEV9	ADC4 EEV8	ADC4 EEV7	ADC4 EEV6	ADC4 MPER	ADC4 MC4	ADC4 MC3	ADC4 MC2	ADC4 MC1
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **ADC4TERST**：タイマ E のリセットおよびカウンタロールオーバーによる ADC トリガ 4(1)
ADC4TCRST の説明を参照してください。

ビット 30 **ADC4TEC4**：タイマ E の比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 29 **ADC4TEC3**：タイマ E の比較 3 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 28 **ADC4TEC2**：タイマ E の比較 2 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 27 **ADC4TDRST**：タイマ D のリセットおよびカウンタロールオーバーによる ADC トリガ 4 (1)
ADC4TCRST の説明を参照してください。

ビット 26 **ADC4TDPER**：タイマ D 周期による ADC トリガ 4
ADC4TAPER の説明を参照してください。

ビット 25 **ADC4TDC4**：タイマ D の比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 24 **ADC4TFPER**：タイマ F 周期による ADC トリガ 4
ADC4TAPER の説明を参照してください。

ビット 23 **ADC4TDC2**：タイマ D の比較 2 による ADC トリガ 2
ADC4TAC2 の説明を参照してください。

ビット 22 **ADC4TCRST**：タイマ C のリセットおよびカウンタロールオーバーによる ADC トリガ 4 (1)
このビットは、ADC トリガ 1 出力に、タイマ C のリセットおよびロールオーバーイベント時に ADC トリガを生成することを有効にします。

ビット 21 **ADC4TCPER**：タイマ C 周期による ADC トリガ 4
ADC4TAPER の説明を参照してください。

ビット 20 **ADC4TCC4**：タイマ C の比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 19 **ADC4TFC4**：タイマ F の比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 18 **ADC4TCC2**：タイマ C の比較 2 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 17 **ADC4TBPER**：タイマ B 周期による ADC トリガ 4
ADC4TAPER の説明を参照してください。

ビット 16 **ADC4TBC4**：タイマ B の比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 15 **ADC4TFC3**：タイマ F の比較 3 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 14 **ADC4TBC2**：タイマ B の比較 2 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 13 **ADC4TAPER**：タイマ A 周期による ADC トリガ 4
このビットは、ADC トリガ 2 出力に、タイマ A 周期イベント時に ADC トリガを生成することを有効にします。

ビット 12 **ADC4TAC4**：タイマ A 比較 4 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 11 **ADC4TFC2**：タイマ F の比較 2 による ADC トリガ 4
ADC4TAC2 の説明を参照してください。

ビット 10 **ADC4TAC2**：タイマ A 比較 2 による ADC トリガ 4
このビットは、ADC トリガ 2 出力に、タイマ A 比較 2 の時に ADC トリガを生成することを有効にします。

ビット 9 **ADC4EEV10**：外部イベント 10 による ADC トリガ 4 (1)
ADC4EEV6 の説明を参照してください。

ビット 8 **ADC4EEV9**：外部イベント 9 による ADC トリガ 4 (1)
ADC4EEV6 の説明を参照してください。

ビット 7 **ADC4EEV8**：外部イベント 8 による ADC トリガ 4 (1)
ADC4EEV6 の説明を参照してください。

ビット 6 **ADC4EEV7**：外部イベント 7 による ADC トリガ 4 (1)
ADC4EEV6 の説明を参照してください。

ビット 5 **ADC4EEV6**：外部イベント 6 による ADC トリガ 4 (1)
このビットは、ADC トリガ 2 出力に、外部イベント 6 の時に ADC トリガの生成を有効にします。

ビット 4 **ADC4MPER**：マスタ周期による ADC トリガ 4
このビットは、ADC トリガ 2 出力に、マスタ周期イベント時に ADC トリガの生成を有効にします。

ビット 3 **ADC4MC4**：マスタの比較 4 による ADC トリガ 4
ADC4MC1 の説明を参照してください。

ビット 2 **ADC4MC3**：マスタの比較 3 による ADC トリガ 4
ADC4MC1 の説明を参照してください。

ビット 1 **ADC4MC2**：マスタの比較 2 による ADC トリガ 4
ADC4MC1 の説明を参照してください。

ビット 0 **ADC4MC1**：マスタの比較 1 による ADC トリガ 4
このビットは、ADC トリガ 2 出力に、マスタ比較 1 イベント時に ADC トリガの生成を有効にします。

1. これらのトリガは、HRTIM_ADC1R/HRTIM_ADC3R と HRTIM_ADC2R/HRTIM_ADC4R とで異なります。

27.5.70 HRTIM DLL 制御レジスタ（HRTIM_DLLCR）

アドレスオフセット：0x3CC

リセット値：0x0000 0000

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

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **CALRTE[1:0]**：DLL 較正頻度

これは DLL 較正の周期を定義します。

00：1048576 * t_{HRTIM} （f_{HRTIM} = 170 MHz の場合 6.168 ms）

01：131072 * t_{HRTIM} （f_{HRTIM} = 170 MHz の場合 771 µs）

10：16384 * t_{HRTIM} （f_{HRTIM} = 170 MHz の場合 96 µs）

11：2048 * t_{HRTIM} （f_{HRTIM} = 170 MHz の場合 12 µs）

ビット 1 **CALEN**：DLL 較正イネーブル

このビットは、CALRTE[1:0] ビットの設定に従って、周期的 DLL 較正を有効にします。CALEN ビットがリセットされているとき、較正は CAL ビットによってシングルショットで開始できます。

0：周期的較正は無効です。

1：CALRTE[1:0] 設定に従って、較正は周期的に実行されます。

注： CALEN は CAL ビットと同時にセットしてはなりません。

ビット 0 **CAL**：DLL 較正開始

このビットは DLL 較正プロセスを開始します。これは書込み専用です。

0：較正リクエストなし

1：較正開始

注： CAL は CALEN ビットと同時にセットしてはなりません。

27.5.71 HRTIM フォールト入力レジスタ 1（HRTIM_FLTINR1）

アドレス･オフセット：0x3D0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLT4LCK	FLT4F[3:0]				FLT4SRC[0]	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC[0]	FLT3P	FLT3E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT2LCK	FLT2F[3:0]				FLT2SRC[0]	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC[0]	FLT1P	FLT1E
rwo	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **FLT4LCK**：フォールト 4 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 30:27 **FLT4F[3:0]**：フォールト 4 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 26 FLT4SRC[0]：フォールト 4 ソースビット 0
HRTIM_FLTINR2 レジスタの FLT5SRC[0] の説明を参照してください。

ビット 25 **FLT4P**：フォールト 4 極性
HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 24 **FLT4E**：フォールト 4 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 23 **FLT3LCK**：フォールト 3 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 22:19 **FLT3F[3:0]**：フォールト 3 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 18 FLT3SRC[0]：フォールト 3 ソースビット 0
HRTIM_FLTINR2 レジスタの FLT5SRC[0] の説明を参照してください。

ビット 17 **FLT3P**：フォールト 3 極性
HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 16 **FLT3E**：フォールト 3 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 15 **FLT2LCK**：フォールト 2 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 14:11 **FLT2F[3:0]**：フォールト 2 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 10 FLT2SRC[0]：フォールト 2 ソースビット 0
HRTIM_FLTINR2 レジスタの FLT5SRC[0] の説明を参照してください。

ビット 9 **FLT2P**：フォールト 2 極性
HRTIM_FLTINR2 レジスタの FLT2P の説明を参照してください。

ビット 8 **FLT2E**：フォールト 2 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

ビット 7 **FLT1LCK**：フォールト 1 ロック
HRTIM_FLTINR2 レジスタの FLT5LCK の説明を参照してください。

ビット 6:3 **FLT1F[3:0]**：フォールト 1 フィルタ
HRTIM_FLTINR2 レジスタの FLT5F[3:0] の説明を参照してください。

ビット 2 **FLT1SRC[0]**：フォールト 1 ソースビット 0
HRTIM_FLTINR2 レジスタの FLT5SRC[0] の説明を参照してください。

ビット 1 **FLT1P**：フォールト 1 極性
HRTIM_FLTINR2 レジスタの FLT5P の説明を参照してください。

ビット 0 **FLT1E**：フォールト 1 イネーブル
HRTIM_FLTINR2 レジスタの FLT5E の説明を参照してください。

27.5.72 HRTIM フォールト入力レジスタ 2（HRTIM_FLTINR2）

アドレス･オフセット：0x3D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	FLT6 SRC[1]	FLT5 SRC[1]	FLT4 SRC[1]	FLT3 SRC[1]	FLT2 SRC[1]	FLT1 SRC[1]
						rw	rw			rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT6 LCK	FLT6F[3:0]				FLT6 SRC[0]	FLT6P	FLT6E	FLT5 LCK	FLT5F[3:0]				FLT5 SRC[0]	FLT5P	FLT5E
rwo	rw	rw	rw	rw	rw	rw	rw	rwo	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:24 **FLTSD[1:0]**：フォールト･サンプリング･クロック分周比
このビットフィールドは、タイマクロック周波数（f_{HRTIM}）と、デジタルフィルタによって使用されるフォールト信号サンプリングクロック（f_{FLTS}）との間の分周比を示します。
00：$f_{FLTS}=f_{HRTIM}$
01：$f_{FLTS}=f_{HRTIM}$ / 2
10：$f_{FLTS}=f_{HRTIM}$ / 4
11：$f_{FLTS}=f_{HRTIM}$ / 8
注： このビットフィールドは、どの FLTxE イネーブルビットの前にも書き込む必要があります。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 21 **FLT6SRC[1]**：フォールト 6 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 20 **FLT5SRC[1]**：フォールト 5 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 19 **FLT4SRC[1]**：フォールト 4 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 18 **FLT3SRC[1]**：フォールト 3 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 17 **FLT2SRC[1]**：フォールト 2 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 16 **FLT1SRC[1]**：フォールト 1 ソースビット 1
FLT5SRC[0] の説明を参照してください。

ビット 15 **FLT6LCK**：フォールト 6 ロック
FLT5LCK の説明を参照してください。

ビット 14:11 **FLT6F[3:0]**：フォールト 6 フィルタ
FLT5F[3:0] の説明を参照してください。

ビット 10 **FLT6SRC[0]**：フォールト 6 ソースビット 0
FLT5SRC[0] の説明を参照してください。

ビット 9 **FLT6P**：フォールト 6 極性
FLT5P の説明を参照してください。

ビット 8 **FLT6E**：フォールト 6 イネーブル
FLT5E の説明を参照してください。

ビット 7 **FLT5LCK**：フォールト 5 ロック
誤った書込みアクセスからフォールトプログラミングビットを保護するように、FLT5LCK ビットでフォールトプログラミングビットの書込み属性を変更します。
このビットは 1 回のみ書込み可能です。いったんセットされると、次のシステムリセットまで変更できません。
0：FLT5E、FLT5P、FLT5SRC[1:0]、FLT5F[3:0]、および FLT5BLK ビットは読出し／書込み用です。
1：FLT5E、FLT5P、FLT5SRC[1:0]、FLT5F[3:0]、および FLT5BLK ビットは、書込みできません（読出し専用モード）。

ビット 6:3 **FLT5F[3:0]**：フォールト 5 フィルタ
このビットフィールドは、FLT5 入力をサンプルする周波数と、FLT5 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。
0000 : フィルタなし、FLT5 は非同期として動作します。
0001：$f_{SAMPLING} = f_{HRTIM}$、N = 2
0010：$f_{SAMPLING} = f_{HRTIM}$、N = 4
0011：$f_{SAMPLING} = f_{HRTIM}$、N = 8
0100：$f_{SAMPLING} = f_{FLTS}/2$、N = 6
0101：$f_{SAMPLING} = f_{FLTS}/2$、N = 8
0110：$f_{SAMPLING} = f_{FLTS}/4$、N = 6
0111：$f_{SAMPLING} = f_{FLTS}/4$、N = 8
1000：$f_{SAMPLING} = f_{FLTS}/8$、N = 6
1001：$f_{SAMPLING} = f_{FLTS}/8$、N = 8
1010：$f_{SAMPLING} = f_{FLTS}/16$、N = 5
1011：$f_{SAMPLING} = f_{FLTS}/16$、N = 6
1100：$f_{SAMPLING} = f_{FLTS}/16$、N = 8
1101：$f_{SAMPLING} = f_{FLTS}/32$、N = 5
1110：$f_{SAMPLING} = f_{FLTS}/32$、N = 6
1111：$f_{SAMPLING} = f_{FLTS}/32$、N = 8

注： このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。
このビットフィールドは、FLT5LOCK がプログラムされているとき、変更されます。

ビット 2 **FLT5SRC[0]**：フォールト 5 ソースビット 0
FTL5SRC[1:0] ビットフィールドは、FAULT5 入力ソースを選択します（接続の詳細は、表 234 を参照）。
00：フォールト 5 入力は HRTIM_FLT5 入力ピンです。
01：フォールト 5 入力は COMPx 出力に接続されています。
10：フォールト 5 入力は EEV5_muxout 入力ピンです。
01：予約済みです。

注： このビットフィールドに書き込めるのは、FLT5E イネーブル･ビットがリセットされているときだけです。

ビット 1 **FLT5P**：フォールト 5 極性

このビットは、FAULT5 入力の極性を選択します。

0：フォールト 5 入力はアクティブ･ローです。

1：フォールト 5 入力はアクティブ･ハイです。

注： このビットフィールドに書き込めるのは、FLT5E イネーブル･ビットがリセットされているときだけです。

ビット 0 **FLT5E**：フォールト 5 イネーブル

このビットは、グローバル FAULT5 入力回路を有効にします。

0：フォールト 5 入力は無効です。

1：フォールト 5 入力は有効です。

27.5.73 HRTIM バースト DMA マスタタイマ更新レジスタ（HRTIM_BDMUPR）

アドレスオフセット：0x3D8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **MCMP4**：MCMP4R レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 8 **MCMP3**：MCMP3R レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 7 **MCMP2**：MCMP2R レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 6 **MCMP1**：MCMP1R レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 5 **MREP**：MREP レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 4 **MPER**：MPER レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 3 **MCNT**：MCNTR レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 2 **MDIER**：MDIER レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 1 **MICR**：MICR レジスタ更新イネーブル

MCR の説明を参照してください。

ビット 0 **MCR**：MCR レジスタ更新イネーブル

このビットは、マスタタイマ MCR レジスタが、バースト DMA によって更新されるレジスタのリストの一部であるかどうかを定義します。

0：MCR レジスタはバースト DMA アクセスによって更新されません。

1：MCR レジスタはバースト DMA アクセスによって更新されます。

27.5.74 HRTIM バースト DMA タイマ x 更新レジスタ（HRTIM_BDTxUPR）（x = A～F）

アドレスオフセット：ブロック A：0x3DC

アドレス･オフセット：ブロック B：0x3E0

アドレス･オフセット：ブロック C：0x3E4

アドレス･オフセット：ブロック D：0x3E8

アドレス･オフセット：ブロック E：0x3EC

アドレス･オフセット：ブロック F：0x3F4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMx EEFR3	TIMx CR2	TIMxFL TR	TIMxO UTR	TIMxC HPR	TIMxR STR	TIMxE EFR2
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMxE EFR1	TIMxR ST2R	TIMxS ET2R	TIMxR ST1R	TIMxS ET1R	TIMxD TxR	TIMxC MP4	TIMxC MP3	TIMxC MP2	TIMxC MP1	TIMxR EP	TIMxP ER	TIMxC NT	TIMxDI ER	TIMxIC R	TIMxC R
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **TIMxEEFR3**：HRTIM_EEFxR3 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 21 **TIMxCR2**：HRTIM_TIMxCR2 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 20 **TIMxFLTR**：HRTIM_FLTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 19 **TIMxOUTR**：HRTIM_OUTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 18 **TIMxCHPR**：HRTIM_CHPxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 17 **TIMxRSTR**：HRTIM_RSTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 16 **TIMxEEFR2**：HRTIM_EEFxR2 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 15 **TIMxEEFR1**：HRTIM_EEFxR1 レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 14 **TIMxRST2R**：HRTIM_RST2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 13 **TIMxSET2R**：HRTIM_SET2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 12 **TIMxRST1R**：HRTIM_RST1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 11 **TIMxSET1R**：HRTIM_SET1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 10 **TIMxDTR**：HRTIM_DTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 9 **TIMxCMP4**：HRTIM_CMP4xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 8 **TIMxCMP3**：HRTIM_CMP3xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 7 **TIMxCMP2**：HRTIM_CMP2xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 6 **TIMxCMP1**：HRTIM_CMP1xR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 5 **TIMxREP**：HRTIM_REPxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 4 **TIMxPER**：HRTIM_PERxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 3 **TIMxCNT**：HRTIM_CNTxR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 2 **TIMxDIER**：HRTIM_TIMxDIER レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 1 **TIMxICR**：HRTIM_TIMxICR レジスタ更新イネーブル
TIMxCR の説明を参照してください。

ビット 0 **TIMxCR**：HRTIM_TIMxCR レジスタ更新イネーブル
このビットは、マスタタイマ MCR レジスタが、バースト DMA によって更新されるレジスタのリストの一部であるかどうかを定義します。
0：HRTIM_TIMxCR レジスタはバースト DMA アクセスによって更新されません。
1：HRTIM_TIMxCR レジスタはバースト DMA アクセスによって更新されます。

27.5.75 HRTIM バースト DMA データレジスタ（HRTIM_BDMADR）

アドレスオフセット：0x3F0

リセット値：0x0000 0000

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

ビット 31:0 **BDMADR[31:0]**：バースト DMA データ･レジスタ
このレジスタへ書込みアクセスすると、以下のトリガがかかります。
– BDTxUPR および BDMUPR レジスタのビットで有効化されたレジスタにデータ値をコピー
– 次の書込み位置にレジスタポインタをインクリメント

27.5.76 HRTIM ADC 拡張トリガ･レジスタ（HRTIM_ADCER）

アドレスオフセット：0x3F8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	ADC10TRG [4:0]					ADC9TRG [4:0]					ADC8TRG [4:0]				
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ADC7TRG [4:0]					ADC6TRG [4:0]					ADC5TRG [4:0]				
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:26 **ADC10TRG[4:0]**：ADC トリガ 10 選択
このビットは、ADC トリガ 10 のソースを選択します。
ADC6TRG[4:0] の説明を参照してください。

ビット 25:21 **ADC9TRG[4:0]**：ADC トリガ 9 選択
このビットは、ADC トリガ 9 のソースを選択します。
ADC5TRG[4:0] の説明を参照してください。

ビット 20:16 **ADC8TRG[4:0]**：ADC トリガ 8 選択
このビットは、ADC トリガ 8 のソースを選択します。
ADC6TRG[4:0] の説明を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:10 **ADC7TRG[4:0]**：ADC トリガ 7 選択
このビットは、ADC トリガ 7 のソースを選択します。
ADC5TRG[4:0] の説明を参照してください。

ビット 9:5 **ADC6TRG[4:0]**：ADC トリガ 6 選択

このビットは、ADC トリガ 6 のソースを選択します。

0： マスタの比較 1 によるトリガ
1： マスタの比較 2 によるトリガ
2： マスタの比較 3 によるトリガ
3： マスタの比較 4 によるトリガ
4： マスタ周期によるトリガ
5： 外部イベント 6 によるトリガ
6： 外部イベント 7 によるトリガ
7： 外部イベント 8 によるトリガ
8： 外部イベント 9 によるトリガ
9： 外部イベント 10 によるトリガ
10：タイマ A 比較 2 によるトリガ
11：タイマ A 比較 4 によるトリガ
12：タイマ A 周期によるトリガ
13：タイマ B 比較 2 によるトリガ
14：タイマ B 比較 4 によるトリガ
15：タイマ B 周期によるトリガ
16：タイマ C 比較 2 によるトリガ
17：タイマ C 比較 4 によるトリガ
18：タイマ C 周期によるトリガ
19：タイマ C のリセットおよびカウンタ･ロールオーバーによるトリガ
20：タイマ D 比較 2 によるトリガ
21：タイマ D 比較 4 によるトリガ
22：タイマ D 周期によるトリガ
23：タイマ D のリセットおよびカウンタ･ロールオーバーによるトリガ
24：タイマ E 比較 2 によるトリガ
25：タイマ E 比較 3 によるトリガ
26：タイマ E 比較 4 によるトリガ
27：タイマ E のリセットおよびカウンタ･ロールオーバーによるトリガ
28：タイマ F 比較 2 によるトリガ
29：タイマ F 比較 3 によるトリガ
30：タイマ F 比較 4 によるトリガ
31：タイマ F 周期によるトリガ

ビット 4:0 **ADC5TRG[4:0]**：ADC トリガ 5 選択

このビットは、ADC トリガ 5 のソースを選択します。

0： マスタの比較 1 によるトリガ
1： マスタの比較 2 によるトリガ
2： マスタの比較 3 によるトリガ
3： マスタの比較 4 によるトリガ
4： マスタ周期によるトリガ
5： 外部イベント 1 によるトリガ
6： 外部イベント 2 によるトリガ
7： 外部イベント 3 によるトリガ
8： 外部イベント 4 によるトリガ
9： 外部イベント 5 によるトリガ
10：タイマ A 比較 3 によるトリガ
11：タイマ A 比較 4 によるトリガ
12：タイマ A 周期によるトリガ
13：タイマ A のリセットおよびカウンタ･ロールオーバーによるトリガ
14：タイマ B 比較 3 によるトリガ
15：タイマ B 比較 4 によるトリガ
16：タイマ B 周期によるトリガ
17：タイマ B のリセットおよびカウンタ･ロールオーバーによるトリガ
18：タイマ C 比較 3 によるトリガ
19：タイマ C 比較 4 によるトリガ
20：タイマ C 周期によるトリガ
21：タイマ D 比較 3 によるトリガ
22：タイマ D 比較 4 によるトリガ
23：タイマ D 周期によるトリガ
24：タイマ E 比較 3 によるトリガ
25：タイマ E 比較 4 によるトリガ
26：タイマ E 周期によるトリガ
27：タイマ F 比較 2 によるトリガ
28：タイマ F 比較 3 によるトリガ
29：タイマ F 比較 4 によるトリガ
30：タイマ F 周期によるトリガ
31：タイマ F のリセットおよびカウンタ･ロールオーバーによるトリガ

27.5.77 HRTIM ADC トリガ更新レジスタ（HRTIM_ADCUR）

アドレスオフセット：0x3FC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AD10USRC [2:0]			Res.	AD9USRC [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AD8USRC [2:0]			Res.	AD7USRC [2:0]			Res.	AD6USRC [2:0]			Res.	AD5USRC [2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **AD10USRC[2:0]**：ADC トリガ 10 更新ソース
AD5USRC[2:0] の説明を参照してください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **AD9USRC[2:0]**：ADC トリガ 9 更新ソース
AD5USRC[2:0] の説明を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **AD8USRC[2:0]**：ADC トリガ 8 更新ソース
AD5USRC[2:0] の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **AD7USRC[2:0]**：ADC トリガ 7 更新ソース
AD5USRC[2:0] の説明を参照してください。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **AD6USRC[2:0]**：ADC トリガ 6 更新ソース
AD5USRC[2:0] の説明を参照してください。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **AD5USRC[2:0]**：ADC トリガ 5 更新ソース
これらのビットは、HRTIM_ADCER レジスタの ADC5TRG[4:0] ビットフィールドの更新（プリロードからアクティブレジスタへ転送）をトリガするソースを定義します。ソースタイマのみを定義します。正確な条件はタイマ自身の中（HRTIM_MCR の BRSTDMA[1:0] ビットフィールドまたは HRTIM_TIMxCR レジスタの UPDGAT[3:0] ビットフィールド）で定義されます。
000 : マスタ･タイマ
001 : タイマ A
010 : タイマ B
011 : タイマ C
100 : タイマ D
101 : タイマ E
110 : タイマ F
111 : 予約済みです。

27.5.78 HRTIM ADC ポスト･スケーラ･レジスタ 1（HRTIM_ADCPS1）

アドレス･オフセット：0x400

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ADC5PSC [4:0]					Res.	ADC4PSC [4:0]					Res.	ADC3PSC[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC3PSC [3:0]				Res.	ADC2PSC [4:0]					Res.	ADC1PSC [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **ADC5PSC[4:0]**：ADC 5 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **ADC4PSC[4:0]**：ADC 4 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **ADC3PSC[4:0]**：ADC 3 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **ADC2PSC[4:0]**：ADC 2 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **ADC1PSC[4:0]**：ADC 1 ポストスケーラ
このビットは、ADC 1 ポストスケーラ比を選択します。

27.5.79 HRTIM ADC ポストスケーラレジスタ 2（HRTIM_ADCPS2）

アドレス･オフセット：0x404

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	ADC10PSC [4:0]					Res.	ADC9PSC [4:0]					Res.	ADC8PSC[4]
			rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADC8PSC [3:0]				Res.	ADC7PSC [4:0]					Res.	ADC6PSC [4:0]				
rw	rw	rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:24 **ADC10PSC[4:0]**：ADC 10 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:18 **ADC9PSC[4:0]**：ADC 9 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16:12 **ADC8PSC[4:0]**：ADC 8 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:6 **ADC7PSC[4:0]**：ADC 7 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **ADC6PSC[4:0]**：ADC 6 ポストスケーラ
ADC1PSC[4:0] の説明を参照してください。

27.5.80 HRTIM フォールト入力レジスタ 3（HRTIM_FLTINR3）

アドレス･オフセット：0x408

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
FLT4 RSTM	FLT4 CRES	FLT4CNT[3:0]				FLT4 BLKS	FLT4 BLKE	FLT3 RSTM	FLT3 CRES	FLT3CNT[3:0]				FLT3 BLKS	FLT3 BLKE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT2 RSTM	FLT2 CRES	FLT2CNT[3:0]				FLT2 BLKS	FLT2 BLKE	FLT1 RSTM	FLT1 CRES	FLT1CNT[3:0]				FLT1 BLKS	FLT1 BLKE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **FLT4RSTM**：フォールト 4 リセットモード
FLT5RSTM の説明を参照してください。

ビット 30 **FLT4CRES**：フォールト 4 のカウンタリセット
FLT5CRES の説明を参照してください。

ビット 29:26 **FLT4CNT[3:0]**：フォールト 4 カウンタ
FLT5CNT の説明を参照してください。

ビット 25 **FLT4BLKS**：フォールト 4 ブランキングソース
FLT5BLKS の説明を参照してください。

ビット 24 **FLT4BLKE**：フォールト 4 ブランキングイネーブル
FLT5BLKE の説明を参照してください。

ビット 23 **FLT3RSTM**：フォールト 3 リセットモード
FLT5RSTM の説明を参照してください。

ビット 22 **FLT3CRES**：フォールト 3 のカウンタリセット
FLT5CRES の説明を参照してください。

ビット 21:18 **FLT3CNT[3:0]**：フォールト 3 カウンタ
FLT5CNT の説明を参照してください。

ビット 17 **FLT3BLKS**：フォールト 3 ブランキングソース
FLT5BLKS の説明を参照してください。

ビット 16 **FLT3BLKE**：フォールト 3 ブランキングイネーブル
FLT5BLKE の説明を参照してください。

ビット 15 **FLT2RSTM**：フォールト 2 リセットモード
FLT5RSTM の説明を参照してください。

ビット 14 **FLT2CRES**：フォールト 2 のカウンタリセット
FLT5CRES の説明を参照してください。

ビット 13:10 **FLT2CNT[3:0]**：フォールト 2 カウンタ
FLT5CNT の説明を参照してください。

ビット 9 **FLT2BLKS**：フォールト 2 ブランキングソース
FLT5BLKS の説明を参照してください。

ビット 8 **FLT2BLKE**：フォールト 2 ブランキングイネーブル
FLT5BLKE の説明を参照してください。

ビット 7 **FLT1RSTM**：フォールト 1 リセットモード
FLT5RSTM の説明を参照してください。

ビット 6 **FLT1CRES**：フォールト 1 のカウンタリセット
FLT5CRES の説明を参照してください。

ビット 5:2 **FLT1CNT[3:0]**：フォールト 1 カウンタ
FLT5CNT の説明を参照してください。

ビット 1 **FLT1BLKS**：フォールト 1 ブランキングソース
FLT5BLKS の説明を参照してください。

ビット 0 **FLT1BLKE**：フォールト 1 ブランキングイネーブル
FLT5BLKE の説明を参照してください。

27.5.81 HRTIM フォールト入力レジスタ 4（HRTIM_FLTINR4）

アドレス･オフセット：0x40C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLT6 RSTM	FLT6 CRES	FLT6CNT[3:0]				FLT6 BLKS	FLT6 BLKE	FLT5 RSTM	FLT5 CRES	FLT5CNT[3:0]				FLT5 BLKS	FLT5 BLKE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **FLT6RSTM**：フォールト 6 リセットモード

FLT5RSTM の説明を参照してください。

ビット 14 **FLT6CRES**：フォールト 6 のカウンタリセット

FLT5CRES の説明を参照してください。

ビット 13:10 **FLT6CNT[3:0]**：フォールト 6 カウンタ

FLT5CNT の説明を参照してください。

ビット 9 **FLT6BLKS**：フォールト 6 ブランキングソース

FLT5BLKS の説明を参照してください。

ビット 8 **FLT6BLKE**：フォールト 6 ブランキングイネーブル

FLT5BLKE の説明を参照してください。

ビット 7 **FLT5RSTM**：フォールト 5 リセットモード

このビットは FAULT5 カウンタのリセットモードを選択します。

0 :FAULT 5 カウンタは、リセット／ロールオーバー･イベントごとにリセットされます。

1 :最後のカウント周期中にフォールトが発生しなかった場合にのみ、フォールト 5 カウンタは、リセット／ロールオーバー･イベントごとにリセットされます。

注：このビットフィールドに書き込めるのは、FLT5E イネーブル･ビットがリセットされているときだけです。

ビット 6 **FLT5CRES**：フォールト 5 のカウンタリセット

このビットは、FAULT5 カウンタをリセットします。これは、ソフトウェアによってセットされ、ハードウェアによってリセットされます。

0：影響なし。

1：フォールト 5 のカウンタはリセットされます。

ビット 5:2 **FLT5CNT[3:0]**：フォールト 5 カウンタ

このビットフィールドは、FAULT5 カウンタの閾値を選択します。イベントの数が（FLT5CNT[3:0]+1）値に等しいとき、フォールトは有効と見なされます。

ビット 1 **FLT5BLKS**：フォールト 5 ブランキングソース

FTL5BLKS ビットは、FAULT5 ブランキングソースを選択します（詳細は表 236 を参照）。

0：フォールト 5 リセットにアラインされたブランキング･ウィンドウ

1：フォールト 5 移動ブランキング･ウィンドウ

注： このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。

ビット 0 **FLT5BLKE**：フォールト 5 ブランキングイネーブル

FTL5BLKE ビットは、FAULT5 ブランキングモードを選択します。ブランキングソースは、FLT5BLKS ビットによって定義されます。

0：フォールト 5 にはブランキングはありません。

1：フォールト 5 ブランキング･モード

注： このビットフィールドに書き込めるのは、FLT5E イネーブルビットがリセットされているときだけです。

27.5.82 HRTIM レジスタマップ

下表に、HRTIM レジスタの配置の一覧を示します。

表 242. HRTIM レジスタマップとリセット値 - マスタタイマ

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**HRTIM_MCR**	BRSTDMA[1:0]		MREPU	Res.	PREEN	DACSYNC[1:0]		Res.	Res.	TFCEN	TECEN	TDCEN	TCCEN	TBCEN	TACEN	MCEN	SYNCSRC[1:0]		SYNCOUT[1:0]		SYNCSTRTM	SYNCRSTM	SYNCIN[1:0]		INTLVD[1:0]		HALF	RETRIG	CONT	CKPSC[2:0]		
	リセット値	0	0	0	-	0	0	0	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	**HRTIM_MISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPD	SYNC	MREP	MCMP4	MCMP3	MCMP2	MCMP1
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x008	**HRTIM_MICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDC	SYNCC	MREPC	MCMP4C	MCMP3C	MCMP2C	MCMP1C
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x00C	**HRTIM_MDIER**(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDDE	SYNCDE	MREPDE	MCMP4DE	MCMP3DE	MCMP2DE	MCMP1DE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MUPDIE	SYNCIE	MREPIE	MCMP4IE	MCMP3IE	MCMP2IE	MCMP1IE
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x010	**HRTIM_MCNTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCNT[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	**HRTIM_MPER** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MPER[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0x018	**HRTIM_MREP** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MREP[7:0]							
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0
0x01C	**HRTIM_MCMP1R** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP1 [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0x024	**HRTIM_MCMP2R** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP2 [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**HRTIM_MCMP3R** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP3 [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	**HRTIM_MCMP4R** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4 [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. このレジスタはプリロードできます（885 ページの表 227 参照）。

表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x080（x=A） 0x100（x=B） 0x180（x=C） 0x200（x=D） 0x280（x=E） 0x300（x=F）	**HRTIM_ TIMxCR**	UPDGAT [3:0]				PREEN	DACSYNC[1:0]		MSTU	TEU	TDU	TCU	TBU	TAU	TxRSTU	TxREPU	TFU	DELCMP4[1:0]		DELCMP2 [1:0]		SYNCSTRTx	SYNCRSTx	RSYNCU	INTLVD[1:0]		PSHPLL	HALF	RETRIG	CONT	CKPSCx[2:0]		
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x084（x=A） 0x104（x=B） 0x184（x=C） 0x204（x=D） 0x284（x=E） 0x304（x=F）	**HRTIM_ TIMxISR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	O2CPY	O1CPY	O2STAT	O1STAT	IPPSTAT	CPPSTAT	Res.	DLYPRT	RST	RSTx2	SETx2	RSTx1	SETx1	CPT2	CPT1	UPD	Res.	REP	CMP4	CMP3	CMP2	CMP1
	リセット値	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0
0x088（x=A） 0x108（x=B） 0x188（x=C） 0x208（x=D） 0x288（x=E） 0x308（x=F）	**HRTIM_ TIMxICR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DLYPRTC	RSTC	RSTx2C	SET2xC	RSTx1C	SET1xC	CPT2C	CPT1C	UPDC	Res.	REPC	CMP4C	CMP3C	CMP2C	CMP1C
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0
0x08C（x=A） 0x10C（x=B） 0x18C（x=C） 0x20C（x=D） 0x28C（x=E） 0x30C（x=F）	**HRTIM_ TIMxDIER**(1)	Res.	DLYPRTDE	RSTDE	RSTx2DE	SETx2DE	RSTx1DE	SET1xDE	CPT2DE	CPT1DE	UPDDE	Res.	REPDE	CMP4DE	CMP3DE	CMP2DE	CMP1DE	Res.	DLYPRTIE	RSTIE	RSTx2IE	SETx2IE	RSTx1IE	SET1xIE	CPT2IE	CPT1IE	UPDIE	Res.	REPIE	CMP4IE	CMP3IE	CMP2IE	CMP1IE
	リセット値	-	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0
0x090（x=A） 0x110（x=B） 0x190（x=C） 0x210（x=D） 0x290（x=E） 0x310（x=F）	**HRTIM_ CNTxR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNTx[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x094（x=A） 0x114（x=B） 0x194（x=C） 0x214（x=D） 0x294（x=E） 0x314（x=F）	**HRTIM_ PERxR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERx[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0x098（x=A） 0x118（x=B） 0x198（x=C） 0x218（x=D） 0x298（x=E） 0x318（x=F）	**HRTIM_ REPxR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]							
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0
0x09C（x=A） 0x11C（x=B） 0x19C（x=C） 0x21C（x=D） 0x29C（x=E） 0x31C（x=F）	**HRTIM_ CMP1xR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP1x [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A0（x=A） 0x120（x=B） 0x1A0（x=C） 0x220（x=D） 0x2A0（x=E） 0x320（x=F）	**HRTIM_ CMP1CxR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REPx[7:0]								CMP1x[15:0]															
	リセット値	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A4（x=A） 0x124（x=B） 0x1A4（x=C） 0x224（x=D） 0x2A4（x=E） 0x324（x=F）	**HRTIM_ CMP2xR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP2x [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0A8（x=A） 0x128（x=B） 0x1A8（x=C） 0x228（x=D） 0x2A8（x=E） 0x328（x=F）	**HRTIM_ CMP3xR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP3x [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0AC（x=A） 0x12C（x=B） 0x1AC（x=C） 0x22C（x=D） 0x2AC（x=E） 0x32C（x=F）	**HRTIM_ CMP4xR** (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP4x [15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B0（x=A） 0x130（x=B） 0x1B0（x=C） 0x230（x=D） 0x2B0（x=E） 0x330（x=F）	**HRTIM_CPT 1xR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR	CPT1x[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B4（x=A） 0x134（x=B） 0x1B4（x=C） 0x234（x=D） 0x2B4（x=E） 0x334（x=F）	**HRTIM_CPT 2xR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIR	CPT2x[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0B8（x=A） 0x138（x=B） 0x1B8（x=C） 0x238（x=D） 0x2B8（x=E） 0x338（x=F）	**HRTIM_DTx R** (1)	DTFLKx	DTFSLKx	Res.	Res.	Res.	Res.	SDTFx	DTFx[8:0]									DTRLKx	DTRSLKx	Res.	DTPRSC[2:0]			SDTRx	DTRx[8:0]								
	リセット値	0	0	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0BC（x=A） 0x13C（x=B） 0x1BC（x=C） 0x23C（x=D） 0x2BC（x=E） 0x33C（x=F）	**HRTIM_ SETx1R** (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C0（x=A） 0x140（x=B） 0x1C0（x=C） 0x240（x=D） 0x2C0（x=E） 0x340（x=F）	**HRTIM_ RSTx1R** (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C4（x=A） 0x144（x=B） 0x1C4（x=C） 0x244（x=D） 0x2C4（x=E） 0x344（x=F）	**HRTIM_ SETx2R** (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SST
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C8（x=A） 0x148（x=B） 0x1C8（x=C） 0x248（x=D） 0x2C8（x=E） 0x348（x=F）	**HRTIM_ RSTx2R** (1)	UPDATE	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	TIMEVNT9	TIMEVNT8	TIMEVNT7	TIMEVNT6	TIMEVNT5	TIMEVNT4	TIMEVNT3	TIMEVNT2	TIMEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP3	CMP2	CMP1	PER	RESYNC	SRT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0CC（x=A） 0x14C（x=B） 0x1CC（x=C） 0x24C（x=D） 0x2CC（x=E） 0x34C（x=F）	**HRTIM_ EEFxR1**	Res.	Res.	Res.	EE5FLTR [3:0]				EE5LTCH	Res.	EE4FLTR [3:0]				EE4LTCH	Res.	EE3FLTR [3:0]				EE3LTCH	Res.	EE2FLTR [3:0]				EE2LTCH	Res.	EE1FLTR [3:0]				EE1LTCH
	リセット値	-	-	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0
0x0D0（x=A） 0x150（x=B） 0x1D0（x=C） 0x250（x=D） 0x2D0（x=E） 0x350（x=F）	**HRTIM_ EEFxR2**	Res.	Res.	Res.	EE10FLTR [3:0]				EE10LTCH	Res.	EE9FLTR [3:0]				EE9LTCH	Res.	EE8FLTR [3:0]				EE8LTCH	Res.	EE7FLTR [3:0]				EE7LTCH	Res.	EE6FLTR [3:0]				EE6LTCH
	リセット値	-	-	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0

表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0D4	HRTIM_ RSTAR (1)	TIMFCMP2	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMFCMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x154	HRTIM_ RSTBR (1)	TIMFCMP2	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMFCMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1D4	HRTIM_ RSTCR (1)	TIMFCMP2	TIMECMP4	TIMECMP2	TIMECMP1	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMFCMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x254	HRTIM_ RSTDR (1)	TIMFCMP2	TIMECMP4	TIMECMP2	TIMECMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMFCMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2D4	HRTIM_ RSTER (1)	TIMFCMP2	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMFCMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x354	HRTIM_ RSTFR (1)	TIMECMP2	TIMDCMP4	TIMDCMP2	TIMDCMP1	TIMCCMP4	TIMCCMP2	TIMCCMP1	TIMBCMP4	TIMBCMP2	TIMBCMP1	TIMACMP4	TIMACMP2	TIMACMP1	EXTEVNT10	EXTEVNT9	EXTEVNT8	EXTEVNT7	EXTEVNT6	EXTEVNT5	EXTEVNT4	EXTEVNT3	EXTEVNT2	EXTEVNT1	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTPER	CMP4	CMP2	UPDT	TIMECMP1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0D8（x=A） 0x158（x=B） 0x1D8（x=C） 0x258（x=D） 0x2D8（x=E） 0x358（x=F）	HRTIM_ CHPxR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	STRTPW [3:0]				CARDT Y[2:0]			CARFRQ [3:0]			
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0
0x0DC	HRTIM_ CPT1ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x15C	HRTIM_ CPT1BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1DC	HRTIM_ CPT1CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x25C	HRTIM_ CPT1DCR	TECMP2	TECMP1	TE1RST	TE1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2DC	HRTIM_ CPT1ECR	TFCMP2	TFCMP1	TF1RST	TF1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x35C	HRTIM_ CPT1FCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0E0	HRTIM_ CPT2ACR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x160	HRTIM_ CPT2BCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1E0	HRTIM_ CPT2CCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x260	HRTIM_ CPT2DCR	TECMP2	TECMP1	TE1RST	TE1SET	TFCMP2	TFCMP1	TF1RST	TF1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2E0	HRTIM_ CPT2ECR	TFCMP2	TFCMP1	TF1RST	TF1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x360	HRTIM_ CPT2FCR	TECMP2	TECMP1	TE1RST	TE1SET	TDCMP2	TDCMP1	TD1RST	TD1SET	TCCMP2	TCCMP1	TC1RST	TC1SET	TBCMP2	TBCMP1	TB1RST	TB1SET	TACMP2	TACMP1	TA1RST	TA1SET	EXEV10CPT	EXEV9CPT	EXEV8CPT	EXEV7CPT	EXEV6CPT	EXEV5CPT	EXEV4CPT	EXEV3CPT	EXEV2CPT	EXEV1CPT	UPDCPT	SWCPT
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 243. HRTIM レジスタ･マップとリセット値 - TIMx（x = A ~ F）（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0E4（x=A） 0x164（x=B） 0x1E4（x=C） 0x264（x=D） 0x2E4（x=E） 0x364（x=F）	**HRTIM_ OUTxR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DIDL2	CHP2	FAULT2[1:0]		IDLES2	IDLEM2	POL2	Res.	Res.	Res.	Res.	DLYPRT[2:0]			DLYPRTEN	DTEN	DIDL1	CHP1	FAULT1[1:0]		IDLES1	IDLEM1	POL1	Res.
	リセット値	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	-
0x0E8（x=A） 0x168（x=B） 0x1E8（x=C） 0x268（x=D） 0x2E8（x=E） 0x368（x=F）	**HRTIM_ FLTxR**	FLTLCK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6EN	FLT5EN	FLT4EN	FLT3EN	FLT2EN	FLT1EN
	リセット値	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0
0x0EC（x=A） 0x16C（x=B） 0x1EC（x=C） 0x26C（x=D） 0x2EC（x=E） 0x36C（x=F）	**HRTIM_ TIMxCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRGHLF	Res.	Res.	GTCMP3	GTCMP1	FE ROM [1:0]		BM ROM [1:0]		AD ROM [1:0]		OUT ROM [1:0]		ROM [1:0]		Res.	UDM	Res.	DCDR	DCDS	DCDE
	リセット値	-	-	-	-	-	-	-	-	-	-	-	0	-	-	0	0	0	0	0	0	0	0	0	0	0	0	-	0	-	0	0	0
0x0F0（x=A） 0x170（x=B） 0x1F0（x=C） 0x270（x=D） 0x2F0（x=E） 0x370（x=F）	**HRTIM_ TIMxEEFR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EEVACNT[5:0]						EEVASEL [5:0]				Res.	EEVARSTM	EEVACRES	EEVACE
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	-	0	0	0

1. このレジスタはプリロードできます（885 ページの表 227 参照）。

表 244. HRTIM レジスタマップとリセット値 - 共通機能

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0380	HRTIM_CR1	Res.	Res.	Res.	Res.	AD4USRC [2:0]			AD3USRC [2:0]			AD2USRC [2:0]			AD1USRC [2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFUDIS	TEUDIS	TDUDIS	TCUDIS	TBUDIS	TAUDIS	MUDIS
	リセット値	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x0384	HRTIM_CR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SWPF	SWPE	SWPD	SWPC	SWPB	SWPA	Res.	TFRST	TERST	TDRST	TCRST	TBRST	TARST	MRST	Res.	TFSWU	TESWU	TDSWU	TCSWU	TBSWU	TASWU	MSWU
	リセット値	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	-	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0
0x388	HRTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER	DLLRDY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6	SYSFLT	FLT5	FLT4	FLT3	FLT2	FLT1
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x38C	HRTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERC	DLLRDYC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6C	SYSFLTC	FLT5C	FLT4C	FLT3C	FLT2C	FLT1C
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x390	HRTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPERIE	DLLRDYIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6IE	SYSFLTIE	FLT5IE	FLT4IE	FLT3IE	FLT2IE	FLT1IE
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0
0x394	HRTIM_OENR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TF2OEN	TF1OEN	TE2OEN	TE1OEN	TD2OEN	TD1OEN	TC2OEN	TC1OEN	TB2OEN	TB1OEN	TA2OEN	TA1OEN
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0
0x398	HRTIM_ODISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TF2ODIS	TF1ODIS	TE2ODIS	TE1ODIS	TD2ODIS	TD1ODIS	TC2ODIS	TC1ODIS	TB2ODIS	TB1ODIS	TA2ODIS	TA1ODIS
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0
0x39C	HRTIM_ODSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TF2ODS	TF1ODS	TE2ODS	TE1ODS	TD2ODS	TD1ODS	TC2ODS	TC1ODS	TB2ODS	TB1ODS	TA2ODS	TA1ODS
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0
0x3A0	HRTIM_BMCR	BMSTAT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFBM	TEBM	TDBM	TCBM	TBBM	TABM	MTBM	Res.	Res.	Res.	Res.	Res.	BMPREN	BMPRSC[3:0]				BMCLK[3:0]				BMOM	BME
	リセット値	0	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0
0x3A4	HRTIM_BMTRG	OCHPEV	EEV8	EEV7	TDEEV8	TAEEV7	TECMP2	TECMP1	TEREP	TFCMP1	TDCMP2	TFREP	TDREP	TDRST	TFRST	TCCMP1	TCREP	TCRST	TBCMP2	TBCMP1	TBREP	TBRST	TACMP2	TACMP1	TAREP	TARST	MSTCMP4	MSTCMP3	MSTCMP2	MSTCMP1	MSTREP	MSTRST	SW
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3A8	HRTIM_BMCMPR(1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMCMP[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 244. HRTIM レジスタマップとリセット値 - 共通機能 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3AC	HRTIM_ BMPER (1)	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BMPER[15:0]															
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3B0	HRTIM_EECR1	Res.	Res.	EE5FAST	EE5SNS[1:0]		EE5POL	EE5SRC[1:0]		EE4FAST	EE4SNS[1:0]		EE4POL	EE4SRC[1:0]		EE3FAST	EE3SNS[1:0]		EE3POL	EE3SRC[1:0]		EE2FAST	EE2SNS[1:0]		EE2POL	EE2SRC[1:0]		EE1FAST	EE1SNS[1:0]		EE1POL	EE1SRC[1:0]	
	リセット値	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3B4	HRTIM_EECR2	Res.	Res.	Res.	EE10SNS [1:0]		EE10POL	EE10SRC[1:0]		Res.	EE9SNS [1:0]		EE9POL	EE9SRC[1:0]		Res.	EE8SNS [1:0]		EE8POL	EE8SRC[1:0]		Res.	EE7SNS [1:0]		EE7POL	EE7SRC[1:0]		Res.	EE6SNS [1:0]		EE6POL	EE6SRC[1:0]	
	リセット値	-	-	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0
0x3B8	HRTIM_EECR3	EEVSD[1:0]		Res.	Res.	EE10F [3:0]				Res.	Res.	EE9F [3:0]				Res.	Res.	EE8F [3:0]				Res.	Res.	EE7F [3:0]				Res.	Res.	EE6F [3:0]			
	リセット値	0	0	-	-	0	0	0	0	-	-	0	0	0	0	-	-	0	0	0	0	-	-	0	0	0	0	-	-	0	0	0	0
0x3BC	HRTIM_ ADC1R (1)	ADC1TEPER	ADC1TEC4	ADC1TEC3	ADC1TFRST	ADC1TDPER	ADC1TDC4	ADC1TDC3	ADC1TFPER	ADC1TCPER	ADC1TCC4	ADC1TCC3	ADC1TFC4	ADC1TBRST	ADC1TBPER	ADC1TBC4	ADC1TBC3	ADC1TFC3	ADC1TARST	ADC1TAPER	ADC1TAC4	ADC1TAC3	ADC1TFC2	ADC1TAC2	ADC1EEV5	ADC1EEV4	ADC1EEV3	ADC1EEV2	ADC1EEV1	ADC1MPER	ADC1MC4	ADC1MC3	ADC1MC2	ADC1MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C0	HRTIM_ ADC2R (1)	ADC2TERST	ADC2TEC4	ADC2TEC3	ADC2TEC2	ADC2TDRST	ADC2TDPER	ADC2TDC4	ADC2TFPER	ADC2TDC2	ADC2TCRST	ADC2TCPER	ADC2TCC4	ADC2TFC4	ADC2TCC2	ADC2TBPER	ADC2TBC4	ADC2TFC3	ADC2TBC2	ADC2TAPER	ADC2TAC4	ADC2TFC2	ADC2TAC2	ADC2EEV10	ADC2EEV9	ADC2EEV8	ADC2EEV7	ADC2EEV6	ADC2MPER	ADC2MC4	ADC2MC3	ADC2MC2	ADC2MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C4	HRTIM_ ADC3R (1)	ADC3TEPER	ADC3TEC4	ADC3TEC3	ADC3TFRST	ADC3TDPER	ADC3TDC4	ADC3TDC3	ADC3TFPER	ADC3TCPER	ADC3TCC4	ADC3TCC3	ADC3TFC4	ADC3TBRST	ADC3TBPER	ADC3TBC4	ADC3TBC3	ADC3TFC3	ADC3TARST	ADC3TAPER	ADC3TAC4	ADC3TAC3	ADC3TFC2	ADC3TAC2	ADC3EEV5	ADC3EEV4	ADC3EEV3	ADC3EEV2	ADC3EEV1	ADC3MPER	ADC3MC4	ADC3MC3	ADC3MC2	ADC3MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3C8	HRTIM_ ADC4R (1)	ADC4TERST	ADC4TEC4	ADC4TEC3	ADC4TEC2	ADC4TDRST	ADC4TDPER	ADC4TDC4	ADC4TFPER	ADC4TDC2	ADC4TCRST	ADC4TCPER	ADC4TCC4	ADC4TFC4	ADC4TCC2	ADC4TBPER	ADC4TBC4	ADC4TFC3	ADC4TBC2	ADC4TAPER	ADC4TAC4	ADC4TFC2	ADC4TAC2	ADC4EEV10	ADC4EEV9	ADC4EEV8	ADC4EEV7	ADC4EEV6	ADC4MPER	ADC4MC4	ADC4MC3	ADC4MC2	ADC4MC1
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03CC	HRTIM_DLLCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALRTE [1:0]		CALEN	CAL
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0

表 244. HRTIM レジスタマップとリセット値 - 共通機能 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3D0	HRTIM_ FLTINR1	FLT4LCK	FLT4F[3:0]				FLT4SRC[0]	FLT4P	FLT4E	FLT3LCK	FLT3F[3:0]				FLT3SRC[0]	FLT3P	FLT3E	FLT2LCK	FLT2F[3:0]				FLT2SRC[0]	FLT2P	FLT2E	FLT1LCK	FLT1F[3:0]				FLT1SRC[0]	FLT1P	FLT1E
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3D4	HRTIM_ FLTINR2	Res.	Res.	Res.	Res.	Res.	Res.	FLTSD[1:0]		Res.	Res.	FLT6SRC[1]	FLT5SRC[1]	FLT4SRC[1]	FLT3SRC[1]	FLT2SRC[1]	FLT1SRC[1]	FLT6LCK	FLT6F[3:0]				FLT6SRC[0]	FLT6P	FLT6E	FLT5LCK	FLT5F[3:0]				FLT5SRC[0]	FLT5P	FLT5E
	リセット値	-	-	-	-	-	-	0	0	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3D8	HRTIM_ BDMUPDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCMP4	MCMP3	MCMP2	MCMP1	MREP	MPER	MCNT	MDIER	MICR	MCR
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0
0x3DC	HRTIM_ BDTAUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMAEEFR3	TIMACR2	TIMAFLTR	TIMAOUTR	TIMACHPR	TIMARSTR	TIMAEEFR2	TIMAEEFR1	TIMARST2R	TIMASET2R	TIMARST1R	TIMASET1R	TIMADTxR	TIMACMP4	TIMACMP3	TIMACMP2	TIMACMP1	TIMAREP	TIMAPER	TIMACNT	TIMADIER	TIMAICR	TIMACR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3E0	HRTIM_ BDTBUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMBEEFR3	TIMBCR2	TIMBFLTR	TIMBOUTR	TIMBCHPR	TIMBRSTR	TIMBEEFR2	TIMBEEFR1	TIMBRST2R	TIMBSET2R	TIMBRST1R	TIMBSET1R	TIMBDTxR	TIMBCMP4	TIMBCMP3	TIMBCMP2	TIMBCMP1	TIMBREP	TIMBPER	TIMBCNT	TIMBDIER	TIMBICR	TIMBCR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3E4	HRTIM_ BDTCUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMCEEFR3	TIMCCR2	TIMCFLTR	TIMCOUTR	TIMCCHPR	TIMCRSTR	TIMCEEFR2	TIMCEEFR1	TIMCRST2R	TIMCSET2R	TIMCRST1R	TIMCSET1R	TIMCDTxR	TIMCCMP4	TIMCCMP3	TIMCCMP2	TIMCCMP1	TIMCREP	TIMCPER	TIMCCNT	TIMCDIER	TIMCICR	TIMCCR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3E8	HRTIM_ BDTDUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMDEEFR3	TIMDCR2	TIMDFLTR	TIMDOUTR	TIMDCHPR	TIMDRSTR	TIMDEEFR2	TIMDEEFR1	TIMDRST2R	TIMDSET2R	TIMDRST1R	TIMDSET1R	TIMDDTxR	TIMDCMP4	TIMDCMP3	TIMDCMP2	TIMDCMP1	TIMDREP	TIMDPER	TIMDCNT	TIMDDIER	TIMDICR	TIMDCR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3EC	HRTIM_ BDTEUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMEEEFR3	TIMECR2	TIMEFLTR	TIMEOUTR	TIMECHPR	TIMERSTR	TIMEEEFR2	TIMEEEFR1	TIMERST2R	TIMESET2R	TIMERST1R	TIMESET1R	TIMEDTxR	TIMECMP4	TIMECMP3	TIMECMP2	TIMECMP1	TIMEREP	TIMEPER	TIMECNT	TIMEDIER	TIMEICR	TIMECR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3F0	HRTIM_ BDMADR	BDMADR[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3F4	HRTIM_ BDTFUPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIMFEEFR3	TIMFCR2	TIMFFLTR	TIMFOUTR	TIMFCHPR	TIMFRSTR	TIMFEEFR2	TIMFEEFR1	TIMFRST2R	TIMFSET2R	TIMFRST1R	TIMFSET1R	TIMFDTxR	TIMFCMP4	TIMFCMP3	TIMFCMP2	TIMFCMP1	TIMFREP	TIMFPER	TIMFCNT	TIMFDIER	TIMFICR	TIMFCR
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 244. HRTIM レジスタマップとリセット値 - 共通機能 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x3F8	**HRTIM_ ADCER** (1)	Res.	ADC10TRG [4:0]					ADC9TRG [4:0]					ADC8TRG [4:0]					Res.	ADC7TRG [4:0]					ADC6TRG [4:0]					ADC5TRG [4:0]				
	リセット値	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x3FC	**HRTIM_ ADCUR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AD10U SRC [2:0]			Res.	AD9U SRC [2:0]			Res.	AD8U SRC [2:0]			Res.	AD7U SRC [2:0]			Res.	AD6U SRC [2:0]		Res.	AD5U SRC [2:0]			
	リセット値	-	-	-	-	-	-	-	-	-	0	0	0	-	0	0	0	-	0	0	0	-	0	0	0	-	0	0	0	-	0	0	0
0x400	**HRTIM_ADCPS1**	Res.	Res.	Res.	ADC5PSC [4:0]			Res.	ADC4PSC [4:0]				Res.	ADC3PSC [4:0]				Res.	ADC2PSC [4:0]					Res.	ADC1PSC [4:0]								
	リセット値	-	-	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0
0x404	**HRTIM_ADCPS2**	Res.	Res.	Res.	ADC10PSC [4:0]			Res.	ADC9PSC [4:0]				Res.	ADC8PSC [4:0]				Res.	ADC7PSC [4:0]					Res.	ADC6PSC [4:0]								
	リセット値	-	-	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0	-	0	0	0	0	0
0x408	**HRTIM_ FLTINR3**	FLT4RSTM	FLT4CRES	FLT4CNT[3:0]				FLT4BLKS	FLT4BLKE	FLT3RSTM	FLT3CRES	FLT3CNT[3:0]				FLT3BLKS	FLT3BLKE	FLT2RSTM	FLT2CRES	FLT2CNT[3:0]				FLT2BLKS	FLT2BLKE	FLT1RSTM	FLT1CRES	FLT1CNT[3:0]				FLT1BLKS	FLT1BLKE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40C	**HRTIM_ FLTINR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLT6RSTM	FLT6CRES	FLT6CNT[3:0]				FLT6BLKS	FLT6BLKE	FLT5RSTM	FLT5CRES	FLT5CNT[3:0]				FLT5BLKS	FLT5BLKE
	リセット値	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

1. このレジスタはプリロードできます（885 ページの表 227 参照）。

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

28 高機能制御タイマ（TIM1/TIM8/TIM20）

28.1 TIM1/TIM8/TIM20 概要

高機能制御タイマ（TIM1/TIM8/TIM20）は、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロード･カウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

高機能制御タイマ（TIM1/TIM8/TIM20）と汎用タイマ（TIMy）は、互いに独立しており、リソースを共有しません。これらのタイマは、セクション 28.3.30：タイマの同期に示すように、相互に同期させることができます。

28.2 TIM1/TIM8/TIM20 の主な特徴

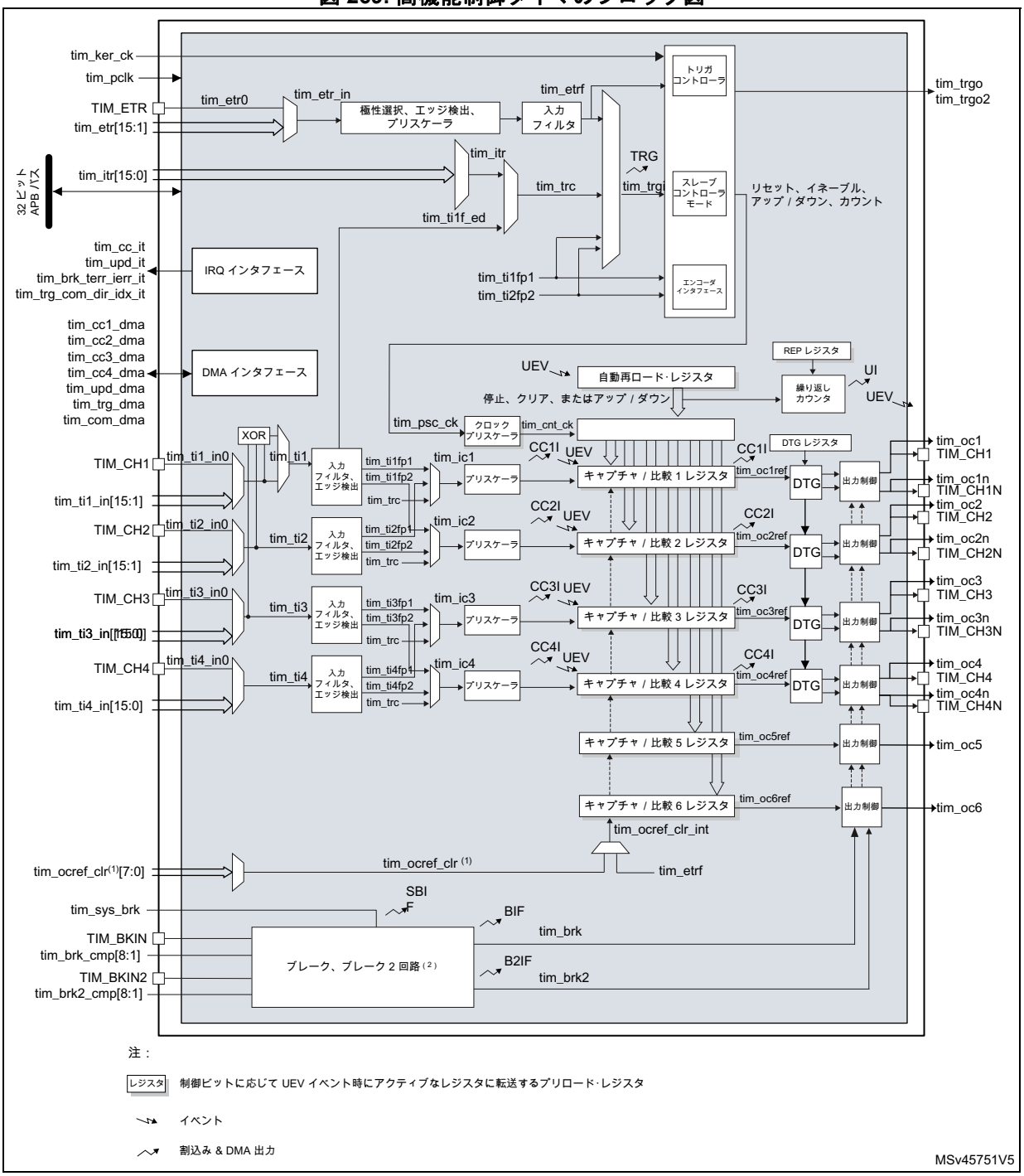
TIM1/TIM8/TIM20 タイマの主な機能は、次のとおりです。

- 16 ビットのアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- 16 ビットのプログラム可能なプリスケーラ。カウンタクロック周波数を 1 から 65536 の間で分周でき、分周比の動作中の変更も可能。
- 次の機能を持つ、最大 6 つの独立チャネル：
  - 入力キャプチャ（ただしチャネル 5 および 6）
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をユーザが選択可能な安全な設定にする 2 つのブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

28.3 TIM1/TIM8/TIM20 機能詳細

28.3.1 ブロック図

図 269. 高機能制御タイマのブロック図



1. この機能は一部のタイマでは利用できません（セクション 28.3.2 : TIM1/TIM8/TIM20 ピンおよび内部信号を参照）。
2. 詳細については、図 316 : ブレークおよびブレーク 2 回路の概要を参照してください。

28.3.2 TIM1/TIM8/TIM20 ピンおよび内部信号

このセクションの以下の表は、TIM の入力および出力の一覧です。

表 245. TIM の入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2 TIM_CH3 TIM_CH4	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、または PWM で使用できます。 TIM_CH1 と TIM_CH2 は、外部クロック（tim_ker_ck クロックの 1/4 未満）、外部トリガ、および直交エンコーダの入力としても使用できます。 TIM_CH1、TIM_CH2、および TIM_CH3 は、デジタルホール効果センサとのインタフェースに使用できます。
TIM_CH1N TIM_CH2N TIM_CH3N TIM_CH4N	出力	デッドタイム挿入の可能性がある TIM_CHx 出力から導き出されるタイマ相補出力。
TIM_ETR	入力	外部トリガ入力。この入力 は、外部トリガまたは外部クロックソースとして使用できます。tim_etr_in プリスケーラが使用されている場合、この入力は tim_ker_ck よりも高い周波数のクロックを受信できます。
TIM_BKIN TIM_BKIN2	入力／出力	ブレークおよびブレーク 2 入力。これらの入力は双方向モードにも設定することができます。

表 246. TIM 内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0] tim_ti3_in [15:0] tim_ti4_in [15:0]	入力	内部タイマ入力バス。tim_ti1_in[15:0] および tim_ti2_in[15:0] の入力は、キャプチャ用、外部クロック（tim_ker_ck クロックの 1/4 未満）、直交エンコーダ信号用として使用できます。
tim_etr[15:0]	入力	外部トリガ内部入力バス。これらの入力は、トリガ、外部クロックとして、またはハードウェアのサイクルごとのパルス幅制御に使用できます。tim_etr_in プリスケーラが使用されている場合、これらの入力は tim_ker_ck よりも高い周波数のクロックを受信できます。
tim_itr[15:0]	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ck クロックの 1/4 未満）として使用できます。
tim_trgo/tim_trgo2	出力	内部トリガ出力。これらのトリガは、他のタイマおよび／またはその他のペリフェラルによって使用されます。

表 246. TIM 内部入力／出力信号 （続き）

内部信号名	信号タイプ	説明
tim_ocref_clr[7:0]	入力	タイマの tim_ocref_clr 入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim ocxref 信号をクリアするために使用できます。
tim_brk_cmp[8:1]	入力	内部信号のブレーク入力
tim_brk2_cmp [8:1]	入力	内部信号のブレーク 2 入力
tim_sys_brk[n:0]	入力	システムブレーク入力この入力は MCU のシステムレベルのエラーを集めます。
tim_pclk	入力	タイマ APB クロック
tim_ker_ck	入力	タイマ･カーネル･クロック。
tim_cc_it	出力	タイマのキャプチャ／比較割込み
tim_upd_it	出力	タイマ更新イベント割込み
tim_brk_terr_ierr_it	出力	タイマブレーク、ブレーク 2、遷移エラー、およびインデックスエラー割込み
tim_trg_com_dir_idx_it	出力	タイマトリガ、転流、方向、およびインデックス割込み
tim_cc1_dma tim_cc2_dma tim_cc3_dma tim_cc4_dma	出力	タイマキャプチャ／比較 1～4 DMA リクエスト
tim_upd_dma	出力	タイマ更新 DMA リクエスト
tim_trg_dma	出力	タイマトリガ DMA リクエスト
tim_com_dma	出力	タイマ転流 DMA リクエスト

表 247、表 248、表 249、および表 250 に、tim_ti[4:1] 入力マルチプレクサに接続されるソースを列挙します。

表 247. tim_ti1 入力マルチプレクサへの相互接続

tim_ti1 入力	ソース		
	TIM1	TIM8	TIM20
tim_ti1_in0	TIM1_CH1	TIM8_CH1	TIM20_CH1
tim_ti1_in1	comp1_out	comp1_out	comp1_out
tim_ti1_in2	comp2_out	comp2_out	comp2_out
tim_ti1_in3	comp3_out	comp3_out	comp3_out
tim_ti1_in4	comp4_out	comp4_out	comp4_out
tim_ti1_in [15:5]	予約済みです。		

表 248. tim_ti2 入力マルチプレクサへの相互接続

tim_ti2 入力	ソース		
	TIM1	TIM8	TIM20
tim_ti2_in0	TIM1_CH2	TIM8_CH2	TIM20_CH2
tim_ti2_in [15:1]	予約済みです。		

表 249. tim_ti3 入力マルチプレクサへの相互接続

tim_ti3 入力	ソース		
	TIM1	TIM8	TIM20
tim_ti3_in0	TIM1_CH3	TIM8_CH3	TIM20_CH3
tim_ti2_in [15:1]	予約済みです。		

表 250. tim_ti4 入力マルチプレクサへの相互接続

tim_ti4 入力	ソース		
	TIM1	TIM8	TIM20
tim_ti4_in0	TIM1_CH4	TIM8_CH4	TIM20_CH4
tim_ti4_in [15:1]	予約済みです。		

表 251 に、tim_itr 入力マルチプレクサに接続される内部ソースを列挙します。

表 251. TIMx 内部トリガ接続

TIMx	TIM1	TIM8	TIM20
tim_itr0	予約済みです。	tim1_trgo	tim1_trgo
tim_itr1	tim2_trgo	tim2_trgo	tim2_trgo
tim_itr2	tim3_trgo	tim3_trgo	tim3_trgo
tim_itr3	tim4_trgo	tim4_trgo	tim4_trgo
tim_itr4	tim5_trgo	tim5_trgo	tim5_trgo
tim_itr5	tim8_trgo	予約済みです。	tim8_trgo
tim_itr6	tim15_trgo	tim15_trgo	tim15_trgo
tim_itr7	tim16_oc1	tim16_oc1	tim16_oc1
tim_itr8	tim17_oc1	tim17_oc1	tim17_oc1
tim_itr9	tim20_trgo	tim20_trgo	予約済みです。
tim_itr10	hrtim_out_sync2	hrtim_out_sync2	hrtim_out_sync2
tim_itr[15:11]	予約済みです。		

表 252 に、tim_etr 入力マルチプレクサに接続される内部ソースを列挙します。

表 252. tim_etr 入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て		
	TIM1	TIM8	TIM20
tim_etr0	TIM1_ETR	TIM8_ETR	TIM20_ETR
tim_etr1	comp1_out	comp1_out	comp1_out
tim_etr2	comp2_out	comp2_out	comp2_out
tim_etr3	comp3_out	comp3_out	comp3_out
tim_etr4	comp4_out	comp4_out	comp4_out
tim_etr5	comp5_out	comp5_out	comp5_out
tim_etr6	comp6_out	comp6_out	comp6_out
tim_etr7	comp7_out	comp7_out	comp7_out
tim_etr8	adc1_awd1	adc2_awd1	adc3_awd1
tim_etr9	adc1_awd2	adc2_awd2	adc3_awd2
tim_etr10	adc1_awd3	adc2_awd3	adc3_awd3
tim_etr11	adc4_awd1	adc3_awd1	adc5_awd1
tim_etr12	adc4_awd2	adc3_awd2	adc5_awd2
tim_etr13	adc4_awd3	adc3_awd3	adc5_awd3
tim_etr[15:14]	予約済みです。		

表 253、表 254、および表 255 に、tim_brk および tim_brk2 入力に接続されるソースを列挙します。

表 253. タイマ･ブレーク相互接続

tim_brk 入力	TIM1	TIM8	TIM20
TIM_BKIN	TIM1_BKIN ピン	TIM8_BKIN ピン	TIM20_BKIN ピン
tim_brk_cmp1	comp1_out	comp1_out	comp1_out
tim_brk_cmp2	comp2_out	comp2_out	comp2_out
tim_brk_cmp3	comp3_out	comp3_out	comp3_out
tim_brk_cmp4	comp4_out	comp4_out	comp4_out
tim_brk_cmp5	comp5_out	comp5_out	comp5_out
tim_brk_cmp6	comp6_out	comp6_out	comp6_out
tim_brk_cmp7	comp7_out	comp7_out	comp7_out
tim_brk_cmp8	予約済みです。		

表 254. タイマ･ブレーク 2 相互接続

tim_brk2 入力	TIM1	TIM8	TIM20
TIM_BKIN2	TIM1_BKIN2 ピン	TIM8_BKIN2 ピン	TIM20_BKIN2 ピン
tim_brk2_cmp1	comp1_out	comp1_out	comp1_out
tim_brk2_cmp2	comp2_out	comp2_out	comp2_out
tim_brk2_cmp3	comp3_out	comp3_out	comp3_out
tim_brk2_cmp4	comp4_out	comp4_out	comp4_out
tim_brk2_cmp5	comp5_out	comp5_out	comp5_out
tim_brk2_cmp6	comp6_out	comp6_out	comp6_out
tim_brk2_cmp7	comp7_out	comp7_out	comp7_out
tim_brk2_cmp8	予約済みです。		

表 255. システム･ブレーク相互接続

tim_sys_brk 入力	TIM1 / TIM8 / TIM20	SYSCFG_CFGR2 レジスタの イネーブルビット
tim_sys_brk0	FPU 搭載 Cortex®-M4 ロックアップ	CLL
tim_sys_brk1	プログラム可能な電圧検出器（PVD）	PVDL
tim_sys_brk2	SRAM パリティ･エラー	SPL
tim_sys_brk3	Flash ダブル ECC エラー	ECCL
tim_sys_brk4	クロック･セキュリティ･システム（CSS）	なし（常に有効）

表 256 に、tim_ocref_clr 入力マルチプレクサに接続される内部ソースを列挙します。

表 256. ocref_clr 入力マルチプレクサへの相互接続

タイマ OCREF クリア信号	タイマ OCREF クリア信号割り当て		
	TIM1	TIM8	TIM20
tim_ocref_clr0	comp1_out	comp1_out	comp1_out
tim_ocref_clr1	comp2_out	comp2_out	comp2_out
tim_ocref_clr2	comp3_out	comp3_out	comp3_out
tim_ocref_clr3	comp4_out	comp4_out	comp4_out
tim_ocref_clr4	comp5_out	comp5_out	comp5_out
tim_ocref_clr5	comp6_out	comp6_out	comp6_out
tim_ocref_clr6	comp7_out	comp7_out	comp7_out
tim_ocref_clr7	予約済みです。		

28.3.3 タイムベース･ユニット

プログラム可能な高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットカウンタです。カウンタはカウントアップ、カウントダウン、またはアップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、カウンタが動作中でも、ソフトウェアで読み書きができます。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力 tim_cnt_ck から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

注： **TIMx_CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があります。**

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周します。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 270 と 図 271 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 270. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

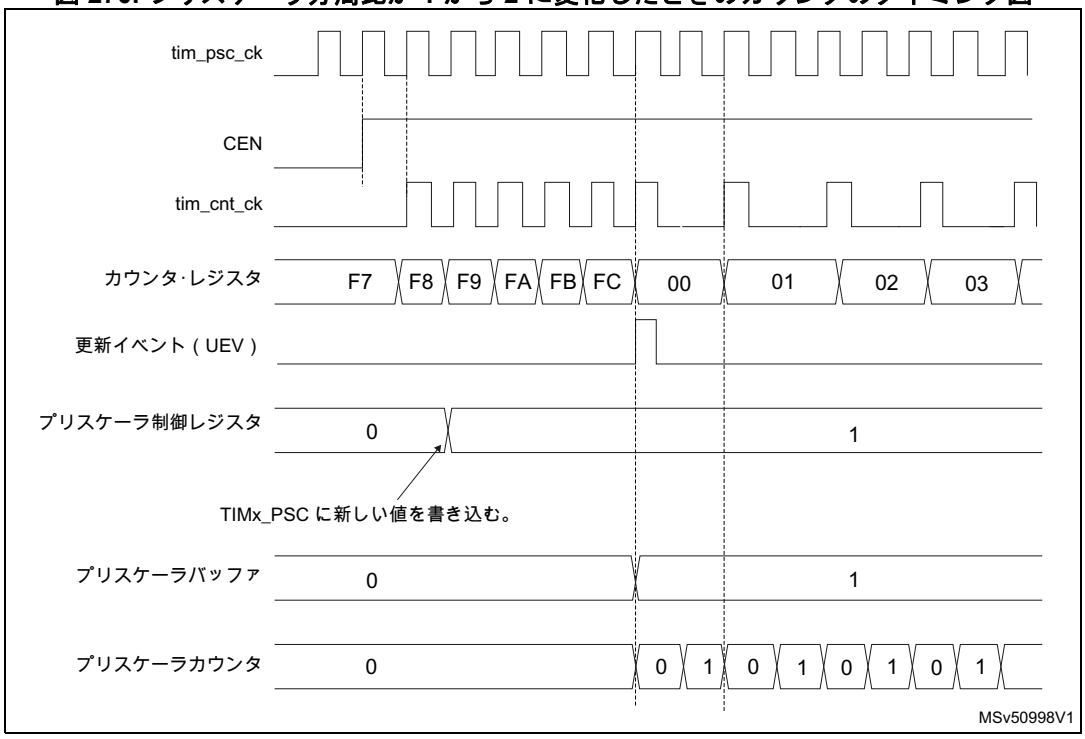
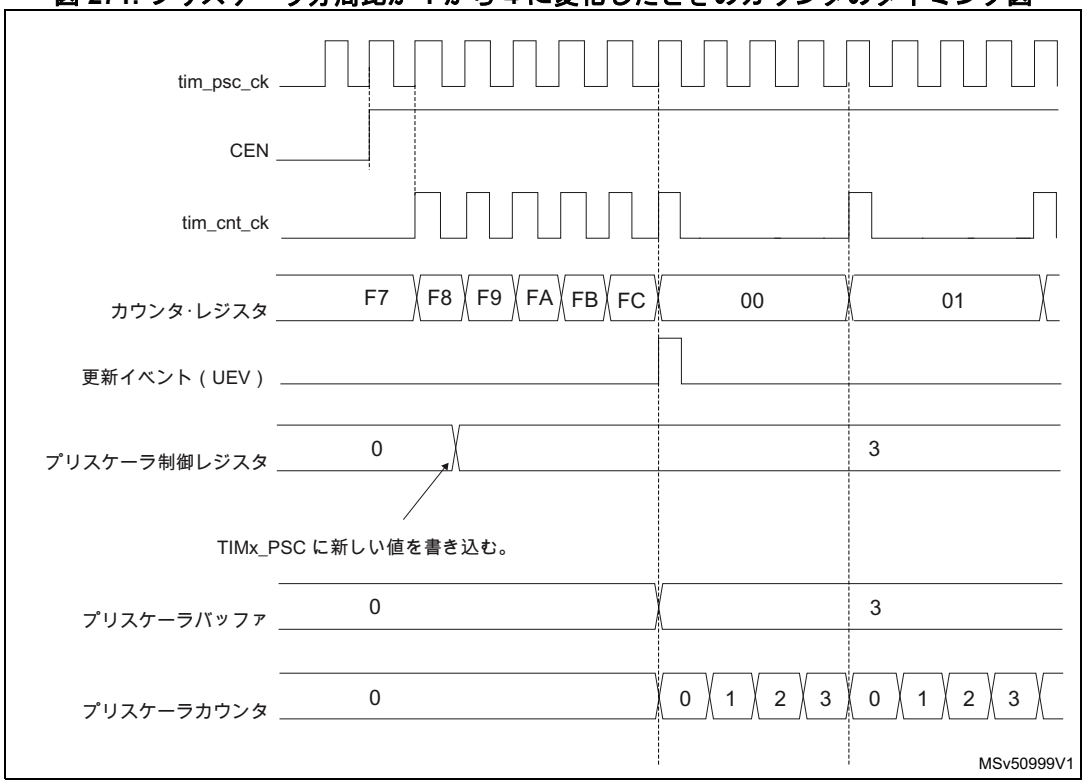


図 271. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



28.3.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 272. 内部クロック分周比が 1 の場合のカウンタのタイミング図

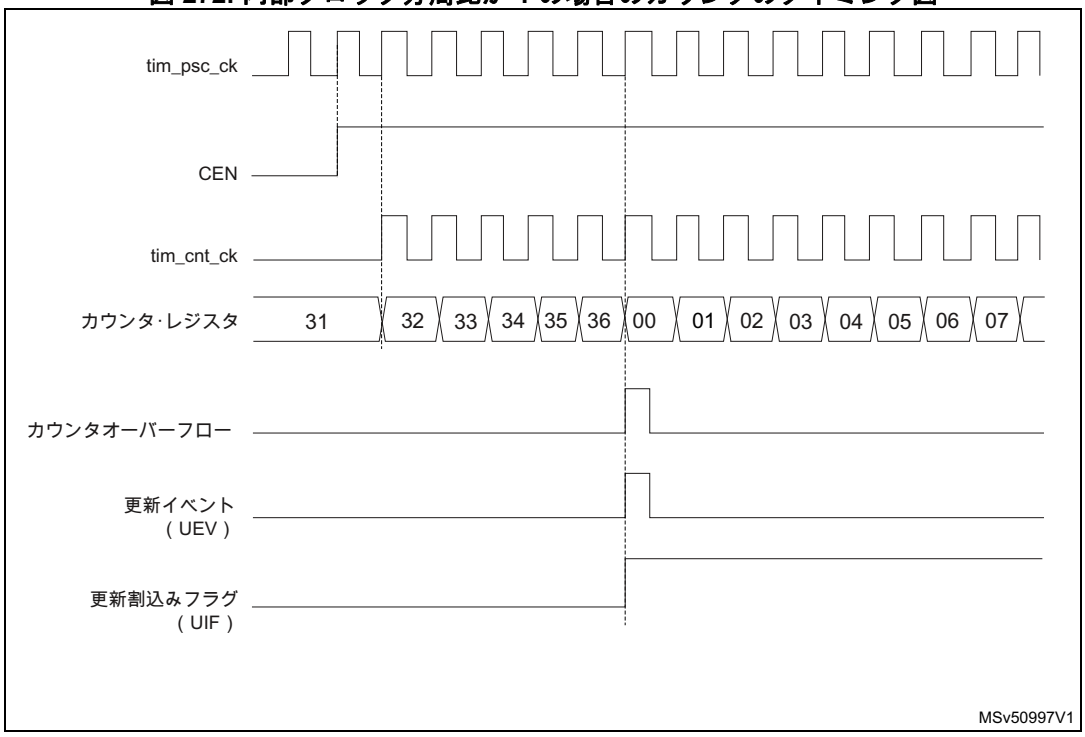


図 273. 内部クロック分周比が 2 の場合のカウンタのタイミング図

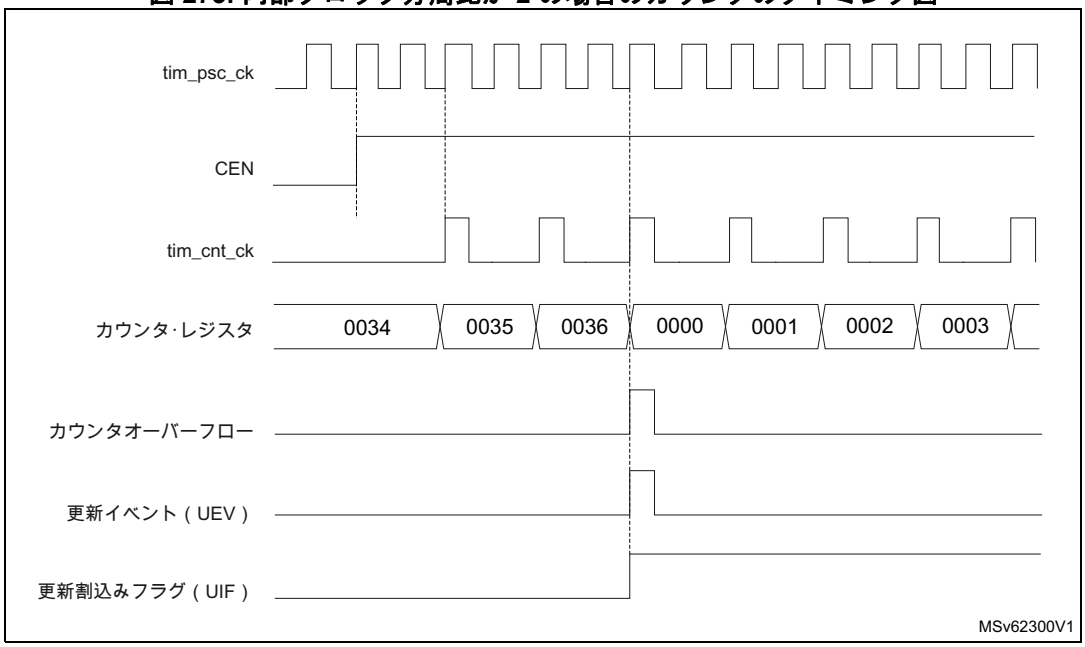


図 274. 内部クロック分周比が 4 の場合のカウンタのタイミング図

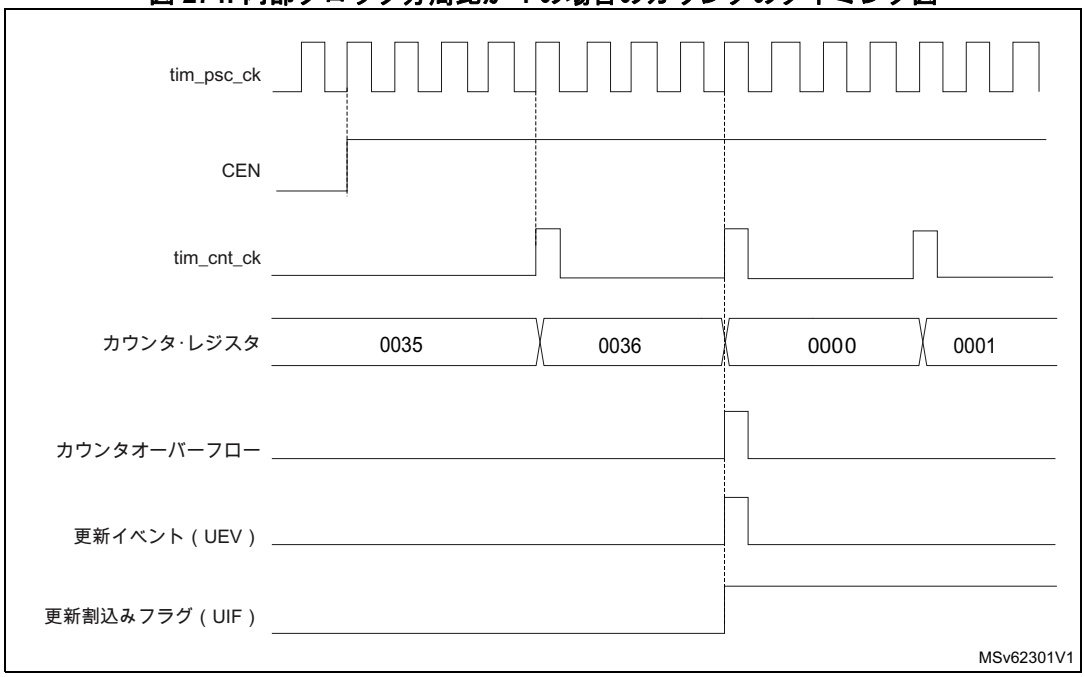


図 275. 内部クロック分周比が N の場合のカウンタのタイミング図

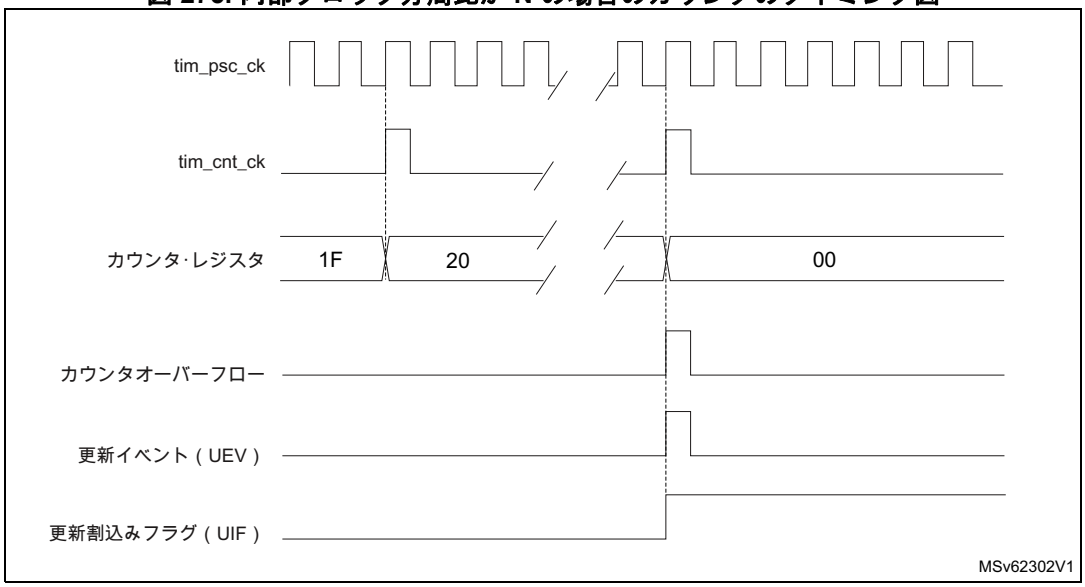


図 276. ARPE=0の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロードされない）

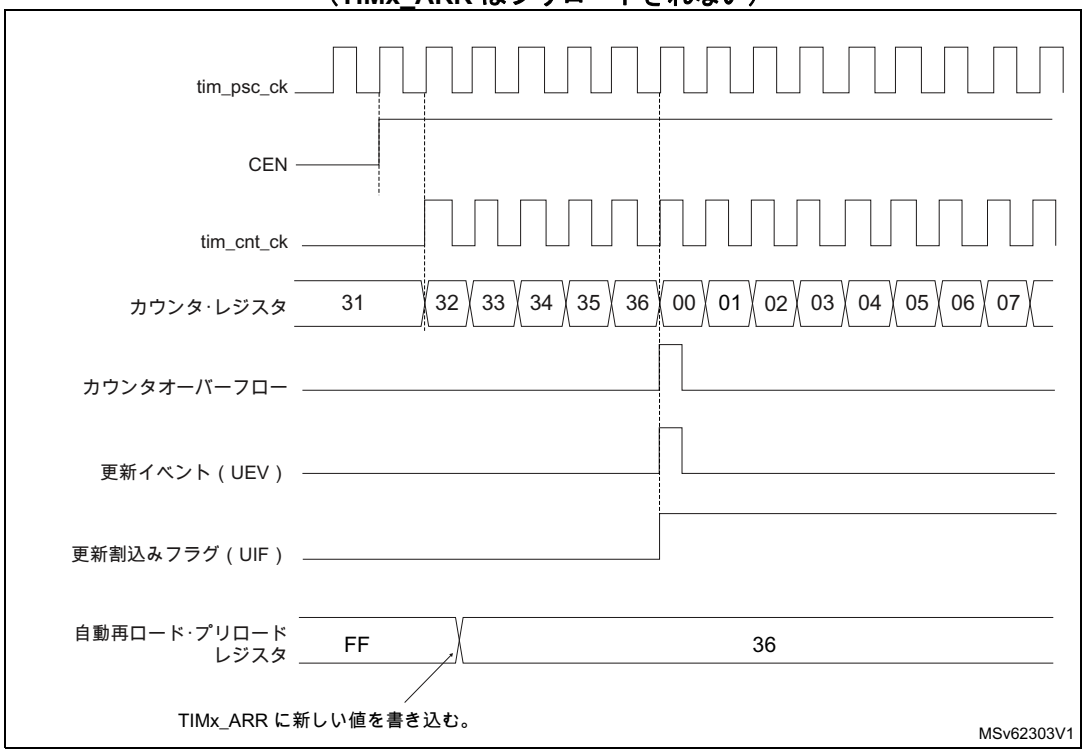
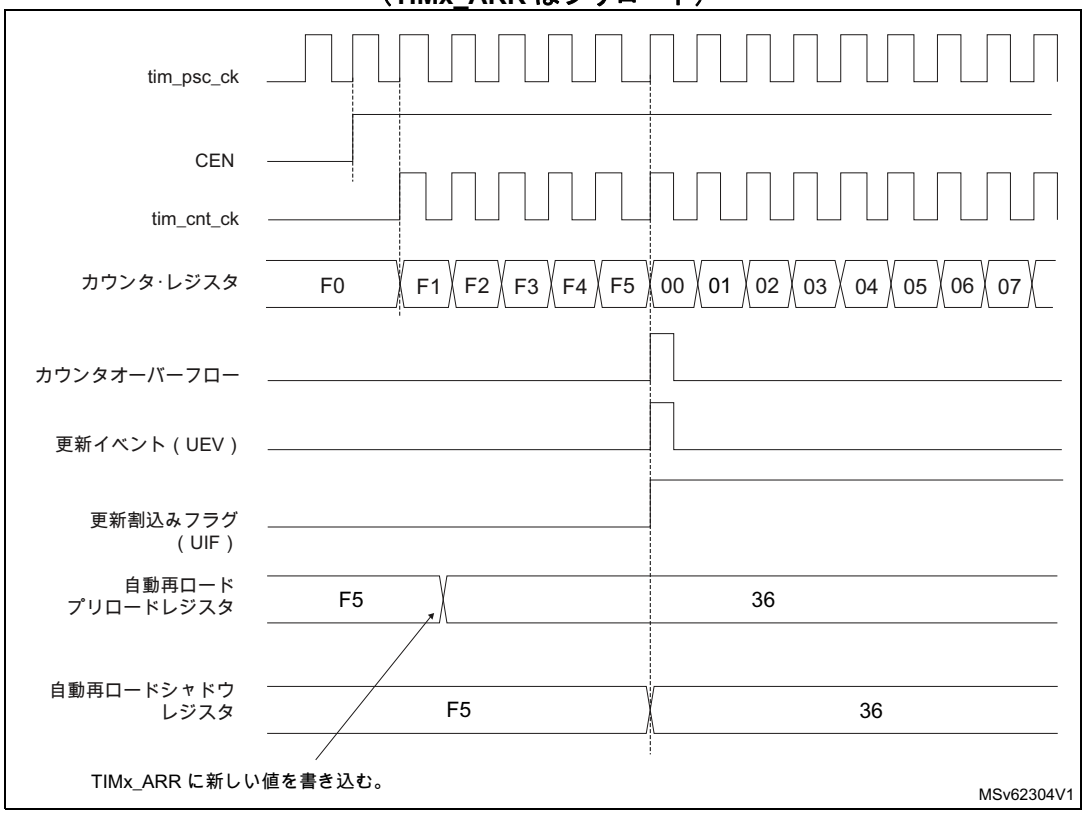


図 277. ARPE=1の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロード）



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）+ 1 までダウンカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのアンダーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 278. 内部クロック分周比が 1 の場合のカウンタのタイミング図

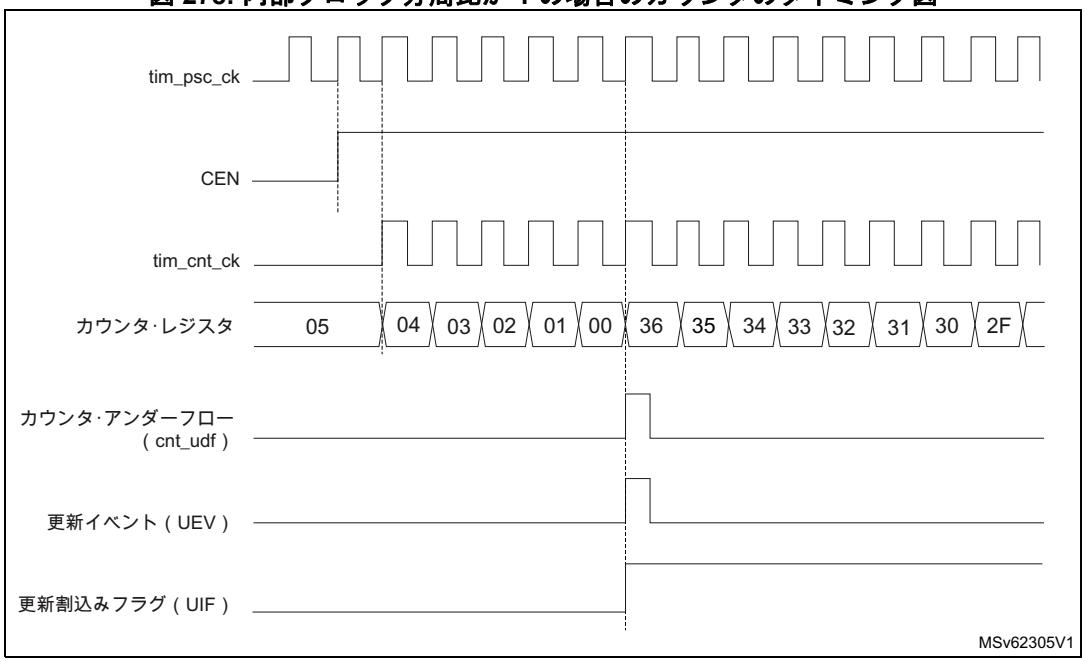


図 279. 内部クロック分周比が 2 の場合のカウンタのタイミング図

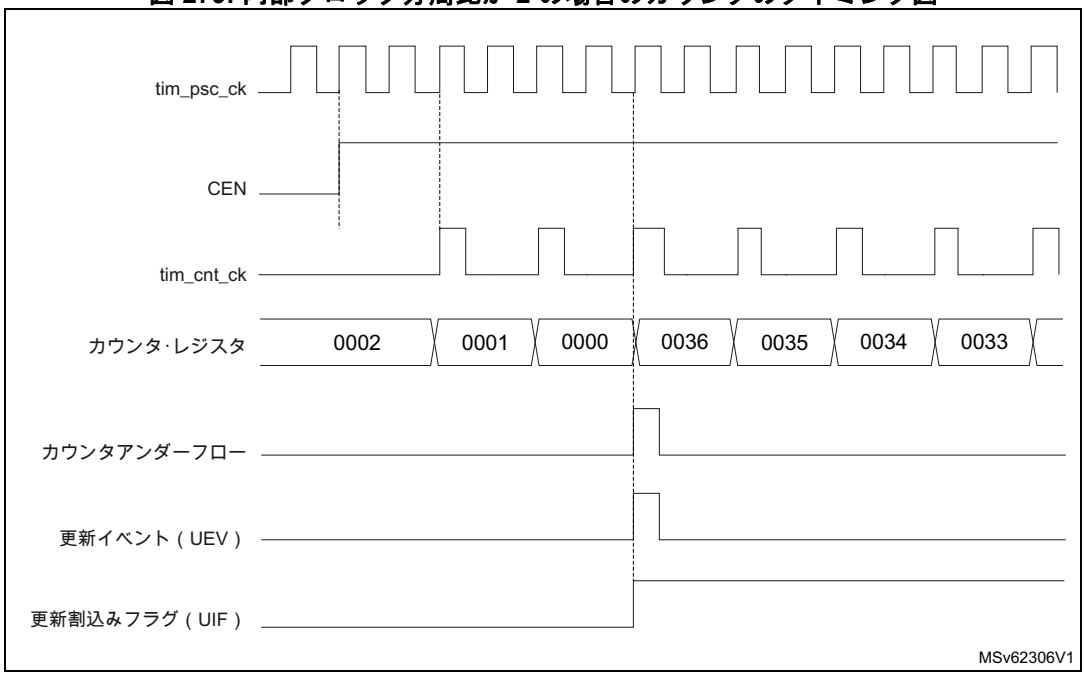


図 280. 内部クロック分周比が 4 の場合のカウンタのタイミング図

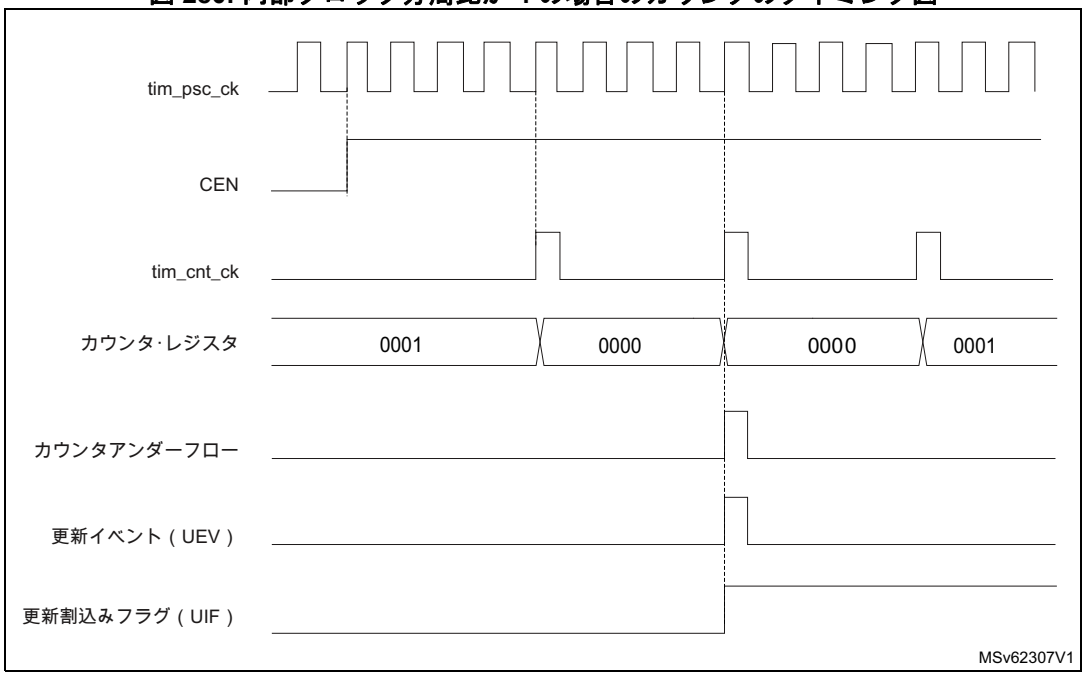


図 281. 内部クロック分周比が N の場合のカウンタのタイミング図

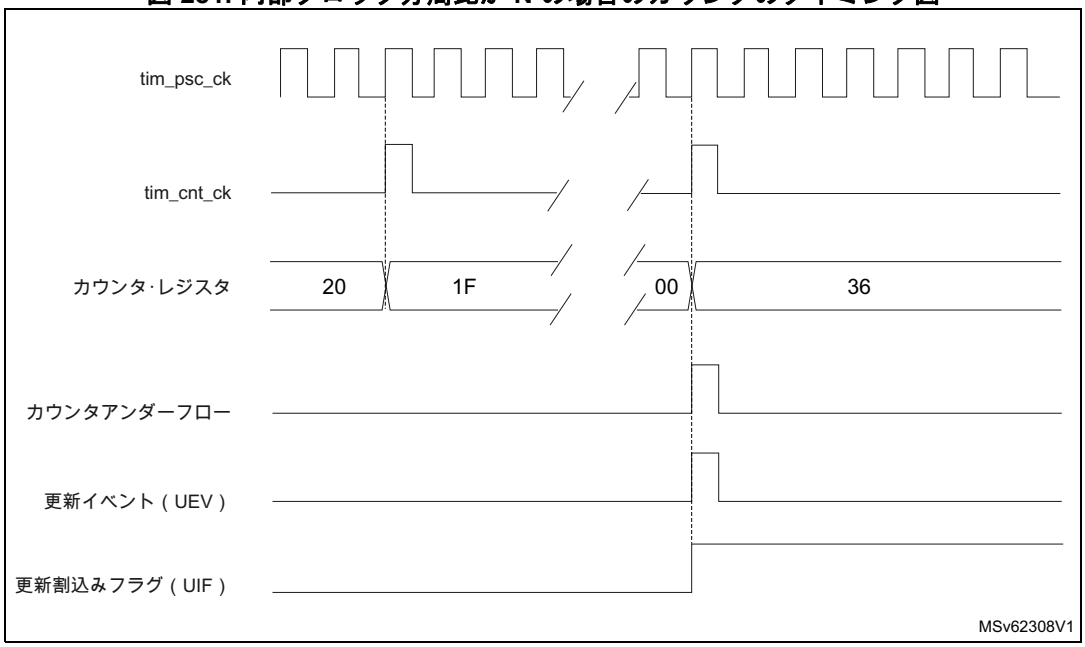
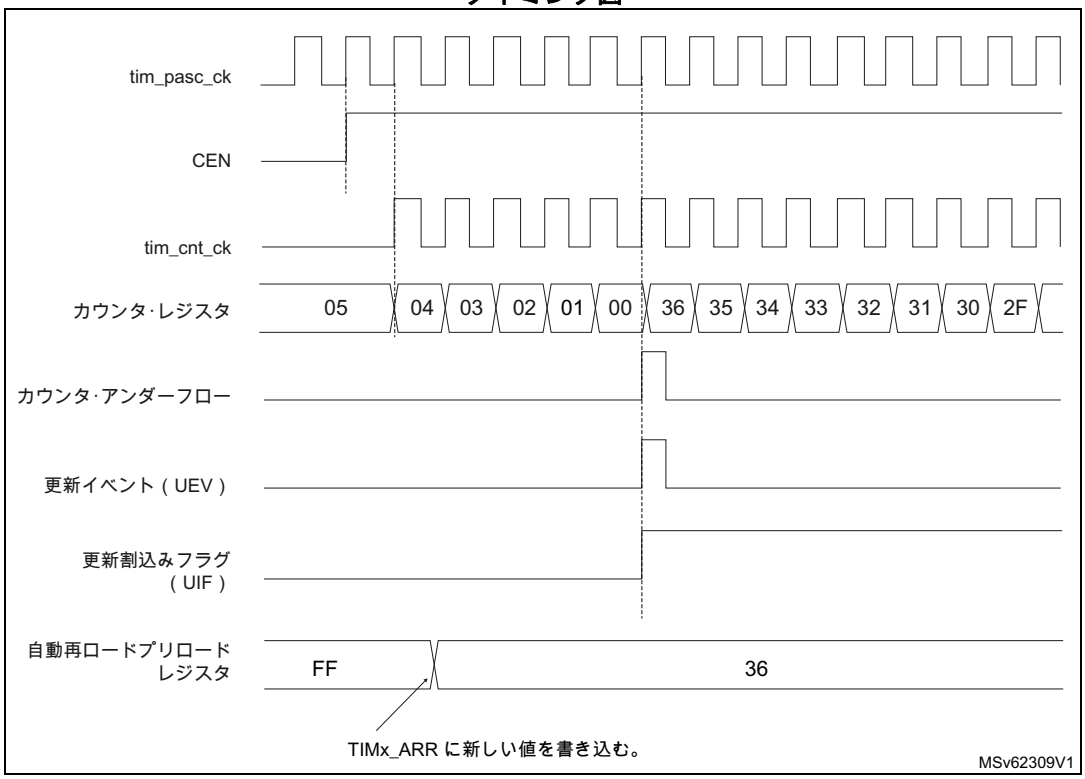


図 282. 繰り返しカウンタが使用されていない場合の更新イベント時のカウンタのタイミング図



センターアラインモード（アップ／ダウンカウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）－1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、TIMx_CR1 レジスタの方向ビット（DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

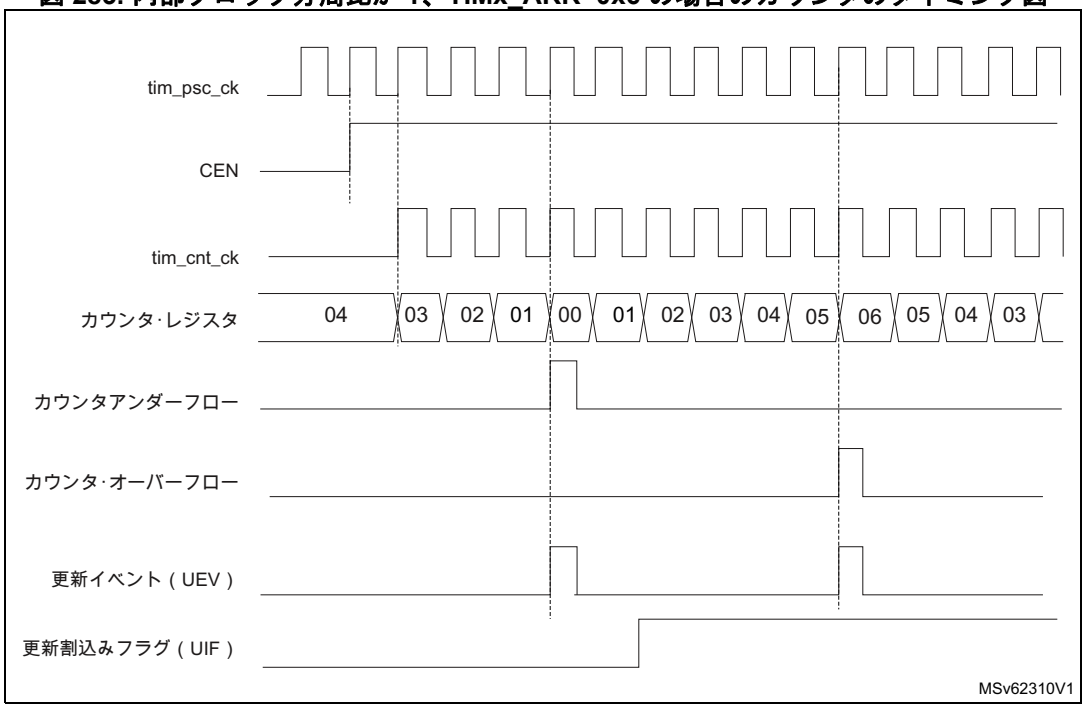
さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合、UG ビットをセットすると UEV 更新イベントが生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 283. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、セクション 28.6：TIM1/TIM8/TIM20 レジスタを参照）。

図 284. 内部クロック分周比が 2 の場合のカウンタのタイミング図

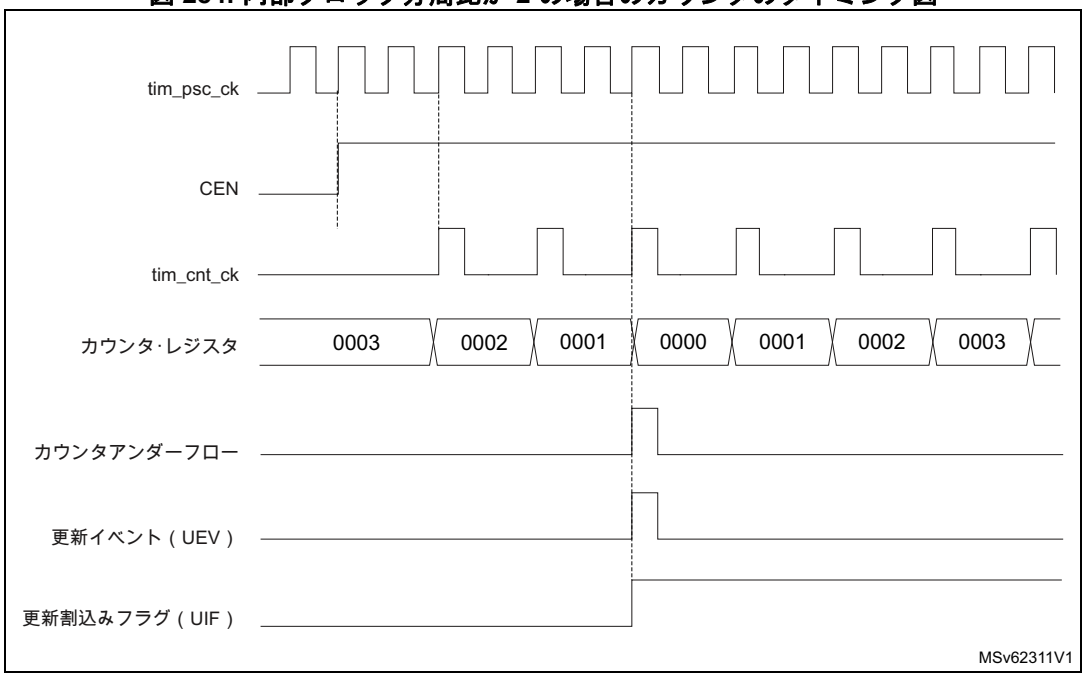


図 285. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図

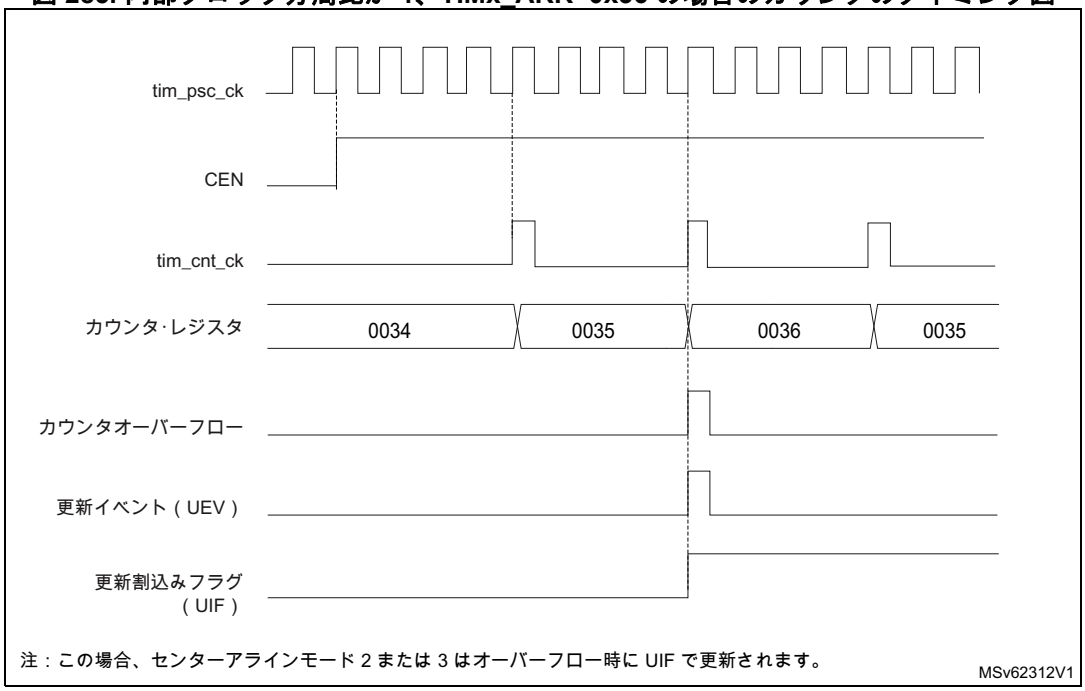


図 286. 内部クロック分周比が N の場合のカウンタのタイミング図

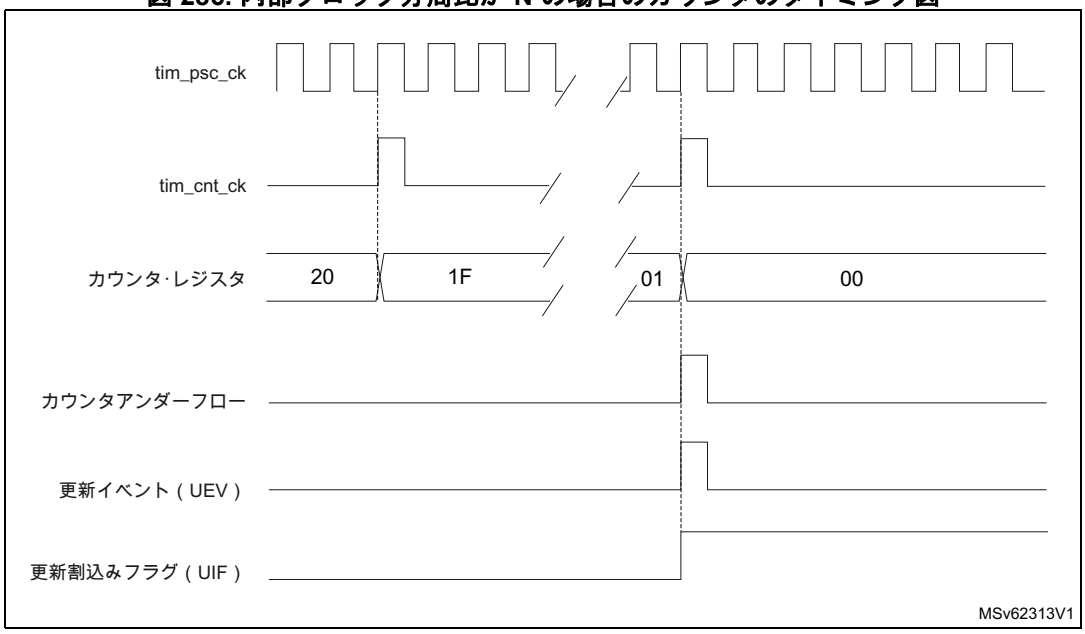


図 287. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時のカウンタのタイミング図

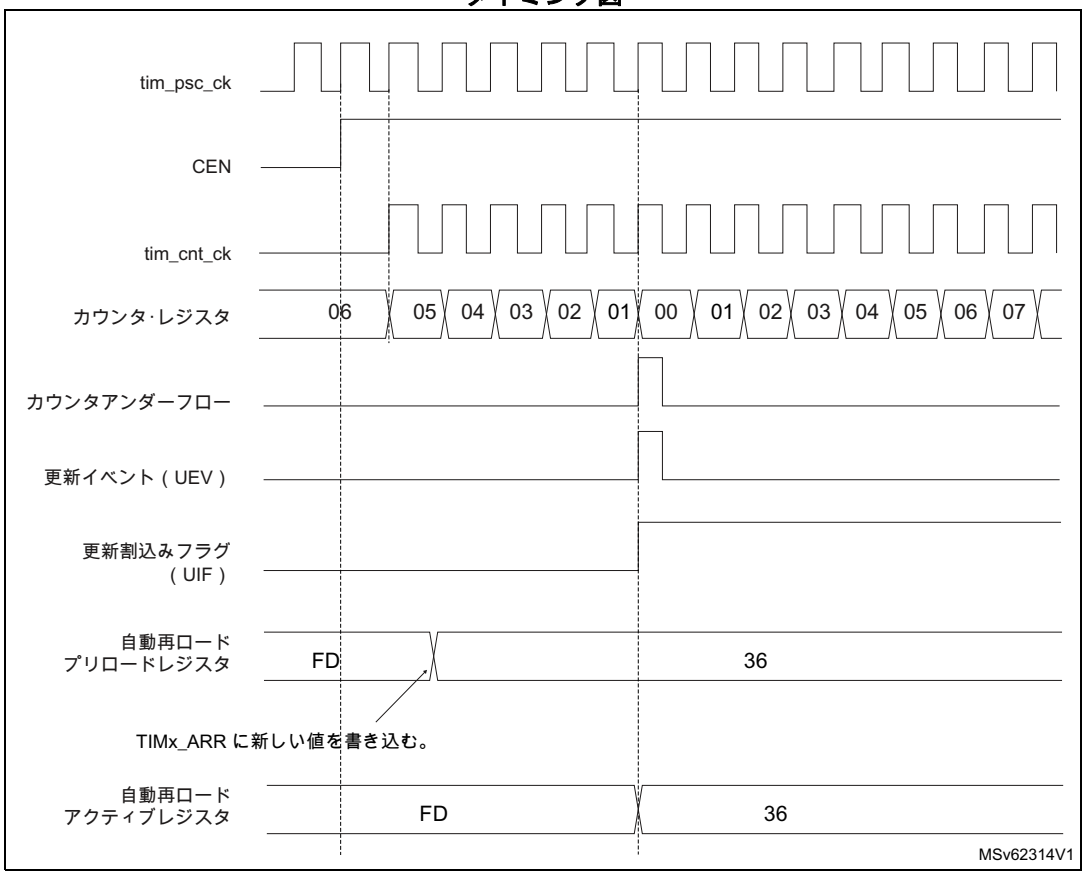
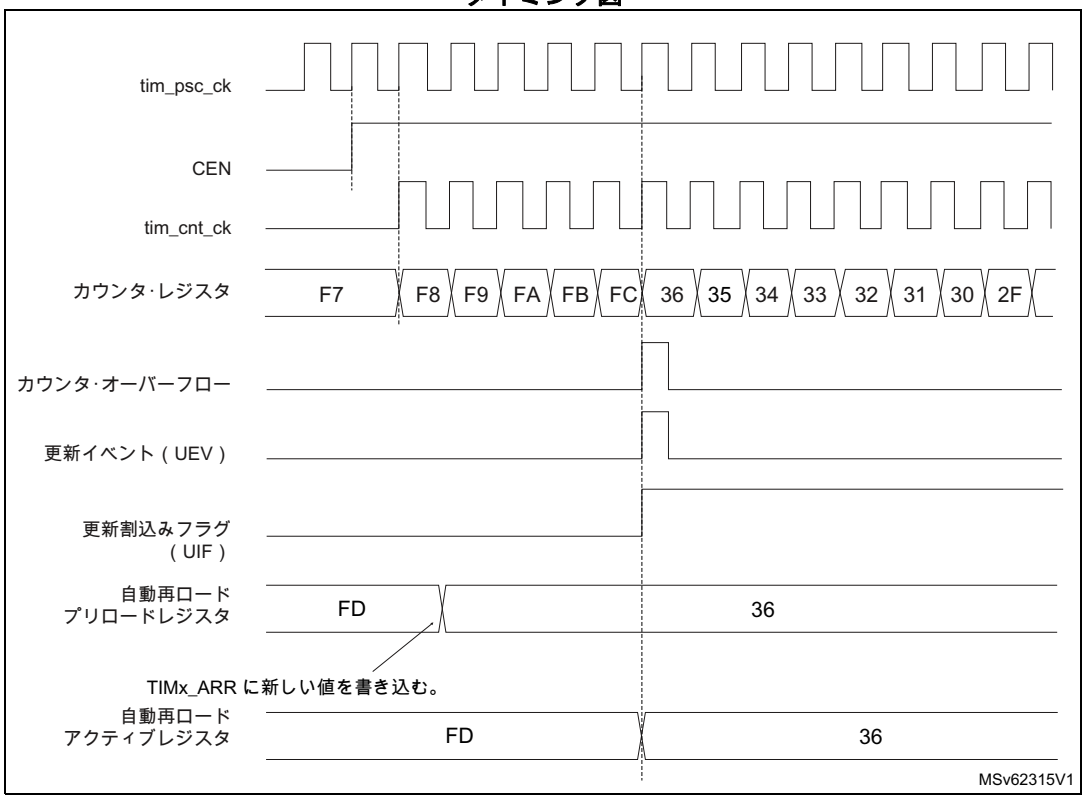


図 288. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



28.3.5 繰り返しカウンタ

セクション 28.3.3：タイムベース·ユニットに、カウンタオーバーフロー/アンダーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N+1 回目のカウンタオーバーフローまたはアンダーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ/比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、次の場合にデクリメントします。

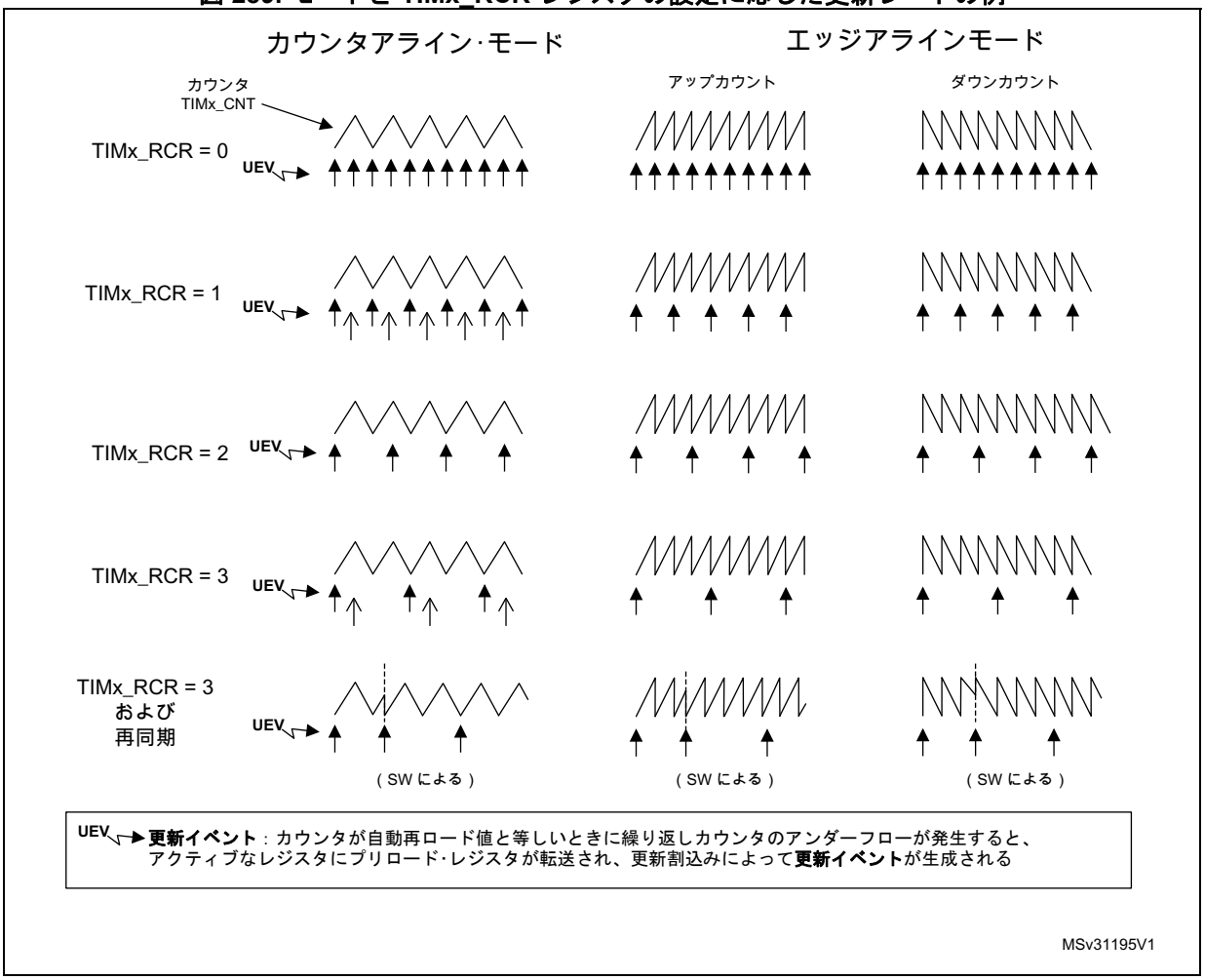
- アップカウントモードで、カウンタオーバーフローごと
- ダウンカウントモードで、カウンタアンダーフローごと
- センターアラインモードで、カウンタオーバーフローとカウンタアンダーフローごと最大繰り返し回数は 32768 PWM サイクルに限られますが、PWM 周期ごとにデューティサイクルを 2 回更新することが可能になります。センターアラインモードで比較レジスタの値を PWM 周期あたり 1 回のみ更新するときには、パターンが対称なので、最大精度は $2xT_{ck}$ です。

繰り返しカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（図 289 を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

センターアラインモードでは、RCR が奇数の場合、RCR レジスタが書き込まれたタイミングおよびカウンタが開始されたタイミングに応じてオーバーフローまたはアンダーフロー時に更新イベントが発生します。カウントの開始前に RCR が書き込まれた場合は、アンダーフローで、UEV が発生します。カウントの開始後に RCR が書き込まれた場合は、オーバーフローで UEV が発生します。

たとえば、TIMx_RCR = 3 の場合、TIMx_RCR の書込みタイミングに応じて 4 回目のオーバーフローイベントまたはアンダーフローイベントごとに UEV が発生します。

図 289. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



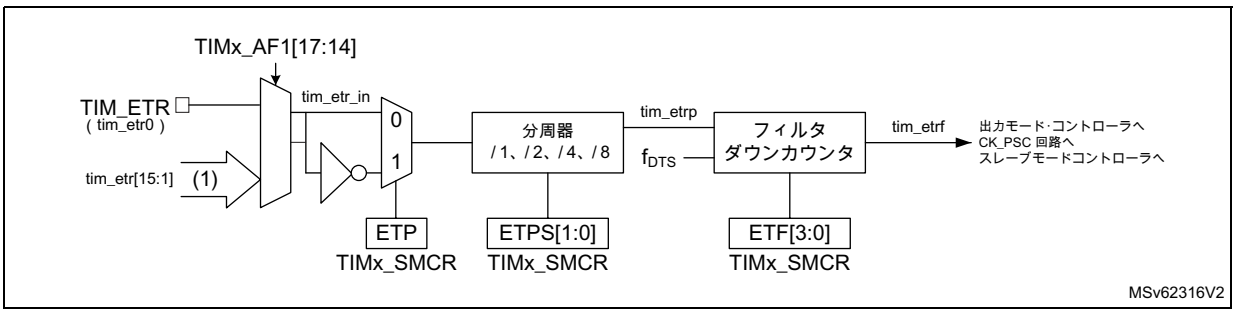
28.3.6 外部トリガ入力

タイマには外部トリガ入力 tim_etr_in 機能があります。以下の目的で使用できます。

- 外部クロック（外部クロックモード 2、セクション 28.3.7：を参照）
- スレーブモードのトリガ（セクション 28.3.30：を参照）
- サイクルごとの電流調整の PWM リセット入力（セクション 28.3.9：を参照）

以下の図 290 は、tim_etr_in 入力の前処理について説明しています。入力の極性は、TIMxSMCR レジスタの ETP ビットで定義されています。トリガは ETPS[1:0] ビットフィールドでプログラムされた分周器でプリスケールし、ETF[3:0] ビットフィールドでデジタル的にフィルタリングすることができます。結果の信号（tim_etrf）は、次の 3 つの目的に使用できます。すなわち、外部クロックとして、出力を調整する（通常は電流制限のために PWM 出力をリセットする）ため、およびスレーブモードコントローラのトリガとしてです。

図 290. 外部トリガ入力ブロック



tim_etr_in 入力は、入力ピン（デフォルト設定）、内部ソースといった複数のソースから入力されます。この選択は、TIMx_AF1 レジスタの ETRSEL[3:0] ビットフィールドで行います。

製品の etr_in 入力へ接続されるソースの一覧についてはセクション 28.3.2:TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

28.3.7 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

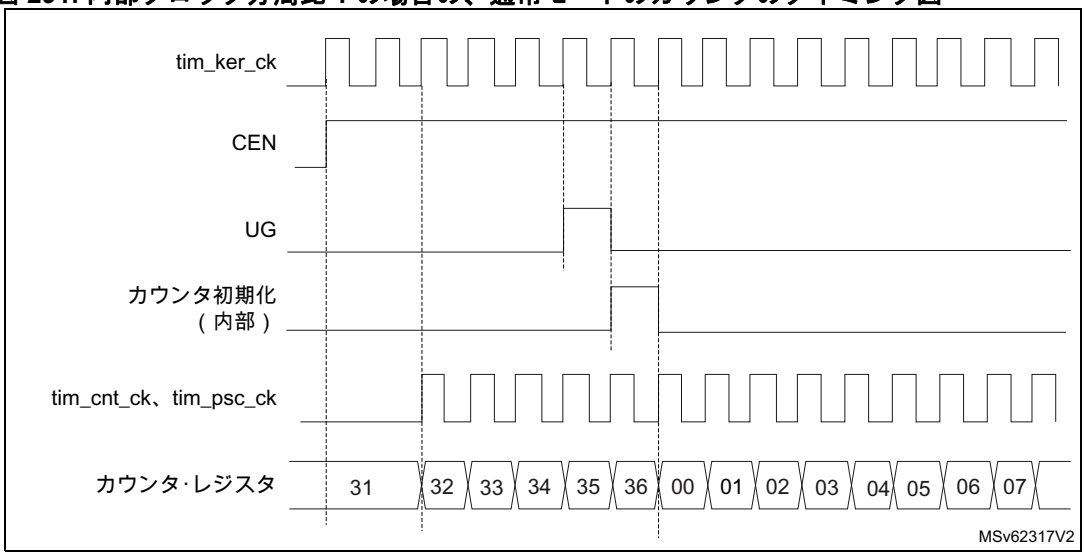
- 内部クロック（tim_ker_ck）
- 外部クロックモード 1：外部入力ピン（tim_ti1 または tim_ti2）
- 外部クロックモード 2：外部トリガ入力（tim_etr_in）
- エンコーダモード

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（SMS=000）、CEN、DIR（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれるとすぐに、プリスケーラには内部クロック tim_ker_ck が供給されます。

図 291 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

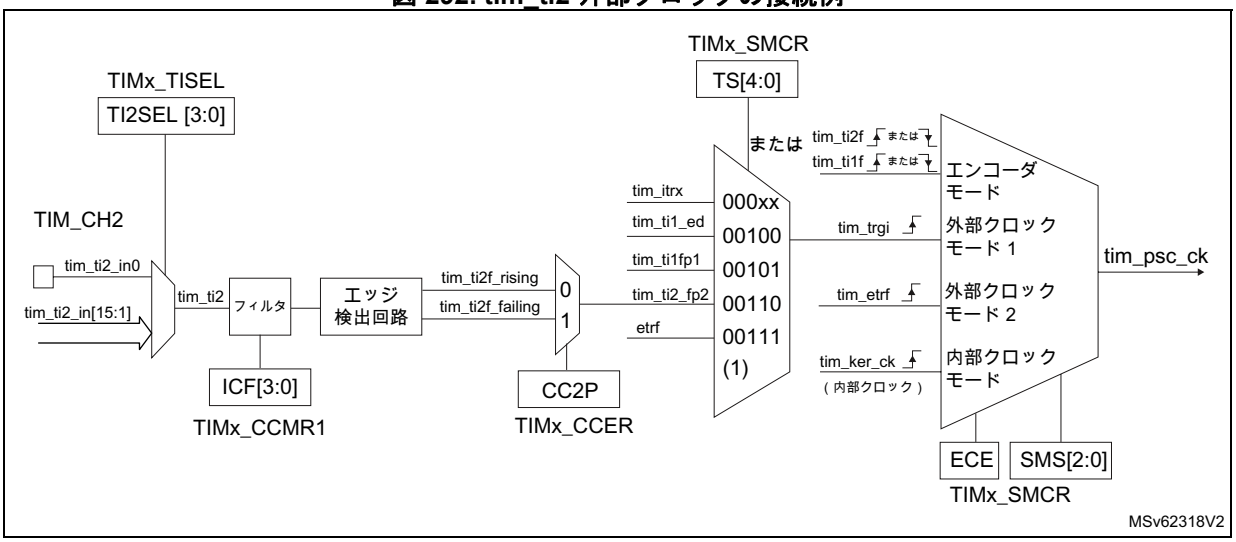
図 291. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 292. tim_ti2 外部クロックの接続例



1. 01000 から 11111 の範囲のコードは予約済み。

たとえば、tim_ti2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

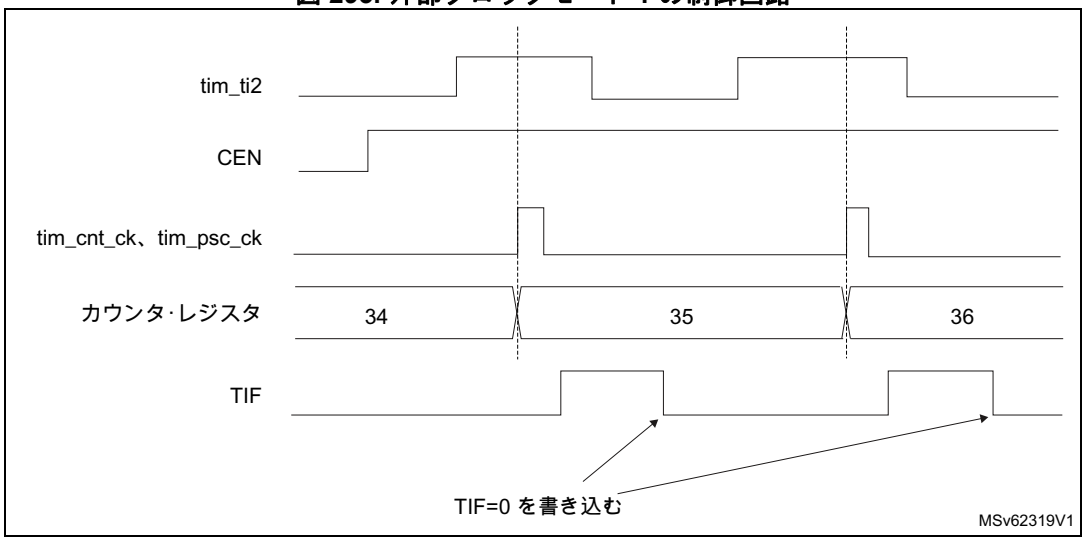
1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が tim_ti2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
3. CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして tim_ti2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。**

tim_ti2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際のカウンタクロックまでの間には、tim_ti2 入力の再同期回路による遅延があります。

図 293. 外部クロックモード 1 の制御回路



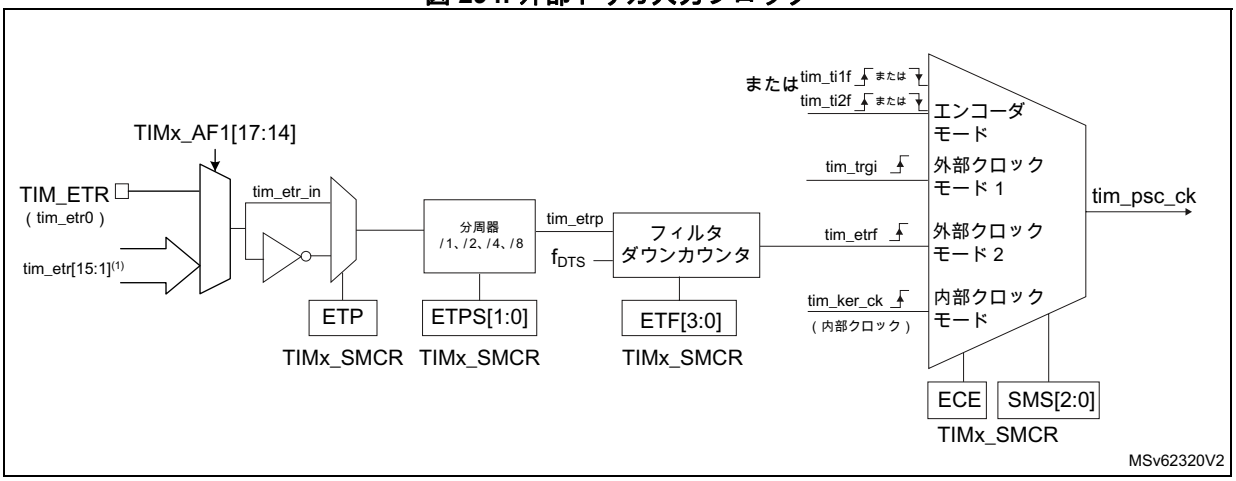
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 tim_etr_in の立ち上がりまたは立ち下がりエッジごとにカウントします。

図 294 に、外部トリガ入力ブロックの概要を示します。

図 294. 外部トリガ入力ブロック



1. セクション 28.3.2 : TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

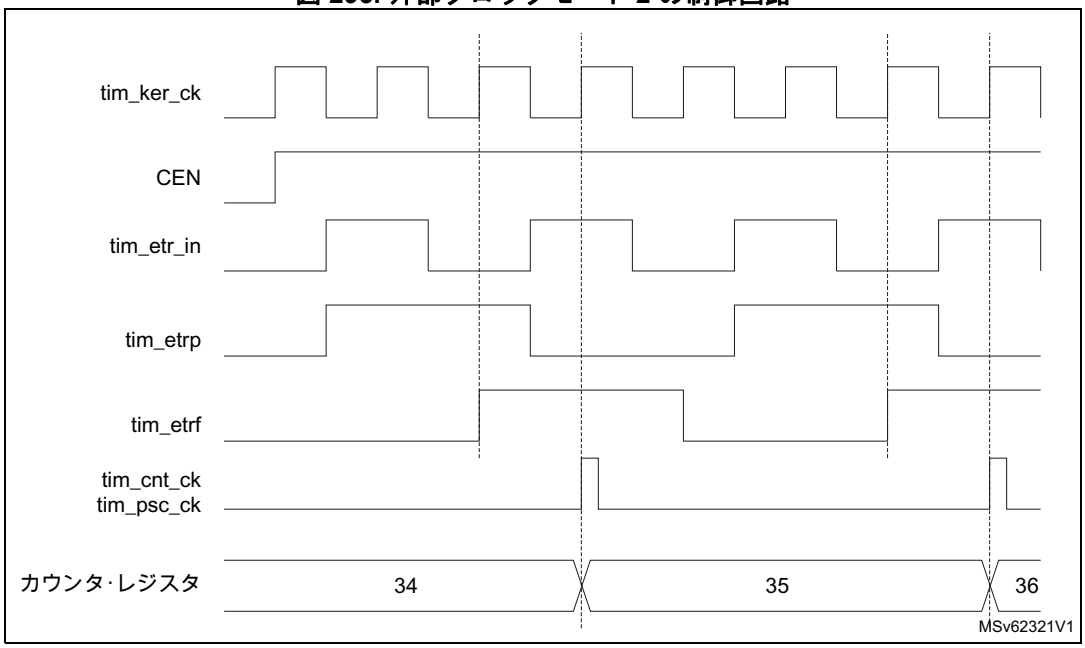
たとえば、tim_etr_in の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、tim_etr_in 入力の立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の tim_etr_in 立ち上がりエッジごとに 1 回カウントします。

tim_etr_in の立ち上がりエッジから実際のカウンタクロックまでの間に、tim_etrp 信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大で tim_ker_ck 周波数の 1/4 です。ETRP 信号が高速の場合、ユーザは適切な ETPS プリスケーラー設定によって外部信号の分周比を適用する必要があります。

図 295. 外部クロックモード 2 の制御回路



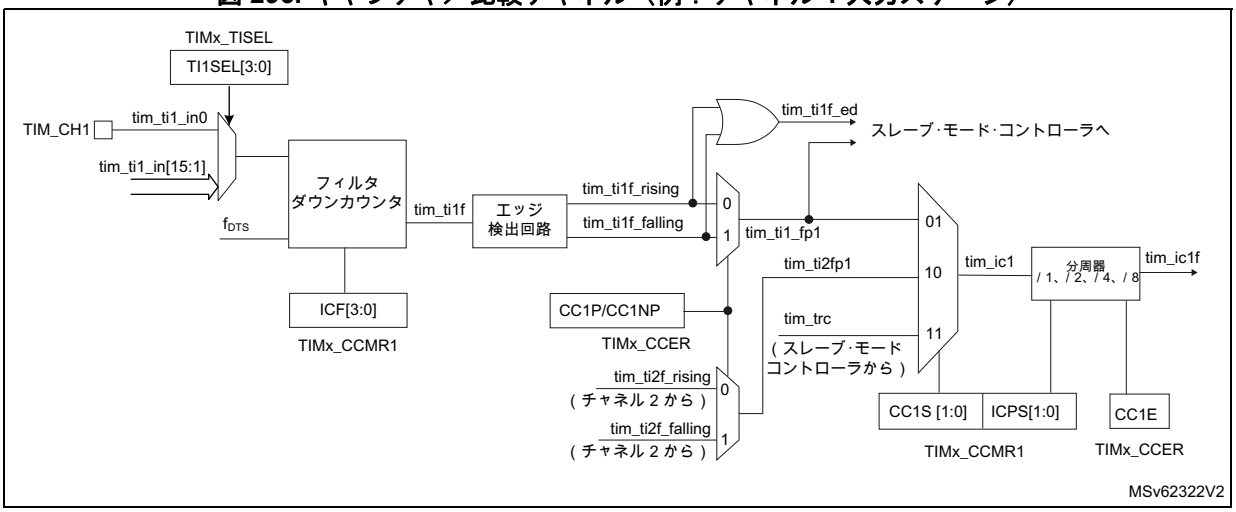
28.3.8 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（チャネル 5 および 6 を除くデジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 296 から 図 299 に、1 つのキャプチャ／比較 チャネルの概要を示します。

入力ステージは、対応する tim_tix 入力をサンプリングして、フィルタリングを行った信号 tim_tixf を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 296. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、その後、基準として使用される中間波形 tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図 297. キャプチャ／比較チャネル 1 メイン回路

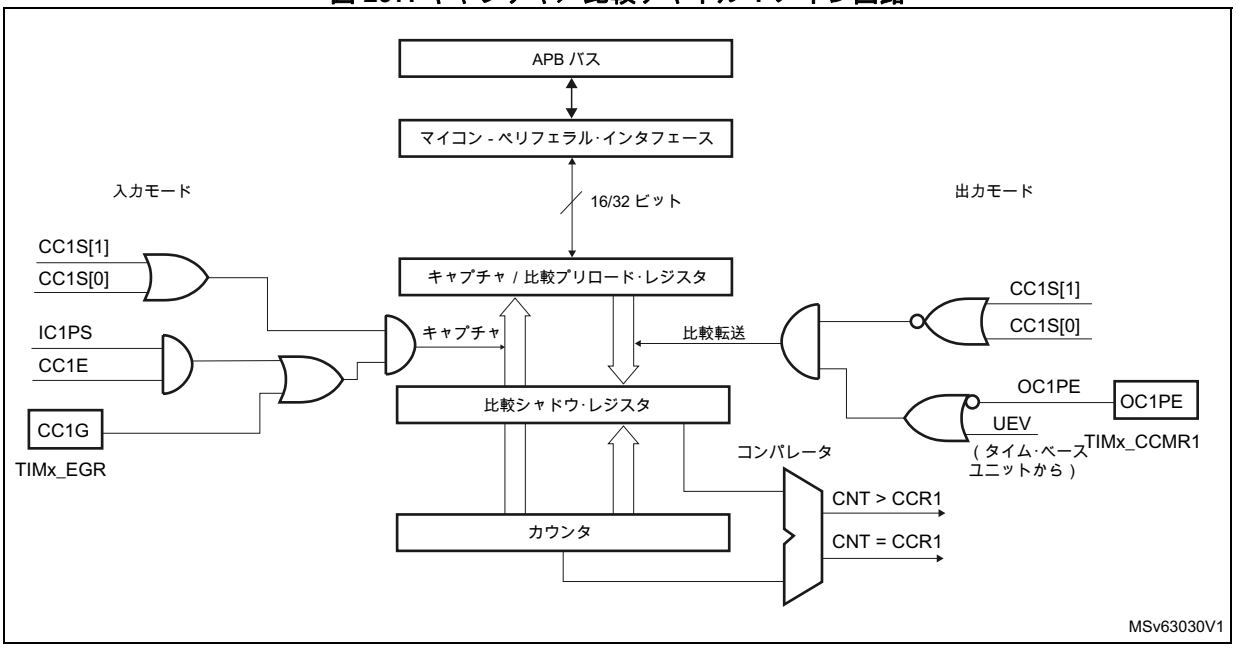
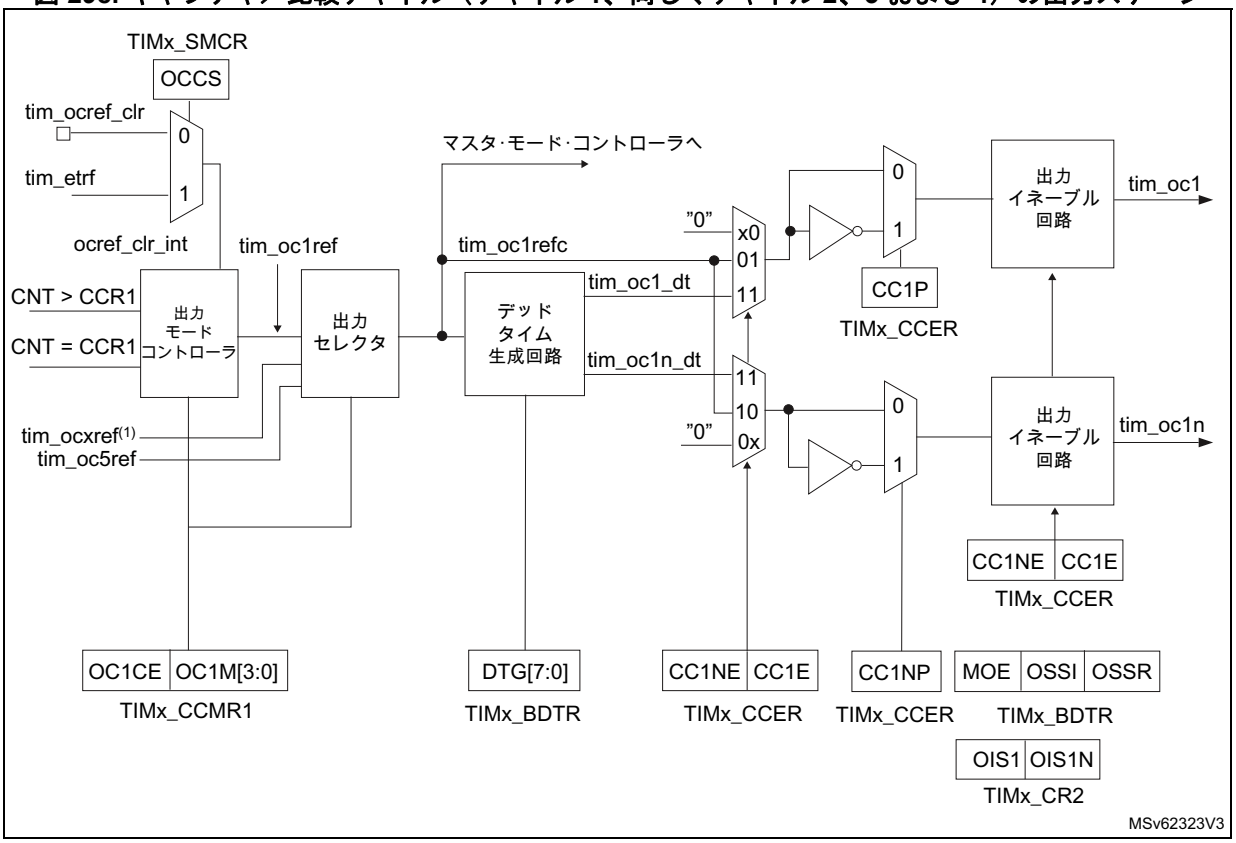
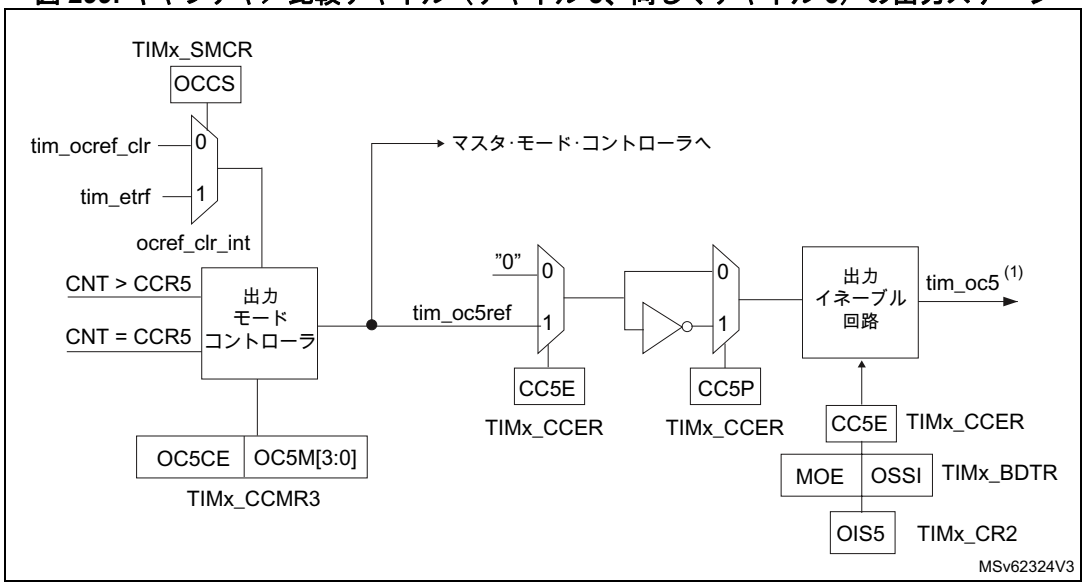


図 298. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3 および 4）の出力ステージ



1. tim_ocxref、ここで x は相補チャネルのランク

図 299. キャプチャ／比較チャネル（チャネル 5、同じくチャネル 6）の出力ステージ



1. 外部的には使用できません。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

28.3.9 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって遷移が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、tim_ti1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

- アクティブ入力を選択します。TIMx_CCR1 は tim_ti1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
- タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力が tim_tix の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、tim_ti1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
- tim_ti1 チャネルのアクティブ変化のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“0”を書き込みます（この場合、立ち上がりエッジの選択）。
- 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
- TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
- 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

28.3.10 PWM 入力モード

このモードでは、単一の tim_tix 入力に接続された PWM 信号の周期とデューティサイクルの両方を以下のように測定できます。

- TIMx_CCR1 レジスタは周期値（2 つの連続した立ち上がりエッジ間の間隔）を保持します。
- TIM_CCR2 レジスタはパルス幅（2 つの連続した立ち上がりと立ち下がりエッジ間の間隔）を保持します。

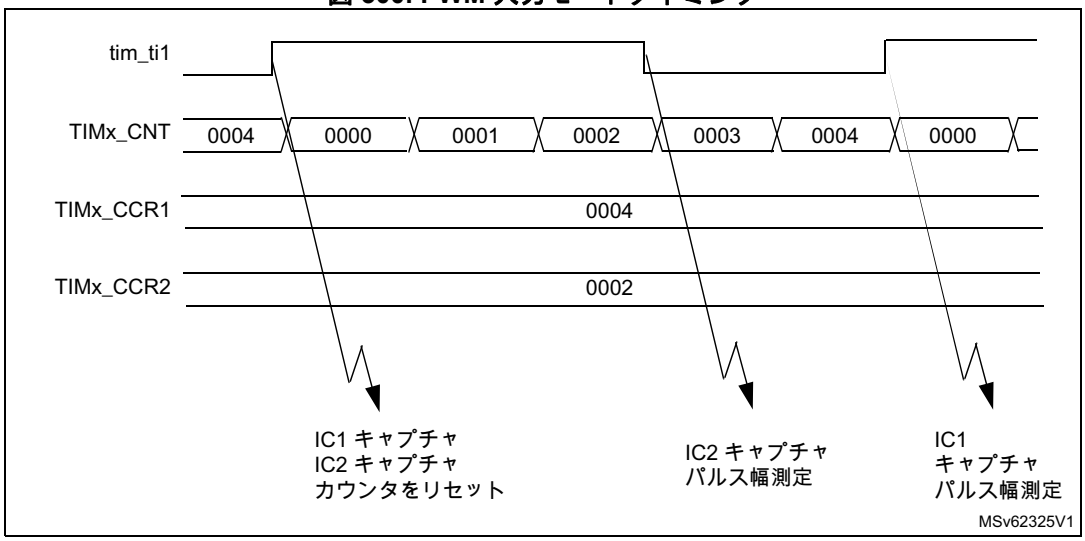
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2 つの ICx 信号が同じ tim_tixfp1 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの tim_tixfp 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

tim_ti1 に適用される PWM 信号の周期とパルス幅は、以下の手順を使用して測定できます。

- TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
- CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、tim_ti1fp1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
- TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
- CC2P ビットと CC2NP ビットに CC2P/CC2NP=“10”を書き込むことによって（立ち下がりエッジでアクティブ）、tim_ti1fp2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
- TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（tim_ti1fp1 を選択）、有効なトリガ入力を選択します。
- TIMx_SMCR レジスタの SMS ビットに 0100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
- TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 300. PWM 入力モードタイミング



28.3.11 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS ビット = 00）では、各出力比較信号（tim_ocxref、そして tim_ocx/tim_ocxn）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 0101 を書き込む必要があります。これにより、tim_ocxref は強制的にハイレベルになり（tim_ocxref は常にアクティブハイ）、tim_ocx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（tim_ocx アクティブハイ）=> tim_ocx は強制的にハイレベルになります。

tim_ocxref 信号は、TIMx_CCMRx レジスタの OCxM ビットに 0100 を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

28.3.12 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。マイクロコントロール内でチャネル 5 および 6 のみが使用可能である場合に、チャネル 1 から 4 を出力できます（たとえば、合成波形生成または ADC トリガのため）。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=0000）、アクティブにセットされるか（OCxM=0001）、非アクティブにセットされるか（OCxM=0010）、または反転されます（OCxM=0011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

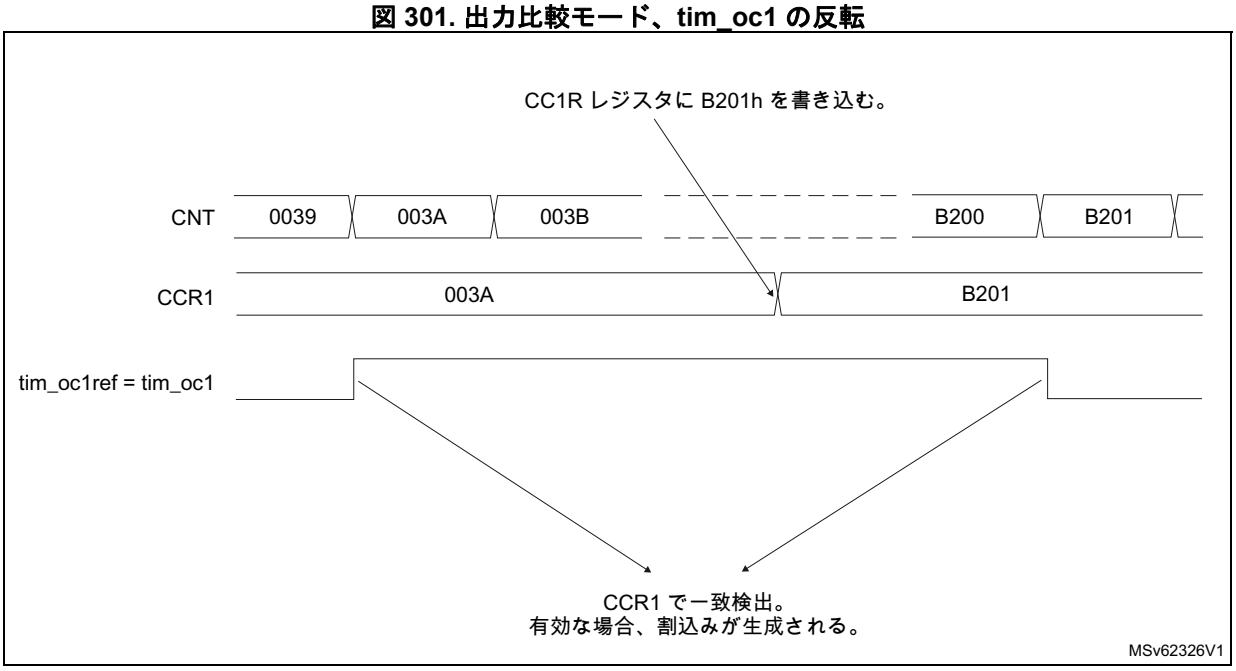
出力比較モードでは、更新イベント UEV は tim_ocxref および tim_ocx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに tim_ocx 出力ピンを反転するには、OCxM ビットに 0011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を図 301 に示します。

図 301. 出力比較モード、tim_oc1 の反転



28.3.13 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“0110”（PWM モード 1）または“0111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（tim_ocx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

tim_ocx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。これは、アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCER レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

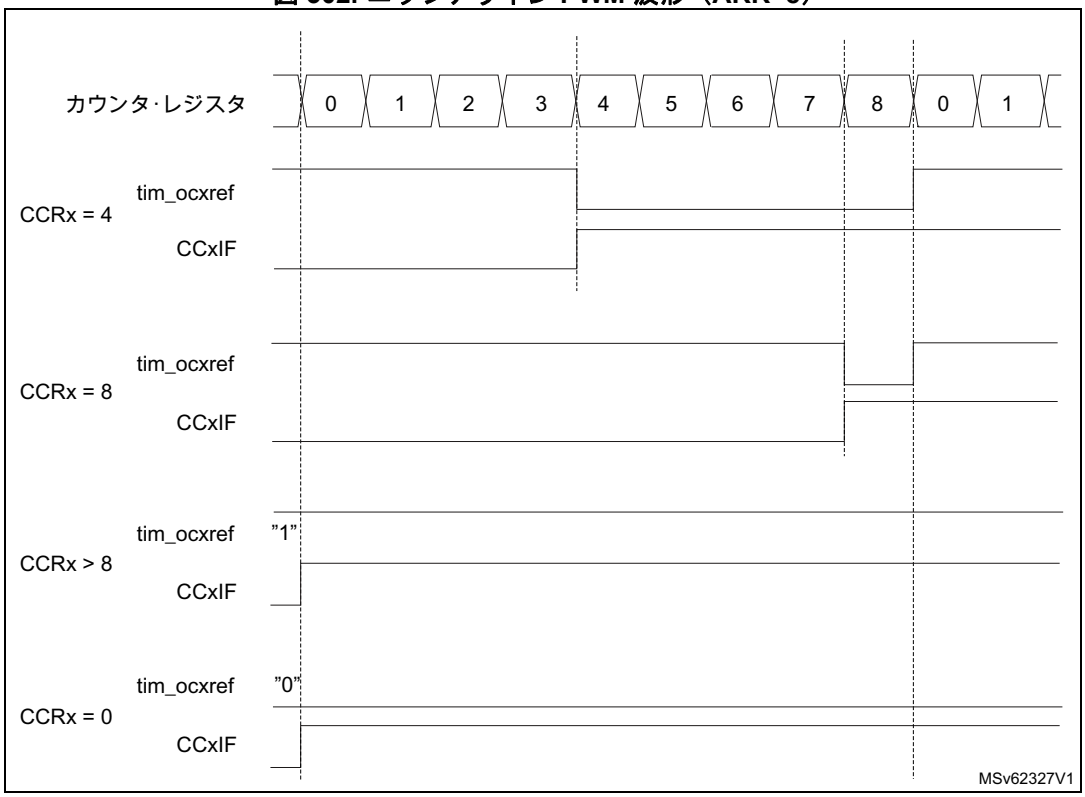
PWM エッジアラインモード

- アップカウント構成

  TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。1064 ページのアップカウントモードを参照してください。

  次の例では、PWM モード 1 を使用しています。PWM 基準信号 tim_ocxref は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、tim_ocxref は“1”に保持されます。比較値が 0 の場合、tim_ocxref は“0”に保持されます。図 302 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 302. エッジアライン PWM 波形（ARR=8）



- ダウンカウント構成

  TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。1068 ページのダウンカウントモードを参照してください。

  PWM モード 1 では、基準信号 tim_ocxref は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、tim_ocxref は“1”に保持されます。このモードでは、0 % の PWM 信号を生成することはできません。

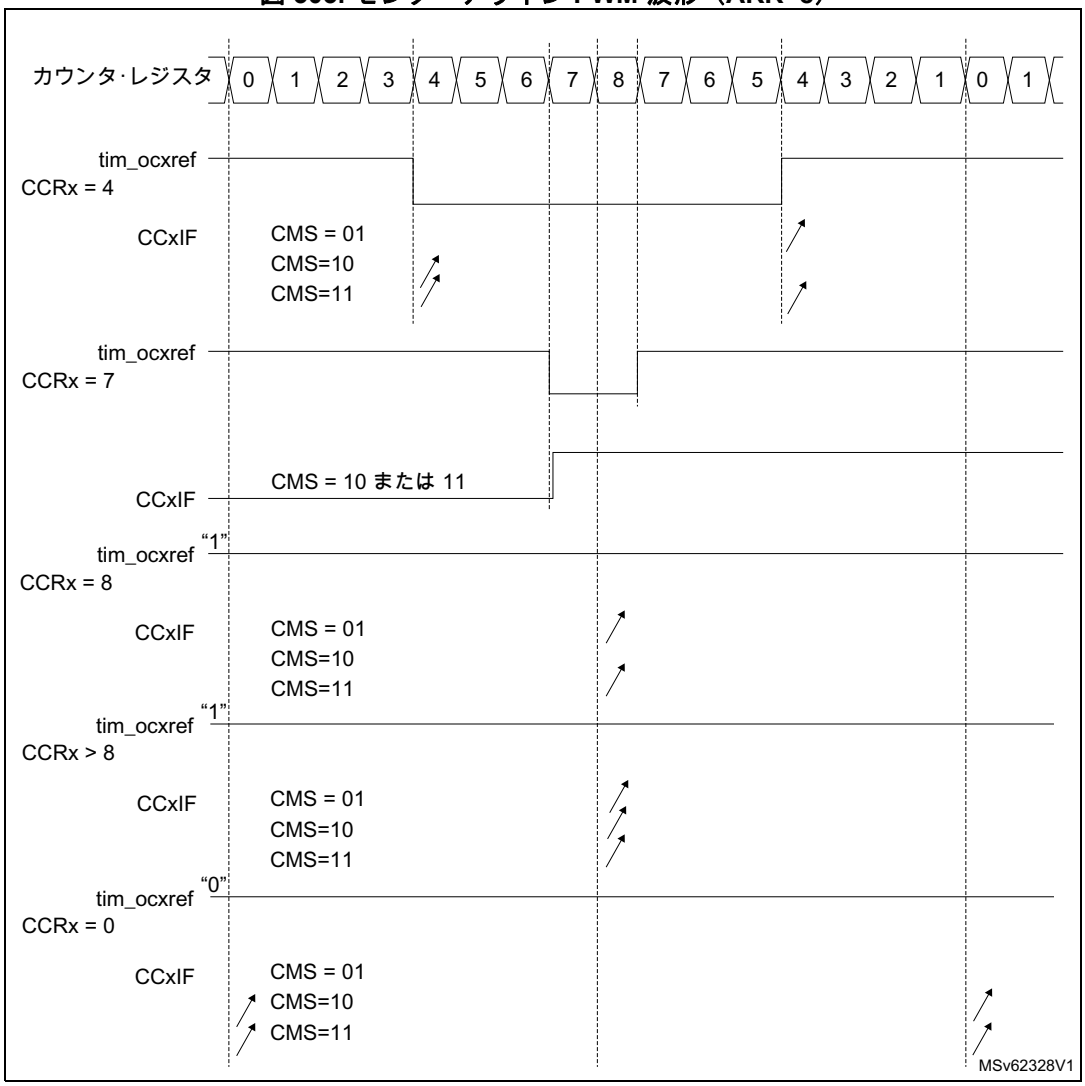
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての設定は、tim_ocxref/tim_ocx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。1072 ページのセンターアラインモード（アップ／ダウンカウント）を参照してください。

図 303 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 303. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント：

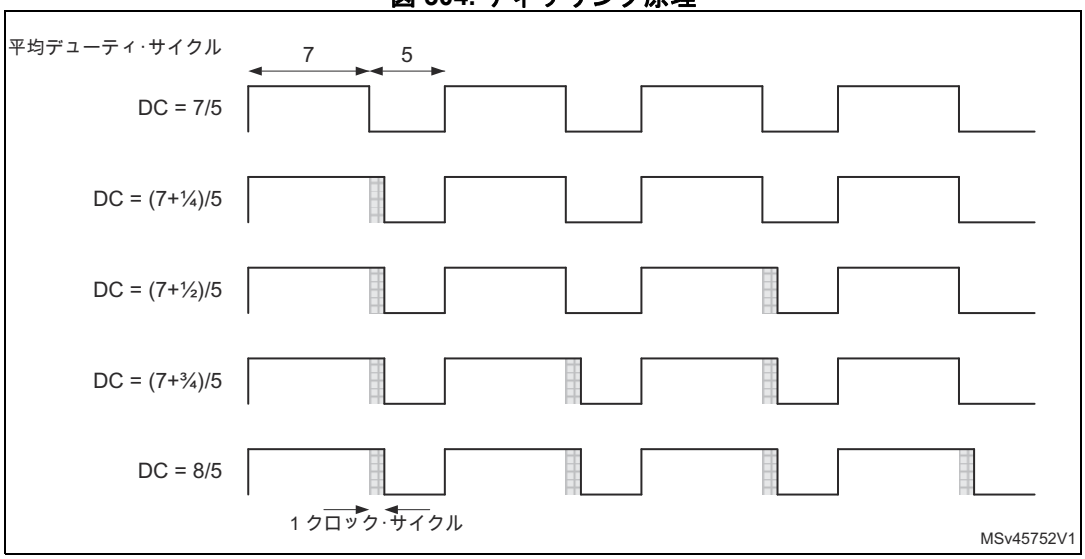
- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書込みを行わないことです。

ディザリングモード

PWM モードの有効分解能は、TIMx_CR1 レジスタの DITHEN ビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）と ARR（PWM 周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16 個の連続した PWM 周期にわたって実際の CCR（または ARR）値をわずかに変更する（1 タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたは PWM 周期を考えると、16 倍の分解能の向上が可能になります。下記の図 304 は、4 つの連続する PWM サイクルに適用されたディザリング原理を示します。

図 304. ディザリング原理

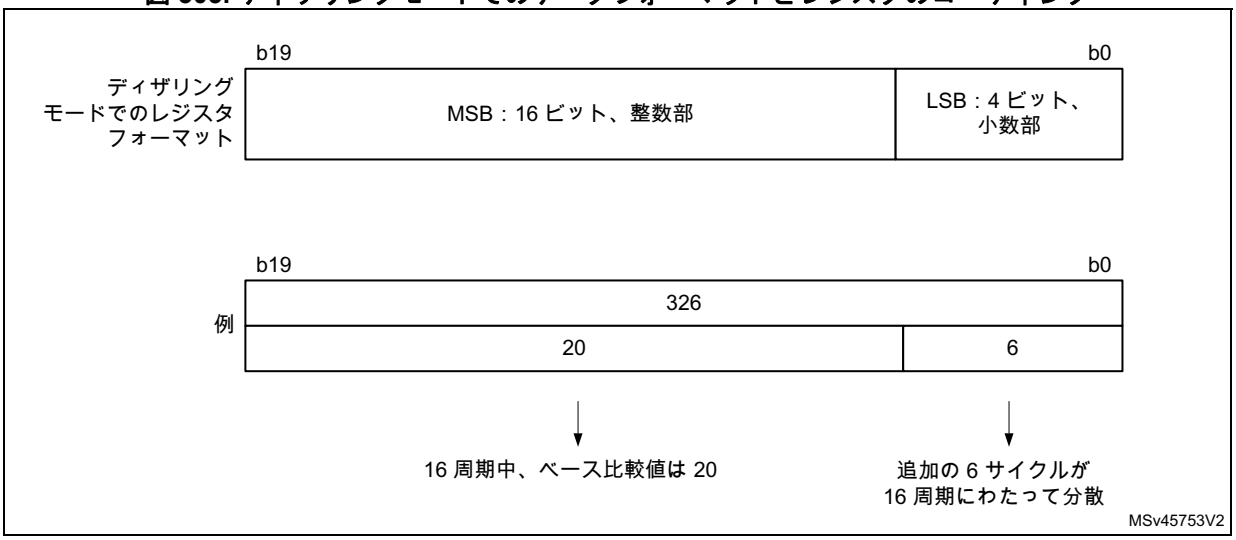


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図 305 を参照）。

- 下位 4 ビットは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット 19:4 に左シフトされ、ベース値用のコーディングになります。

注： **DITHEN ビットがセット／リセットされると、ARR と CCR の値が自動的に更新されます（たとえば、DITHEN = 0 で ARR = 0x05 の場合、DITHEN = 1 で ARR = 0x50 に更新されます）。**
DITHEN ビットをリセットするときは、次の手順に従う必要があります。
1. CEN および ARPE ビットをリセットする必要があります。
2. ARR[3:0] ビットをリセットする必要があります。
3. DITHEN ビットをリセットする必要があります。
4. CCIF フラグをクリアする必要があります。
5. CEN ビットをセットすることができます（最終的に ARPE = 1 とともに）。

図 305. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

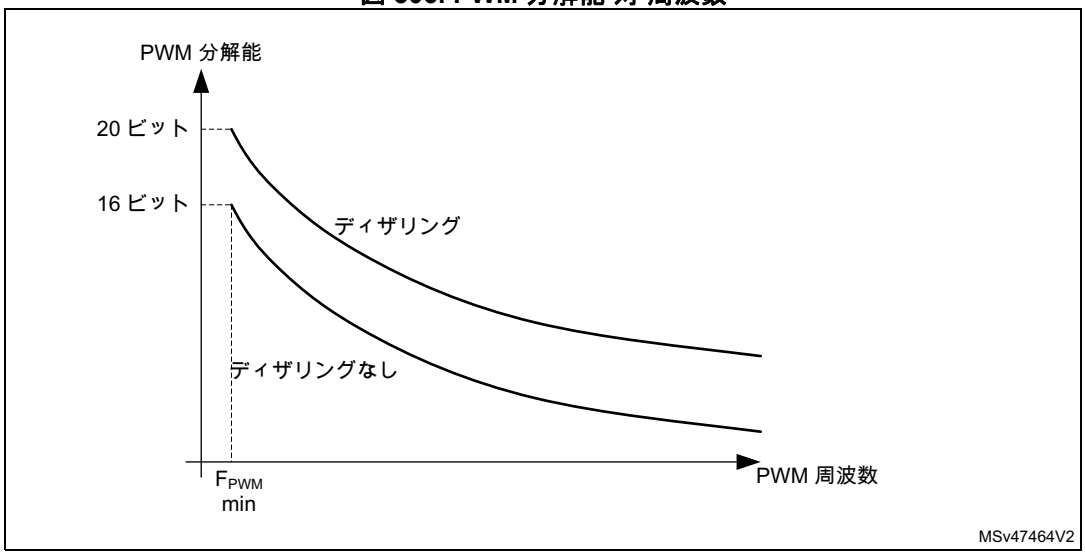
$$\text{ディザリングモード無効時：} F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモード有効時：} F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARR および TIMxCCRy の最大値は、ディザリングモードでは 0xFFFEF に制限されています（整数部では 65534、ディザリング部分では 15 に対応します）。**

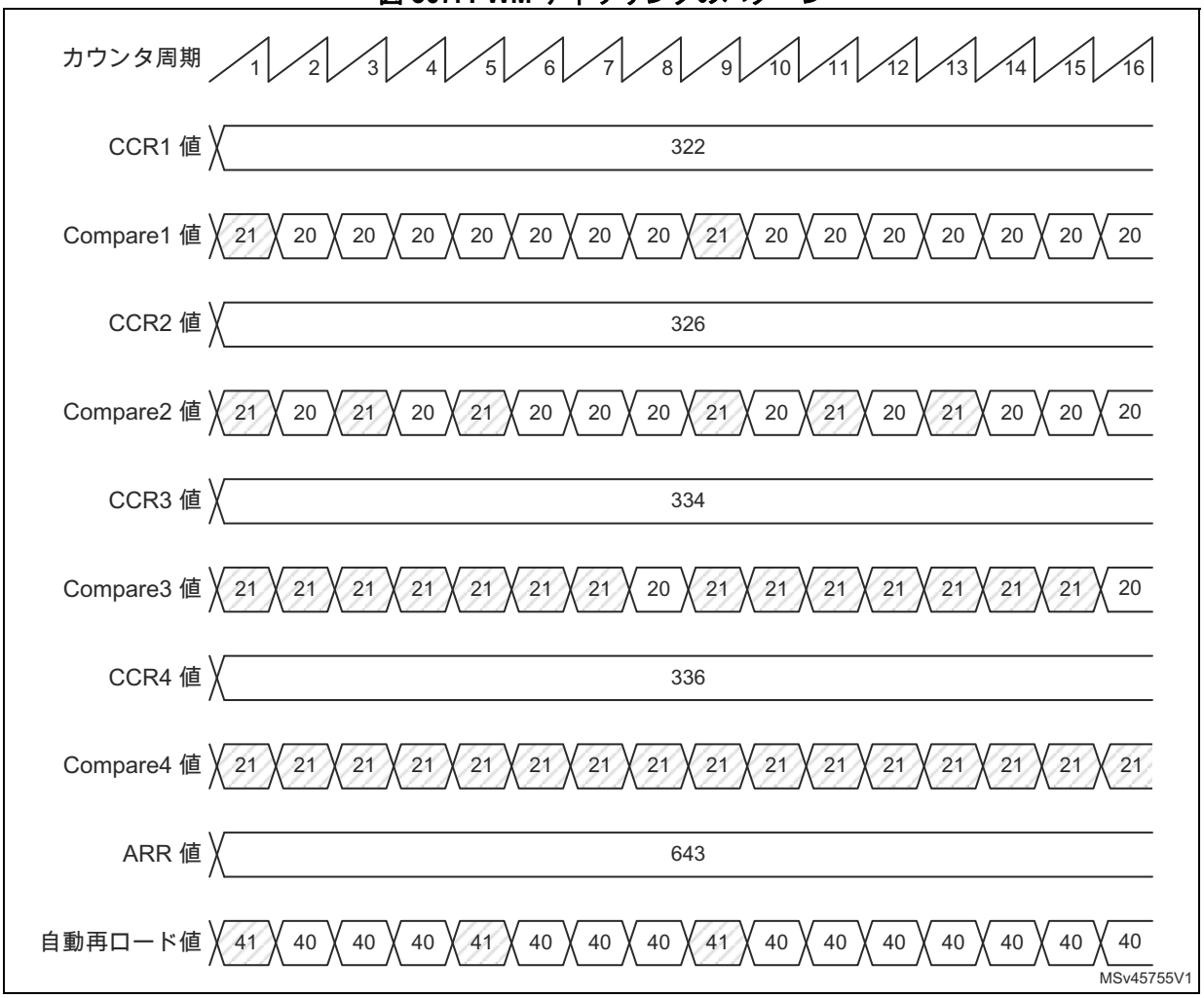
下記の図 306 に示すように、PWM 周波数に関係なく、ディザリングモードによって PWM 分解能を増加することができます。

図 306. PWM 分解能 対 周波数



下記の図 307 で説明するように、デューティサイクルや周期の変更は、連続した 16 周期にわたって広がります。

図 307. PWM ディザリングのパターン



自動再ロード値と比較値の増分は、下記の表 257 に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表 257. CCR および ARR レジスタの変更のディザリングパターン

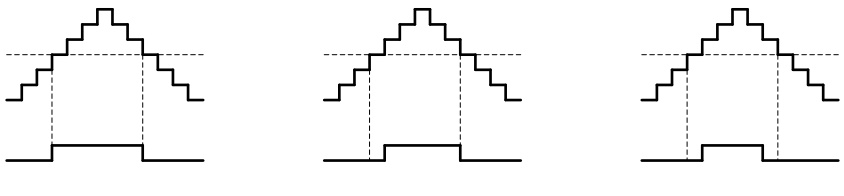
LSB 値	PWM 周期															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-

表 257. CCR および ARR レジスタの変更のディザリングパターン （続き）

LSB 値	PWM 周期															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

ディザリングモードはセンターアライン PWM モード（TIMx_CR1 レジスタの CMS ビットが“00”に等しくないとき）でも使用できます。この場合、ディザリングパターンは、以下の図 308 に示すように、アップおよびダウンカウントフェーズを考慮して、8 つの連続した PWM 周期にわたって適用されます。

図 308. センターアライン PWM モードでのデューティサイクルへのディザリングの影響



下記の表 258 は､センターアライン PWM モードでどのようにディザリングパターンが追加されるかを示しています。

表 258. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン

LSB 値	PWM 周期															
	1		2		3		4		5		6		7		8	
	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-

表 258. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン （続き）

LSB 値	PWM 周期															
	1		2		3		4		5		6		7		8	
	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

28.3.14 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- tim_oc1refc（または tim_oc2refc）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- tim_oc3refc（または tim_oc4refc）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

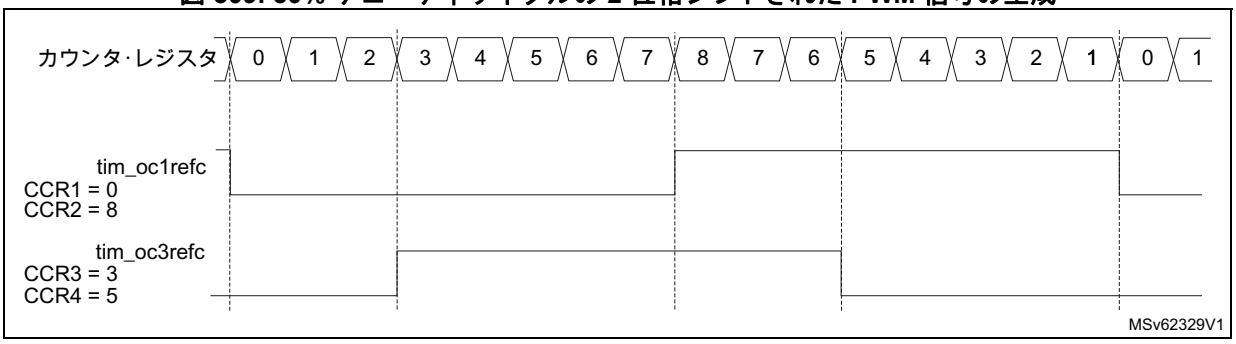
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの tim_ocx 出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

特定のチャネルが非対称の PWM チャネルとして使用されると、その相補チャネルも使用できます。たとえば、tim_oc1refc 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の tim_oc2ref 信号、または非対称 PWM モード 1 の結果として得られる tim_oc2refc 信号を出力できます。

図 309 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 2 として設定されます）。これにより、デッドタイムジェネレータとともにフルブリッジ位相シフト DC-DC コンバータを制御できます。

図 309. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



28.3.15 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 tim_ocxrefc は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- tim_oc1refc（または tim_oc2refc）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- tim_oc3refc（または tim_oc4refc）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの tim_ocx 出力）。

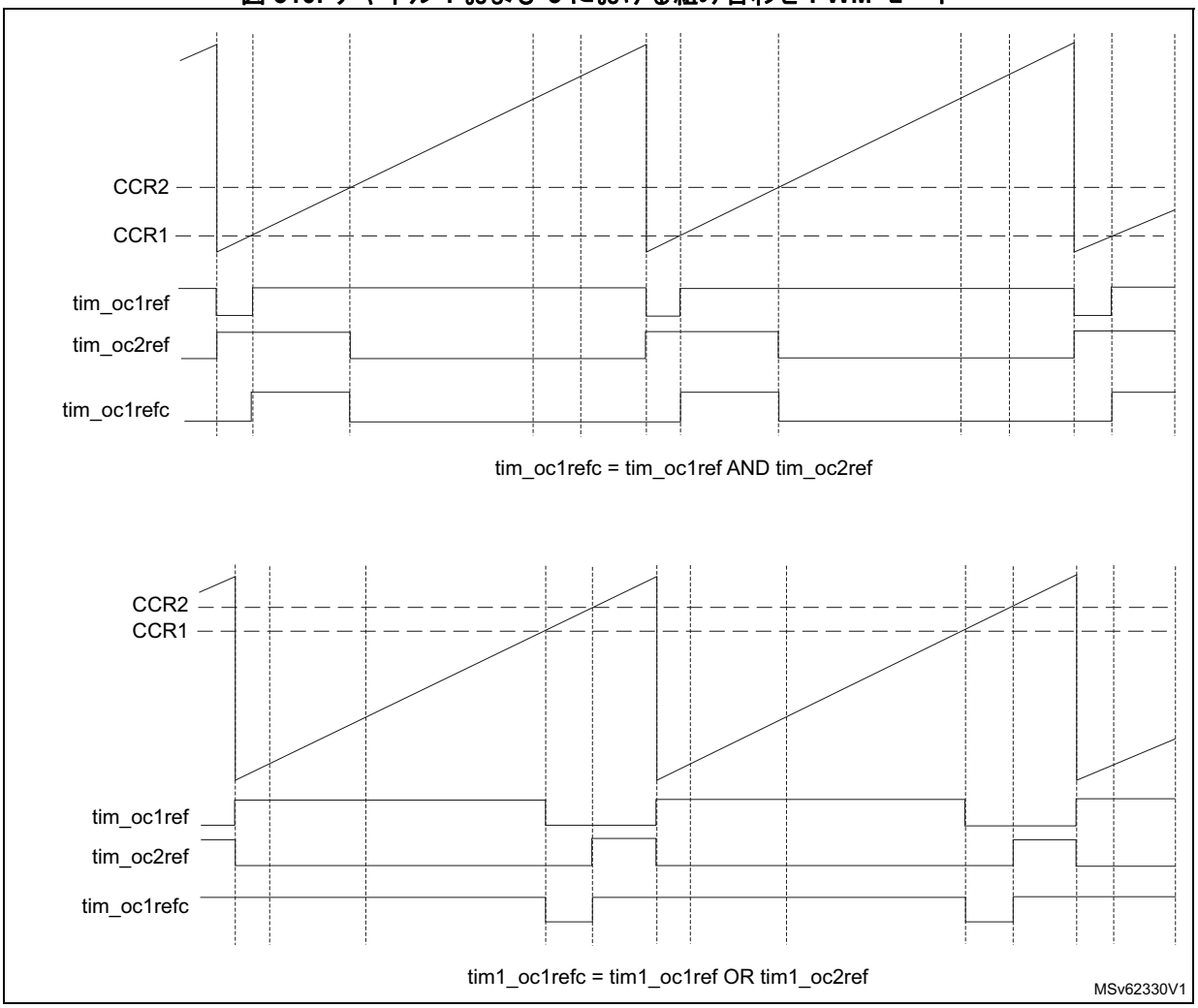
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 310 は、次の設定で取得可能な組み合わせ PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 310. チャネル 1 および 3 における組み合わせ PWM モード



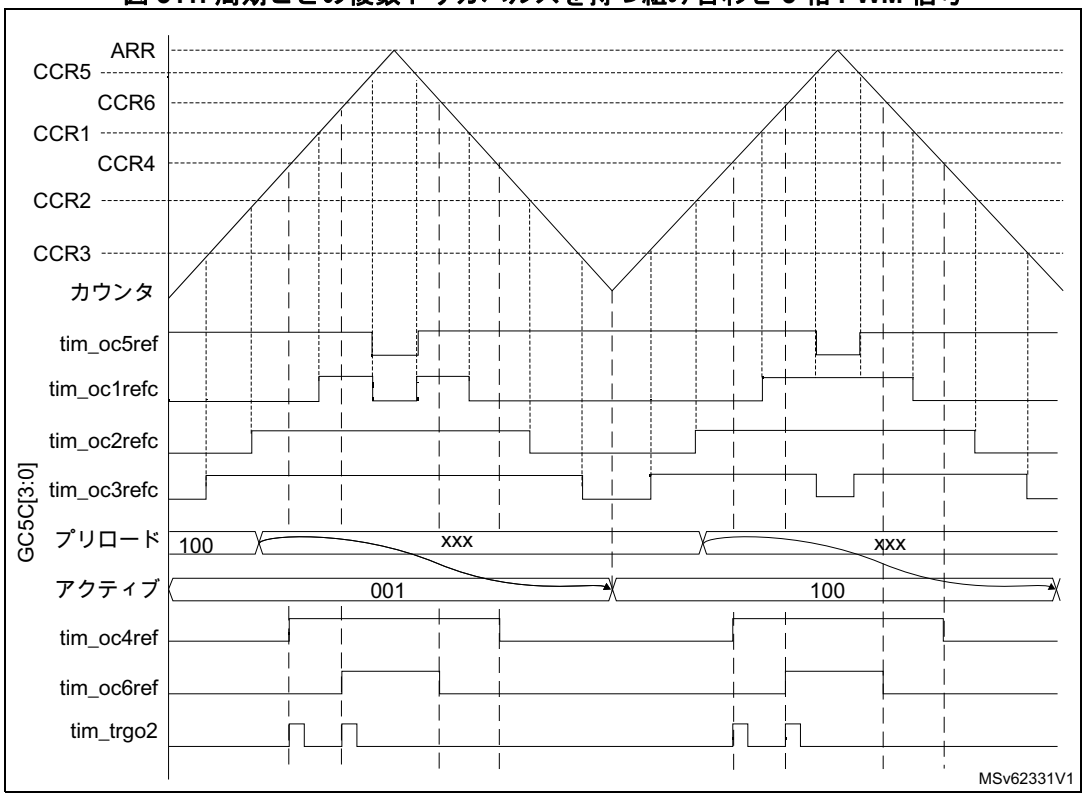
28.3.16 組み合わせ 3 相 PWM モード

組み合わせ 3 相 PWM モードでは、パルスの中間で論理積を取った単一のプログラム可能な信号とともに 1 つから 3 つのセンターアライン PWM 信号を生成できます。結果として得られる組み合わせ信号の定義には、tim_oc5ref 信号が使用されます。TIMx_CCR5 の 3 ビット GC5C[3:1] では、tim_oc5ref を組み合わせる基準信号を選択できます。結果として得られる信号 tim_ocxrefc は、2 つの PWM 基準信号の AND による論理結合から生成されます。

- GC5C1 がセットされると、tim_oc1refc は TIMx_CCR1 および TIMx_CCR5 によって制御されます。
- GC5C2 がセットされると、tim_oc2refc は TIMx_CCR2 および TIMx_CCR5 によって制御されます。
- GC5C3 がセットされると、tim_oc3refc は TIMx_CCR3 および TIMx_CCR5 によって制御されます。

組み合わせ 3 相 PWM モードは、少なくとも3 ビット GC5C[3:1] の 1 つをセットすることでチャネル 1 から 3 で個別に選択できます。

図 311. 周期ごとの複数トリガパルスを持つ組み合わせ 3 相 PWM 信号



tim_trgo2 波形は、特定の 3 相 PWM 信号に ADC を同期できる様子を示します。詳細については、セクション 28.3.31 : ADC トリガを参照してください。

28.3.17 相補出力とデッドタイム挿入

高機能制御タイマ（TIM1/TIM8/TIM20）は、2 つの相補信号を出力して、出力時のスイッチオンおよびスイッチオフを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 tim_ocx または相補 tim_ocxn）の極性は出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書込みによって行います。

相補信号 tim_ocx および tim_ocxn は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMx_CR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、表 266 : 1170 ページのブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビットを参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

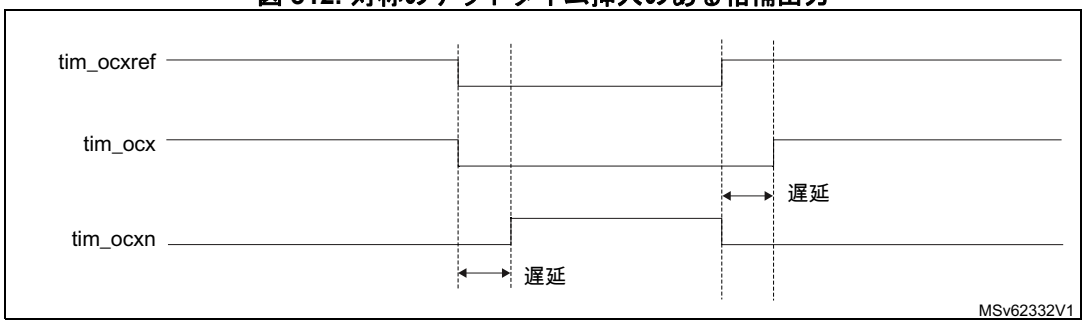
デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 tim_ocxref から tim_ocx と tim_ocxn の 2 つの出力を生成します。tim_ocx と tim_ocxn がアクティブハイの場合、

- tim_ocx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- tim_ocxn 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（tim_ocx または tim_ocxn）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 tim_ocxref との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 312. 対称のデッドタイム挿入のある相補出力



TIMx_DTR2 の DTAE ビットによって、図 313 に示すように、デッドタイム値を、基準信号の立ち上がりエッジと立ち下がりエッジに対して変えることができます。

非対称モード（DTAE = 1）では、立ち上がりエッジ基準デッドタイムは TIMx_BDTR レジスタの DTG[7:0] ビットフィールドによって定義され、立ち下がりエッジ基準は TIMx_DTR2 レジスタの DTGF[7:0] ビットフィールドによって定義されます 。DTAE ビットは、カウンタを有効にする前に書き込む必要があり、CEN=1 の間は変更してはなりません。

プリロードメカニズムを使用して、PWM 運転中にデッドタイム値を動作中に更新することができます。デッドタイムビットフィールド DTG[7:0] と DTGF[7:0] は、TIMX_DTR2 レジスタで DTPE ビットがセットされるとプリロードされます。プリロード値は、次の更新イベントでアクティブレジスタにロードされます。

注： **カウンタが有効なときに DTPE ビットを有効にした場合、最後の更新以降に書き込まれた新しい値は破棄され、以前の値が使用されます。**

図 313. 非対称のデッドタイム

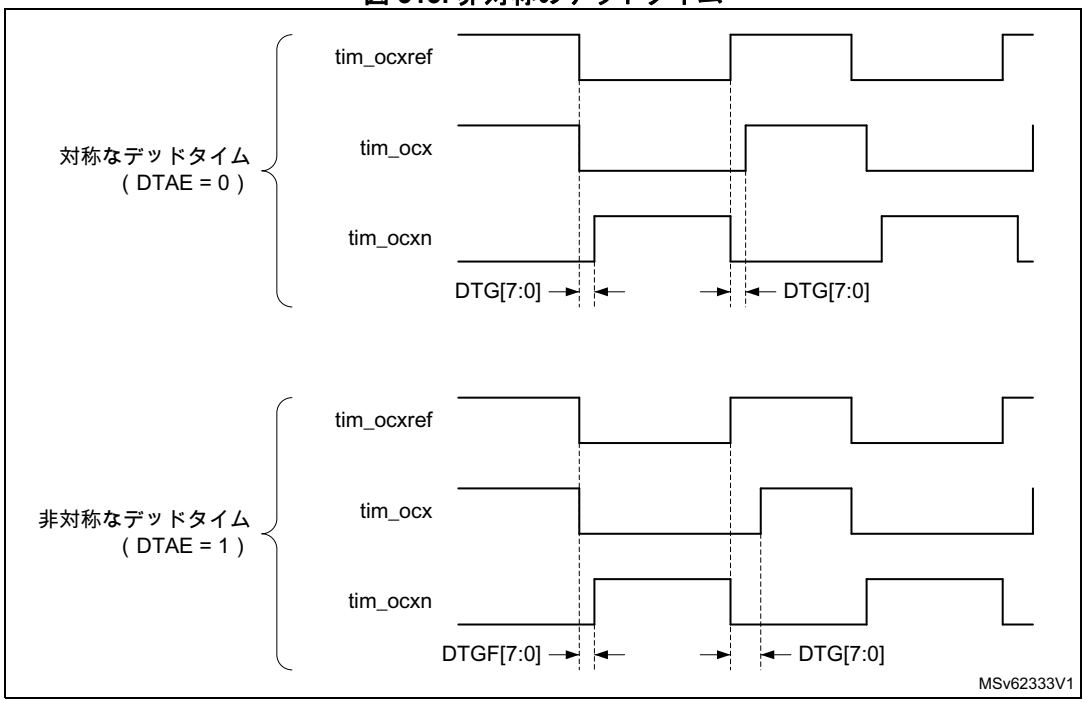


図 314. 負のパルスより長い遅延があるときのデッドタイムの波形

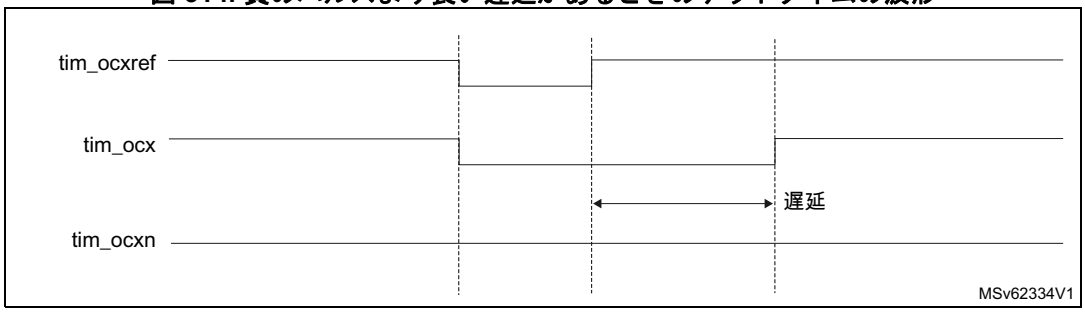
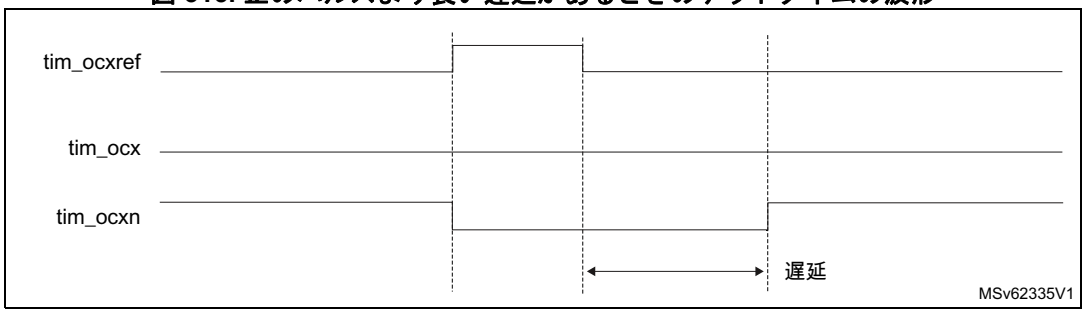


図 315. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、セクション 28.6.20：TIMx ブレークおよびデッドタイム･レジスタ（TIMx_BDTR）（x = 1、8、20）を参照してください。

tim_ocxref 信号の tim_ocx または tim_ocxn へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを設定することによって、tim_ocxref 信号を tim_ocx 出力または tim_ocxn 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **tim_ocxn のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、tim_ocxref がハイレベルになるとすぐにアクティブになります。たとえば、CCxNP=0 の場合は、tim_ocxn=tim_ocxref になります。他方、tim_ocx と tim_ocxn の両方が有効なときには（CCxE=CCxNE=1）、tim_ocxref がハイになると tim_ocx はアクティブになり、tim_ocxn は補信号となり、tim_ocxref がローのときにアクティブになります。**

28.3.18 ブレーク機能の使用

ブレーク機能の目的は、タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。2 つのブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。出力の遮断をトリガするために、いくつかの内部 MCU イベントを選択することも可能です。

このブレークには、2 つのチャネルがあります。システムレベル障害（クロック障害、ECC/パリティ·エラーなど）とアプリケーション障害（入力ピンおよび内蔵コンパレータ）の両方を集め、出力をデッドタイムの持続時間経過後に事前定義されたレベル（アクティブまたはインアクティブ）に強制できるブレーク·チャネル。アプリケーション障害のみを含み、出力をインアクティブ状態に強制できるブレーク 2 チャネル。

ブレーク時の出力有効信号および出力レベルは、いくつかの制御ビットに依存しています。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に tim_ocx 出力と tim_ocxn 出力を両方ともアクティブレベルにセットすることはできません。詳細については、表 266：1170 ページのブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビットを参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットおよび BK2E ビットをセットすることによって有効にできます。ブレーク入力の極性は、同じレジスタの BKP および BK2P ビットを設定することによって選択できます。BKEx と BKPx は、同時に変更できます。BKEx および BKPx ビットが書き込まれるとき、書込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 に設定された場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

ブレーク（tim_brk）チャネルのソースは以下のいずれかです。

- TIMx_BKIN ピンの 1 つに接続された外部ソース（GPIO オルタネート機能選択レジスタの設定による選択。極性選択およびオプションのデジタル･フィルタリングあり）
- 内部ソース：
  - tim_brk_cmpx 入力から着信するもの（製品固有の実装についてはセクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照）
  - システムブレークリクエストから着信するもの（製品固有の実装についてはセクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照）

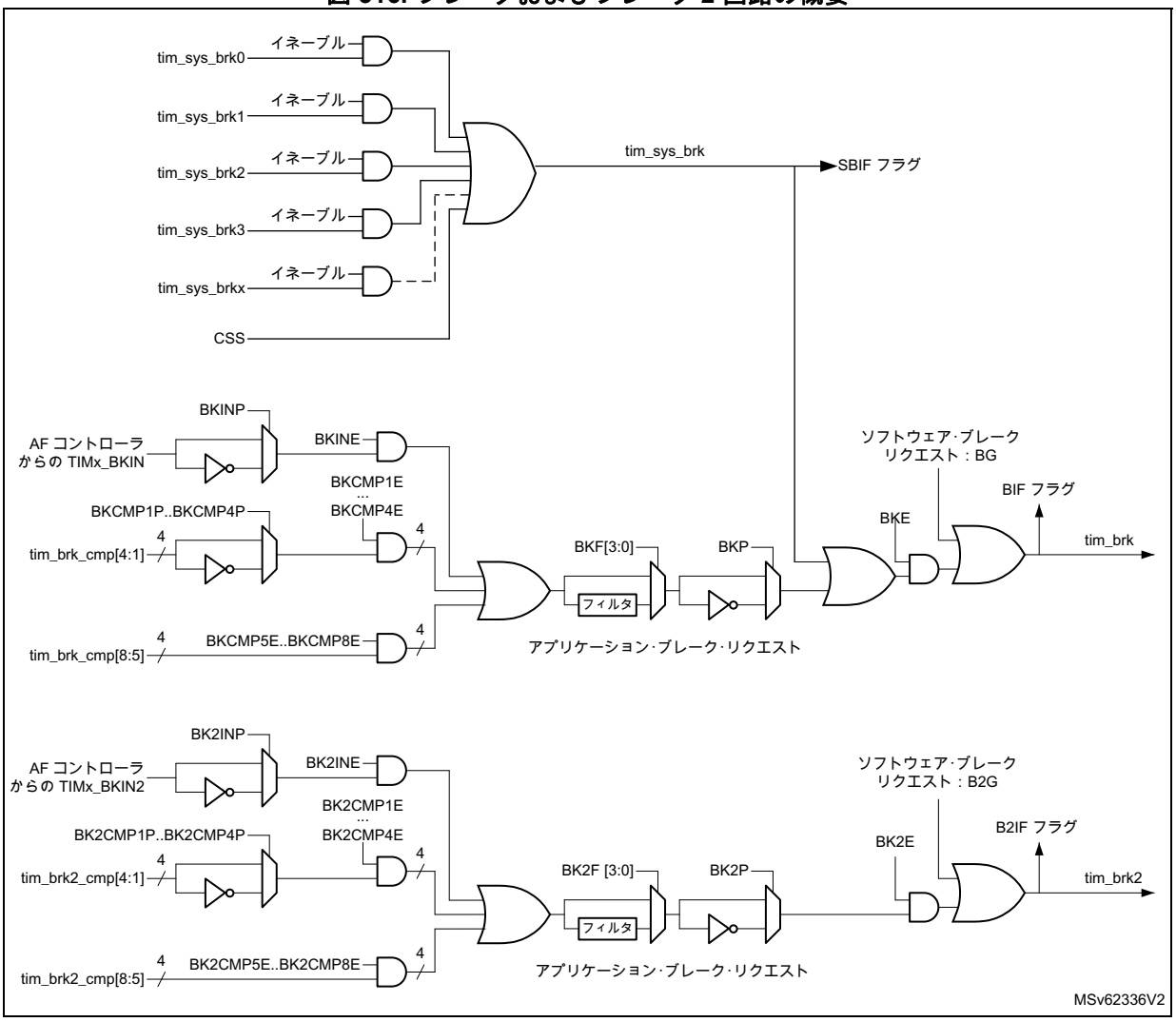
ブレーク 2（tim_brk2）のソースは以下のいずれかです。

- TIMx_BKIN2 ピンの 1 つに接続された外部ソース（GPIO オルタネート機能選択レジスタの設定による選択。極性選択およびオプションのデジタル･フィルタリングあり）
- tim_brk2_cmpx 入力から供給される内部ソース（製品固有の実装については セクション 28.3.2: TIM1/TIM8/TIM20 ピンおよび内部信号を参照）

ブレークイベントは、TIMx_EGR レジスタで BG および B2G ビットを使用して、ソフトウェアによって生成することもできます。

以下の図 316 のとおり、すべてのソースはタイマ tim_brk または tim_brk2 入力に入る前に論理和がとられます。

図 316. ブレークおよびブレーク 2 回路の概要



注： **非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。**

ブレークが 1 つ発生すると（ブレーク入力の 1 つで選択されたレベル）、

- MOE ビットは非同期にクリアされ、出力は、インアクティブ状態またはアイドル状態になるか、GPIO コントローラへの制御が解除されます（OSSI ビットで選択）。これは、MCU オシレータがオフの場合に有効です。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI=0 の場合、タイマは出力の制御（GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままです。
- 相補出力が使用されているときには：
  - 出力は、まずインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、tim_ocx と tim_ocxn を同時にアクティブレベルに駆動することはで

きません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 tim_ker_ck クロックサイクル）。

– OSSI=0 の場合、タイマは出力の制御（ハイインピーダンス状態を強制する GPIO コントローラによって引き継がれた）を解除し、そうでない場合、イネーブル出力はハイのままか、CCxE または CCxNE ビットのどちらかがハイになったときにハイになります。

- ブレーク状態フラグ（TIMx_SR レジスタの SBIF、BIF、および B2IF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合、割込みが生成されます。TIMx_DIER レジスタのBDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合、MOE ビットは次の更新イベント（UEV）で再び自動的にセットされます。たとえば、これを使用してレギュレーションを行うことができます。そうでない場合、MOE ビットはアプリケーションが再び“1”をセットするまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

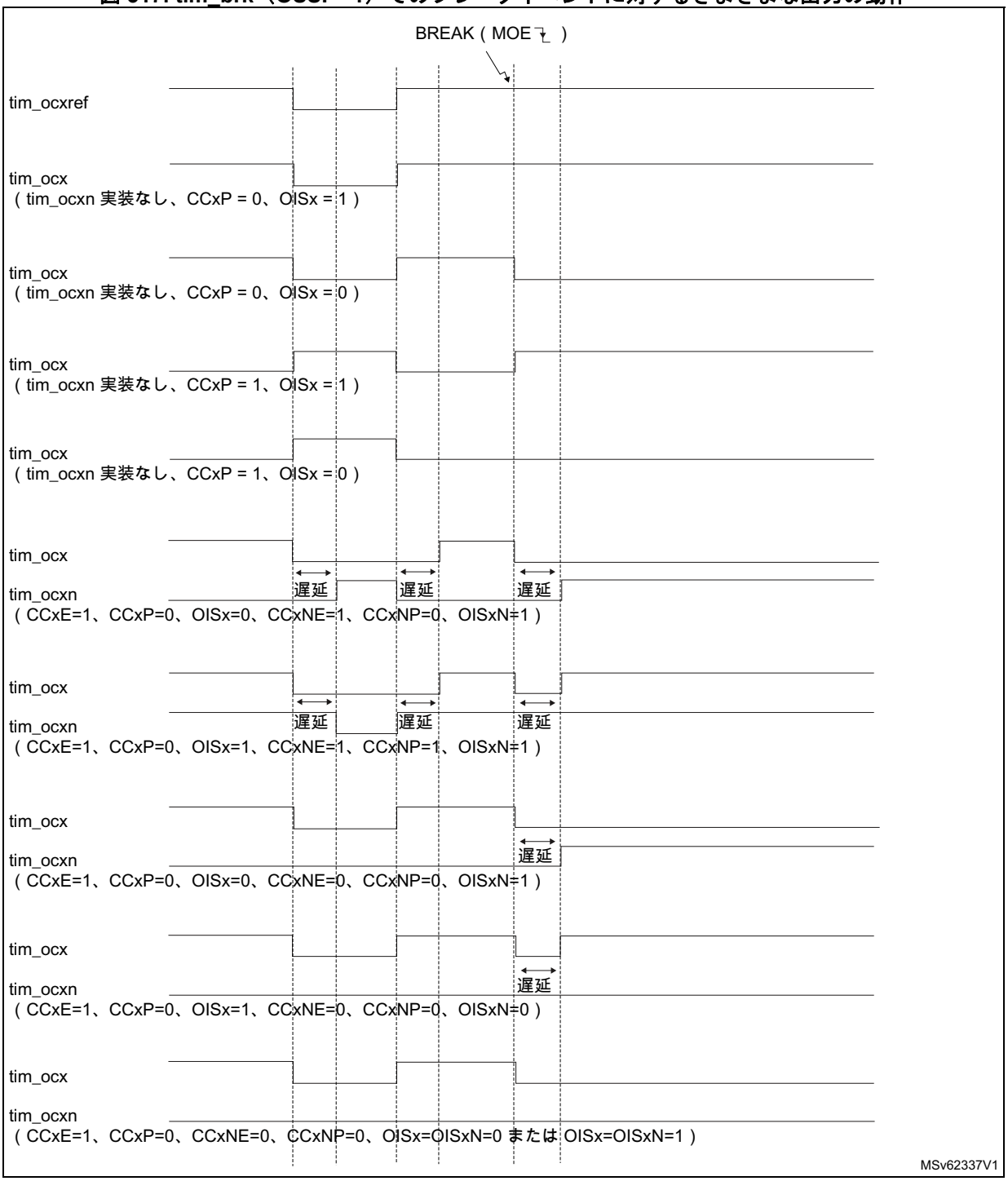
注： **AOE がセットされた状態で MOE が CPU によってリセットされると、出力はアイドル状態になり、OSSI の値に応じて非アクティブ・レベルまたはハイ・インピーダンスに強制的に設定されます。MOE と AOE の両方が CPU によってリセットされた場合は、出力が無効化され、TIMx_CR2 レジスタの OISx ビットにプログラムされたレベルに駆動されます。**

注： **ブレーク入力はアクティブでオン・レベルです。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF および B2IF をクリアできません。**

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これにより、いくつかのパラメータ（デッドタイムの長さ、tim_ocx/tim_ocxn 極性、無効時の状態、OCxM 構成、ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、3 レベルの保護を選択することができます。セクション 28.6.20 : TIMx ブレークおよびデッドタイム・レジスタ（TIMx_BDTR）（x = 1、8、20）を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 317 に、ブレークに対する出力の動作例を示します。

図 317. tim_brk（OSSI = 1）でのブレークイベントに対するさまざまな出力の動作



2 つのブレーク入力は、タイマ出力で異なる動作を示します。

- tim_brk 入力は、無効化（インアクティブ状態）するか、PWM 出力を強制的に事前定義した安全な状態に移行できます。
- tim_brk2 は、PWM 出力の無効化（インアクティブ状態）のみ可能です。

表 259 に示すように、tim_brk の優先順位は tim_brk2 入力よりも高いです。

注： **tim_brk2 は OSSR = OSSI = 1 の場合にのみ使用してください。**

表 259. タイマ出力と tim_brk/tim_brk2 入力の動作

tim_brk	tim_brk2	タイマ出力状態	通常の使用例	
			tim_ocxn 出力（ローサイドスイッチ）	tim_ocx 出力（ハイサイドスイッチ）
アクティブ	X	– 非アクティブから強制される出力状態（デッドタイム後） – OSSI = 0 の場合、出力は無効（GPIO ロジックが制御を引き継ぐ）	デッドタイム挿入後 ON	OFF
非アクティブ	アクティブ	非アクティブ	OFF	OFF

図 318 は、tim_brk および tim_brk2 入力で信号がアクティブな場合の、tim_ocx および tim_ocx 出力の動作例を示しています。この場合、両方の出力がアクティブハイ極性になります（TIMx_CCER レジスタの CCxP = CCxNP = 0）。

図 318. tim_brk および tim_brk2 のアサート後の PWM 出力状態（OSSI=1）

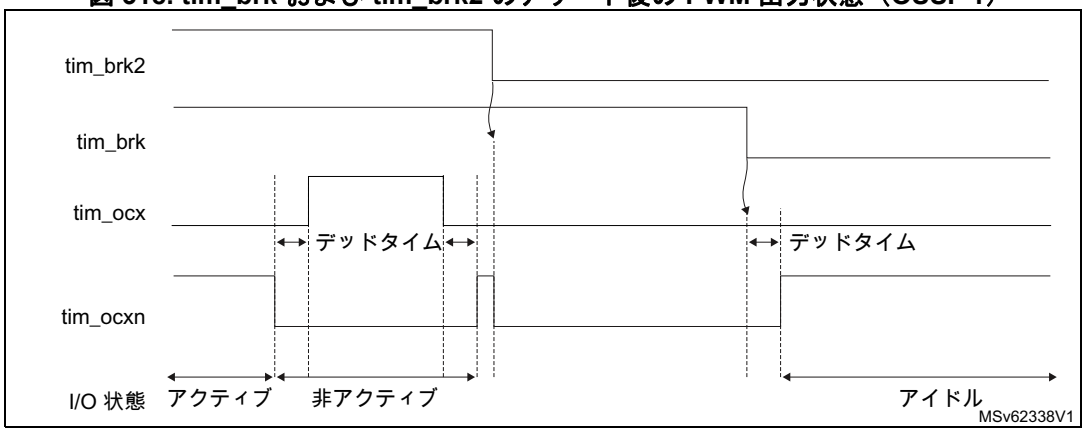
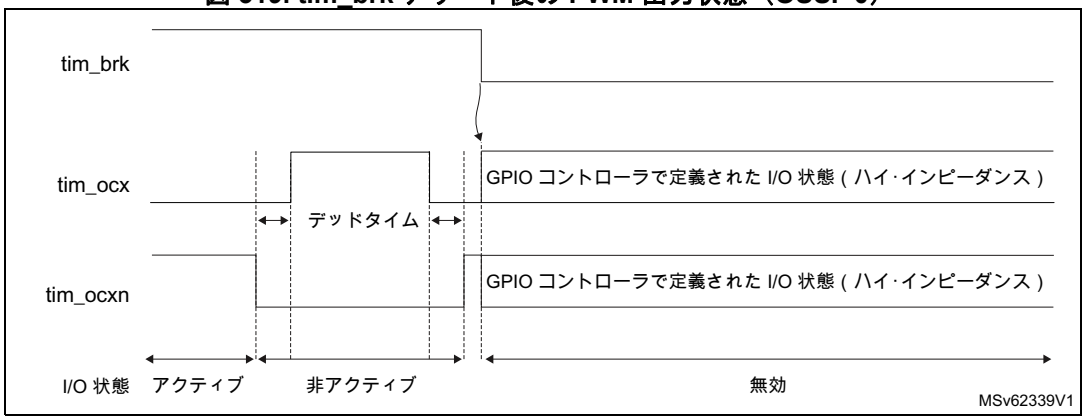


図 319. tim_brk アサート後の PWM 出力状態（OSSI=0）



28.3.19 双方向ブレーク入力

TIM1/TIM8/TIM20 には、図 320 に示すとおり、双方向ブレーク I/O が搭載されています。

これによって、以下の機能がサポートされます。

- 入力および出力ステータス·ピンどちらにもなるユニークなピンを持ち、外部のマイクロコントローラやゲート·ドライバに障害を通知するために使用できるボード·レベルのグローバル·ブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、ユニークなブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインソース

tim_brk および tim_brk2 入力は、TIMxBDTR レジスタの BKBID および BK2BID ビットを使用して双方向モードに設定されます。BKBID プログラミングビットは、TIMxBDTR レジスタの LOCK ビットを使用して読出し専用モードにロックできます（LOCK レベル 1 以上）。

双方向モードは、tim_brk と tim_brk2 入力の両方で使用でき、I/O をアクティブロー極性でオープンドレインモードに設定する必要があります（BKINP、BKP、BK2INP、BK2P ビットを使用）。システム（例：CSS）、オンチップペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローとなります。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

ブレークソフトウェアイベント（BG および B2G）も、ブレーク I/O を強制的に 0 にして、タイマがブレーク状態に移行されたことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BKE または B2KE = 1）にのみ有効です。ソフトウェアブレークイベントが BKE または B2KE = 0 で生成された場合、出力が安全な状態になり、ブレークフラグはセットされますが、TIMx_BKIN および TIMx_BKIN2 I/O には効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRM（BK2DSRM）ビットが 1 にセットされると、この機能によりブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM（BK2DSRM）ビットがセットされており、オーブンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM 出力の再開を防ぐことができます。
- 出力が有効になっている（MOE ビットがセットされている）限り、BKDSRM（BK2DSRM）ビットでブレーク保護を解除できません（表 260 を参照）。

表 260. ブレーク保護解除条件

MOE	BKBID（BK2BID）	BKDSRM（BK2DSRM）	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

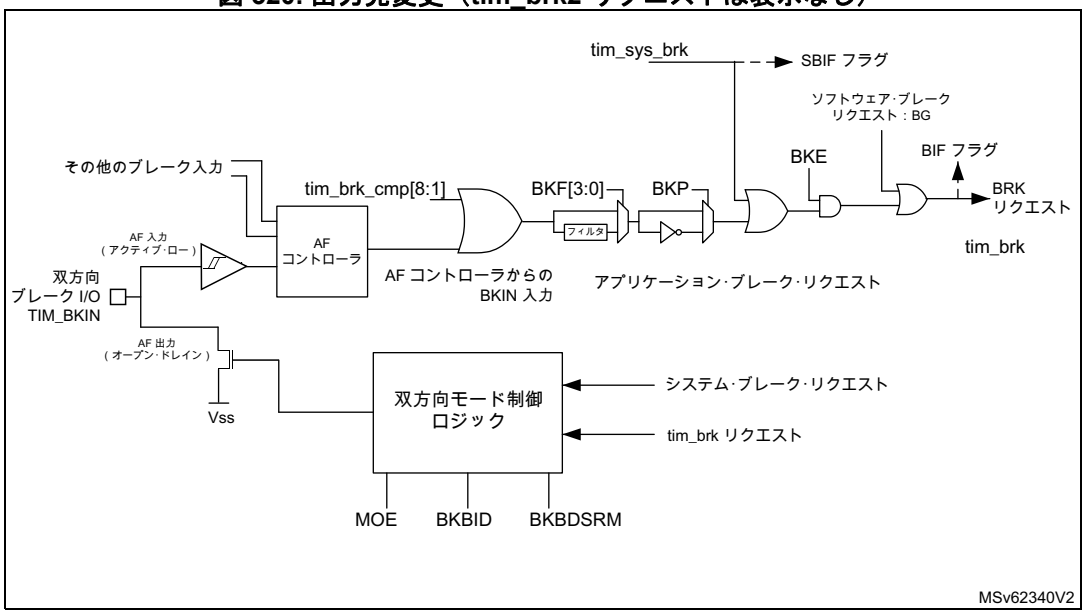
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレーク（ブレーク 2）イベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRM（BK2DSRM）ビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM（BK2DSRM）をハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOE ビットをセットして、PWM 出力を再び有効にできます。

図 320. 出力先変更（tim_brk2 リクエストは表示なし）

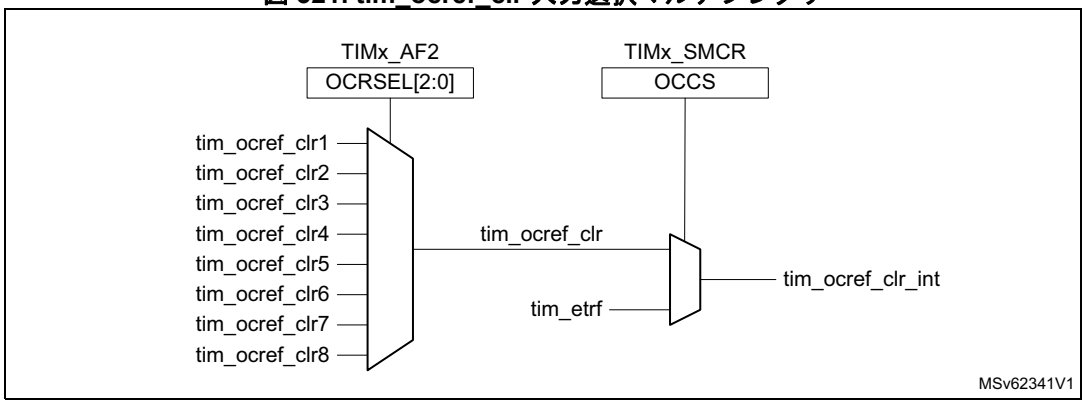


28.3.20 外部イベントによる tim_ocxref 信号のクリア

特定のチャネルの tim_ocxref 信号は tim_ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。tim_ocxref は、次の更新イベント（UEV）が発生するまでローのままになります。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。tim_ocref_clr_int 入力は、TIMx_SMCR レジスタで OCCS ビットを設定することで、tim_ocref_clr 入力と tim_etrf（フィルタ後の tim_etr_in）の間で選択できます。

tim_ocref_clr 入力は、以下の図 321 に示すように、TIMx_AF2 レジスタの OCRSEL[2:0] ビットフィールドを使用して複数の入力から選択できます。製品で利用できるソースの一覧については、セクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

図 321. tim_ocref_clr 入力選択マルチプレクサ

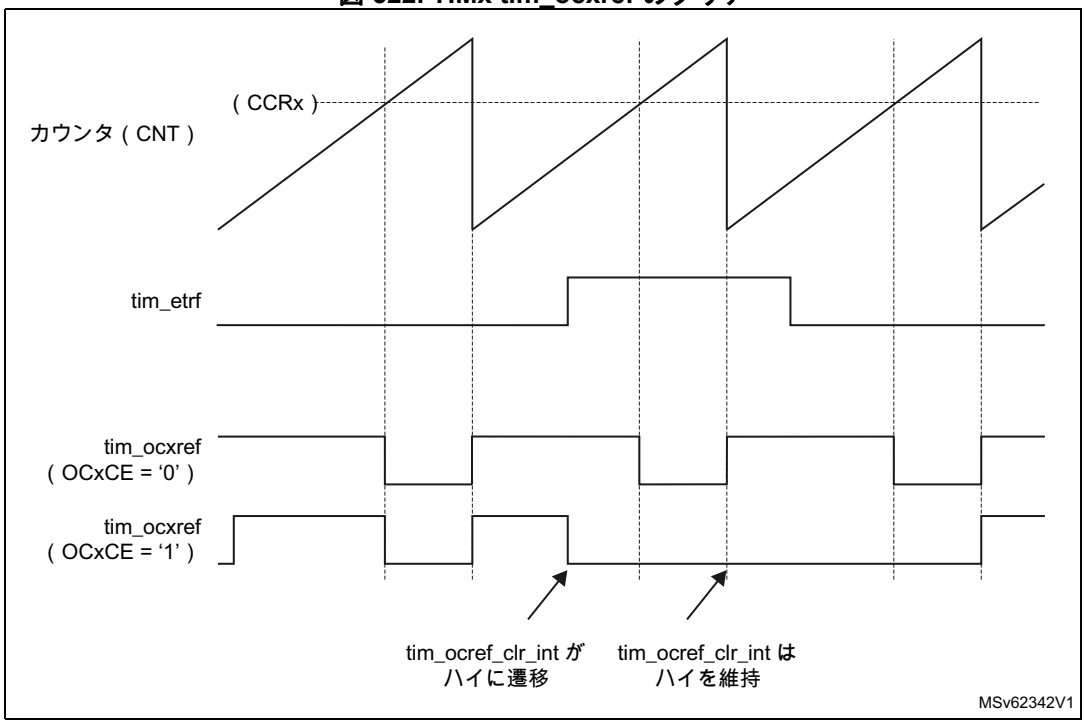


tim_etrf が選択された場合、tim_etr_in は次のように設定する必要があります。

1. 外部トリガプリスケーラはオフ状態に維持する必要があります（TIMx_SMCR レジスタの ETPS[1:0] ビットを“00”にセット）。
2. 外部クロックモード 2 を無効にします（TIMx_SMCR レジスタの ECE ビットを”0”にセット）。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて（トリガに接続されたソースの極性、およびフィルタを使ってノイズを除去する最終的なニーズに従って）設定できます。

図 322 に、イネーブルビット OCxCE の両方の値について、tim_etrf 入力がハイレベルになったときの tim_ocxref 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 322. TIMx tim_ocxref のクリア



注： **100% デューティサイクルの PWM の場合 （CCRx > ARR の場合）、次のカウンタオーバーフローで tim_ocxref が再度有効になります。**

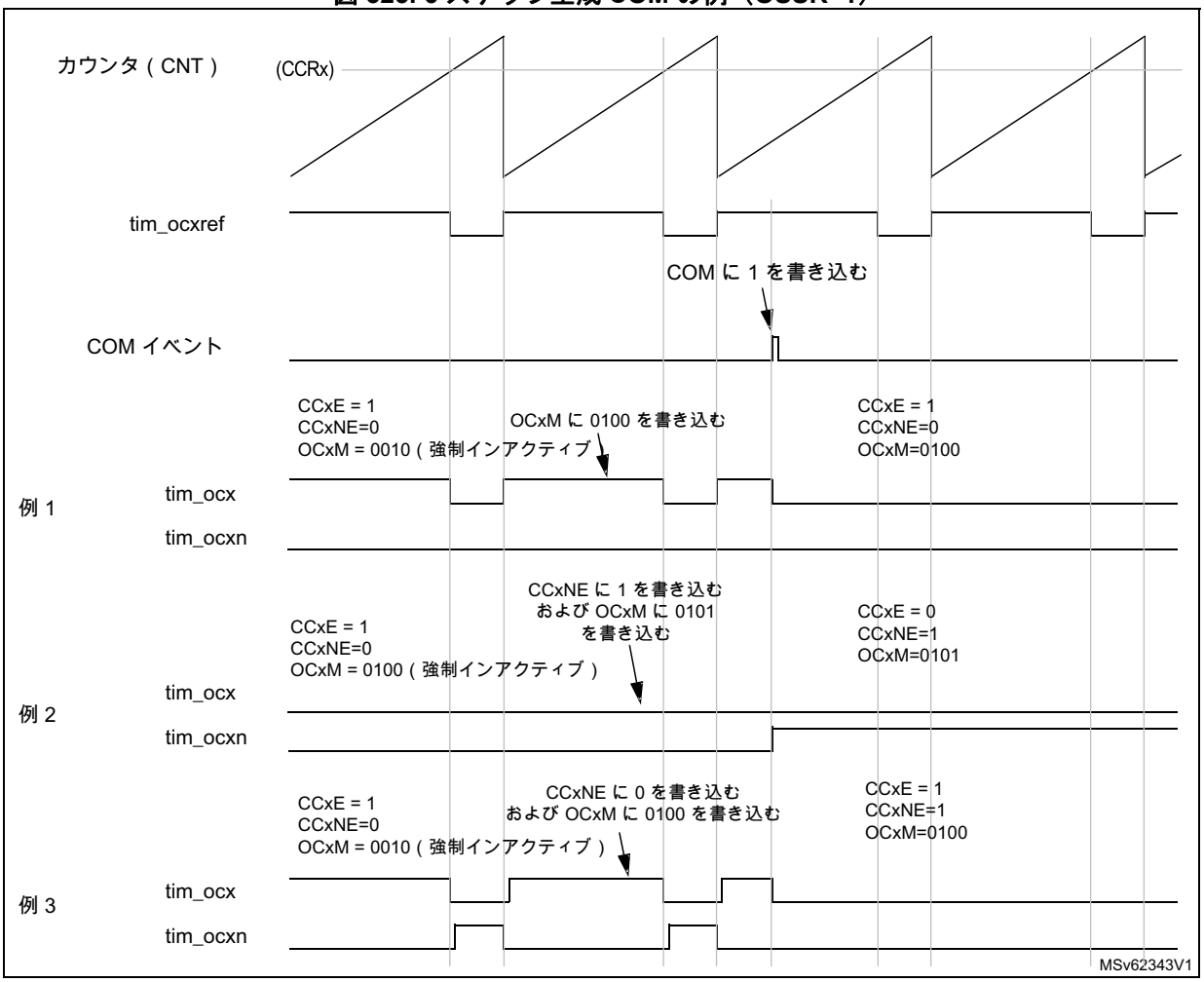
28.3.21 6 ステップ PWM 生成

チャネルで相補出力が使用されているときには、OCxM、CCxE、および CCxNE ビットでプリロードビットが使用できます。プリロードビットは、COM 転流イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COM は、TIMx_EGR レジスタの COM ビットをセットすることによってソフトウェアによって、またはハードウェアによって（tim_trgi 立ち上がりエッジで）生成することができます。

フラグは、COM イベントが発生したときにセットされ（TIM x _SR レジスタの COMIF ビット）、これによって割込み（TIMx_DIER レジスタの COMIE ビットがセットされている場合）または DMA リクエスト（TIMx_DIER レジスタの COMDE ビットがセットされている場合）を生成できます。

図 323 に、COM イベントが発生したときの tim_ocx と tim_ocxn 出力の動作を、3 種類のプログラミング構成の例で示します。

図 323. 6 ステップ生成 COM の例（OSSR=1）



28.3.22 ワンパルスモード

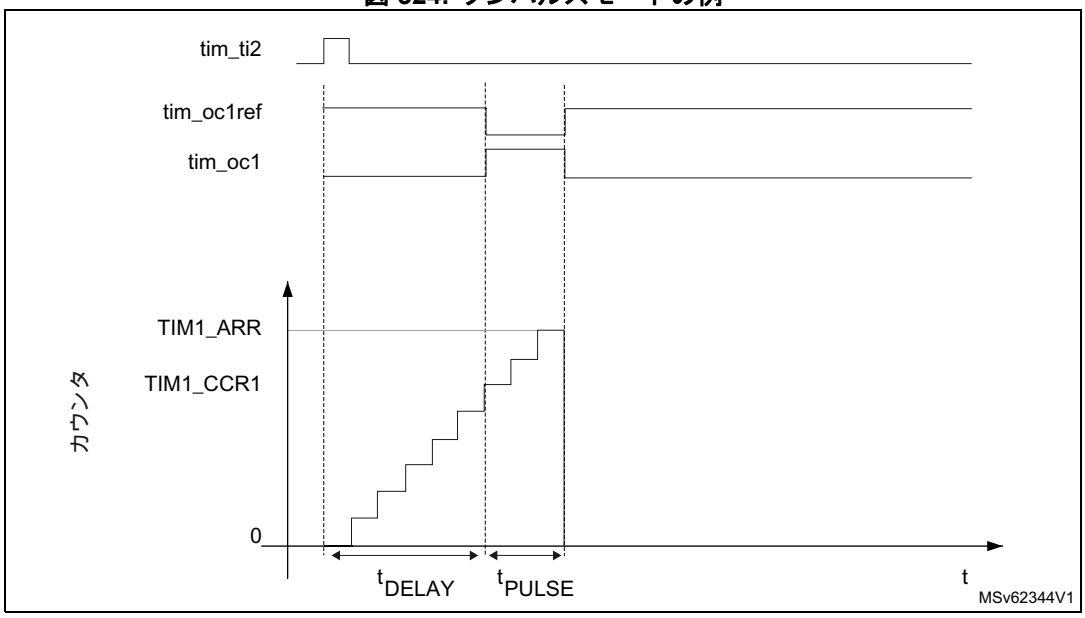
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- アップカウント時：CNT < CCRx≤ ARR（特に、0 < CCRx）
- ダウンカウント時：CNT>CCRx

図 324. ワンパルスモードの例



たとえば、tim_ti2 入力ピンで立ち上がりエッジが検出されたときに、tim_oc1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

tim_ti2fp2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタの CC2S ビットに”01”を書き込むことによって、tim_ti2fp2 を tim_ti2 に配置します。
- tim_ti2fp2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
- tim_ti2fp2 をスレーブモードコントローラのトリガ （tim_trgi）として構成するために、TIMx_SMCR レジスタの TS=“00110”を書き込みます。
- tim_ti2fp2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE='1' を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、tim_ti2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊ケース：tim_ocx 高速イネーブル：

ワンパルスモードでは、tim_tix 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、tim_ocxref（および tim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

28.3.23 再トリガ可能なワンパルス･モード

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 28.3.22 : で説明する再トリガ不可能なワンパルス･モードと、次のような違いがあります。

- トリガが発生するとパルスが直ちに始まります（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

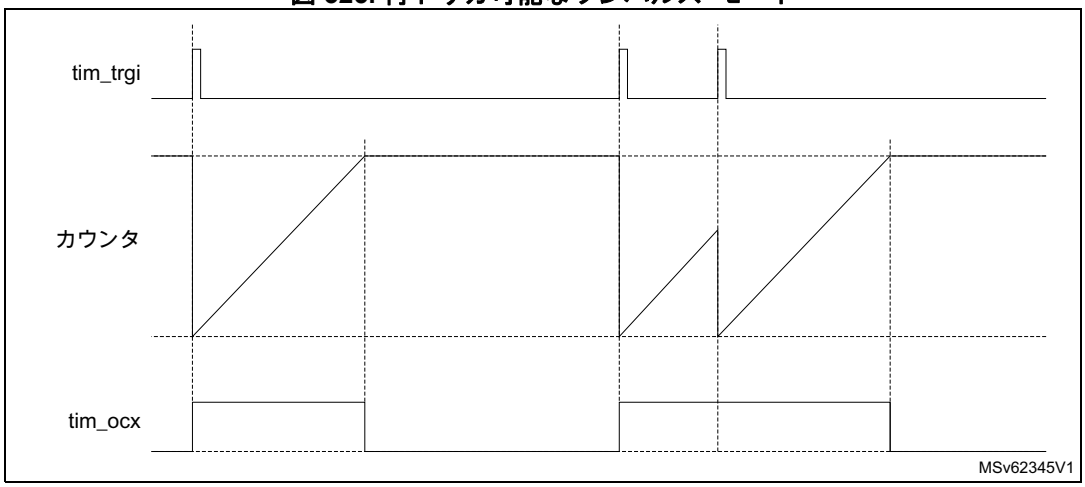
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **OCxM[3:0] および SMS[3:0]ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 325. 再トリガ可能なワンパルス･モード

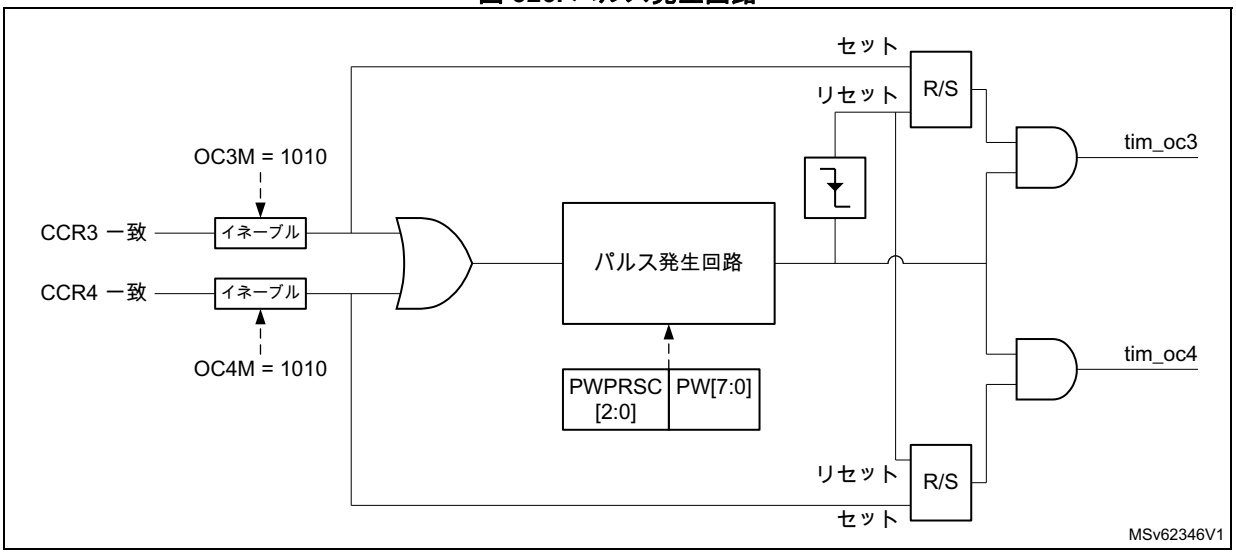


28.3.24 比較モードでのパルス

比較一致イベント時にパルスを生成できます。デバッグまたは同期のために、カウンタ値が特定の比較値と等しいときに、プログラム可能なパルス幅を持つ信号が生成されます。

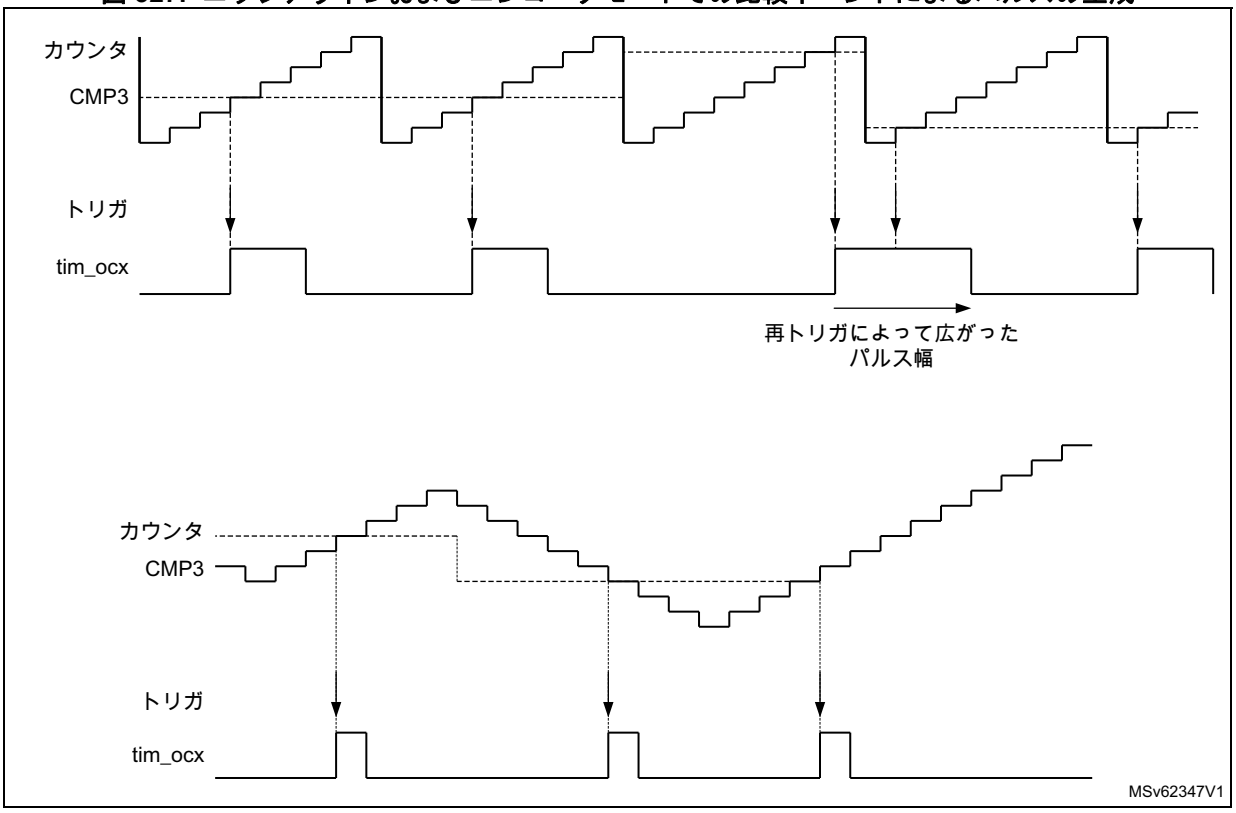
このモードは、エンコーダモードを含むすべてのスレーブモードの選択で、エッジアラインおよびセンタアラインのカウントモードで使用できます。これは、チャネル 3 とチャネル 4 のみで使用できます。以下の図 326 に示すように、パルス発生回路はただ 1 つであり、2 つのチャネルで共有されます。

図 326. パルス発生回路



下記の図 327 は、エッジアラインおよびエンコーダ動作モードでパルスがどのように生成されるかを示しています。

図 327. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成



この出力比較モードは、TIMx_CCMR2 レジスタの OC3M[3:0] および OC4M[3:0] ビットフィールドを使用して選択されます。

パルス幅は、次のように、PWPRSC[2:0] ビットに従ってプリスケールされた特定のクロックを使用し、レジスタの PW[7:0] ビットフィールドを使用してプログラムします。

$t_{PW} = PW[7:0] \times t_{PWG}$

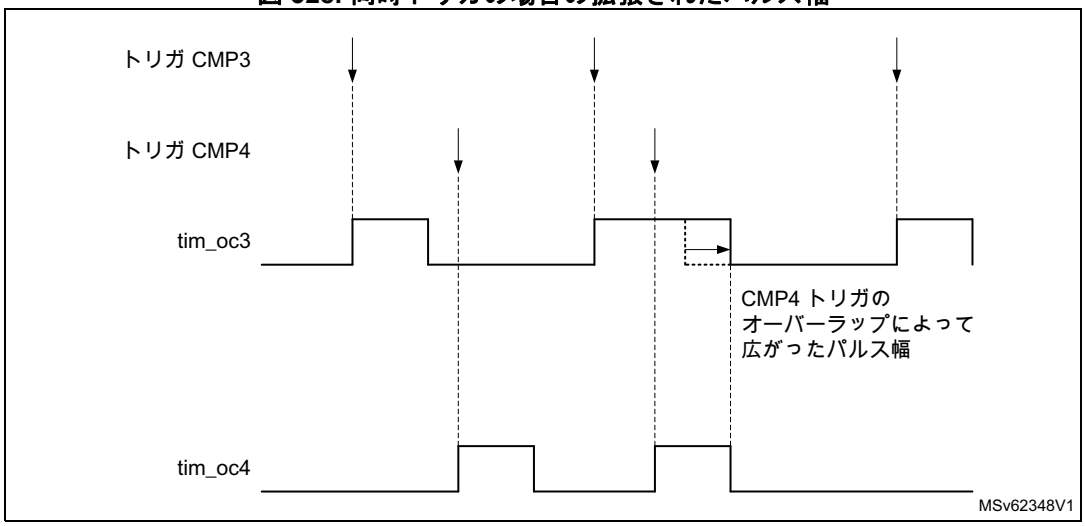
ここで、$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$。

上記は、プリスケーラの値に応じた分解能と最大値を示します。

パルスは再トリガ可能です。パルスが駆動されている間に新しいトリガが入ると、パルスは延長されます。

注： **2 つのチャネルが同時に有効になっている場合、いずれかのチャネルのトリガが、同時出力に生成されたパルスとオーバーラップしない限り、両チャネルのパルスは互いに独立して発行されます。反対に、2 つのトリガがオーバーラップしている場合、最初に到着したトリガに関連するパルス幅は（再トリガのために）拡張されますが、後に到着したトリガのパルス幅は正しくなります（以下の図 328 に示すとおり）。**

図 328. 同時トリガの場合の拡張されたパルス幅



28.3.25 エンコーダインタフェースモード

直交エンコーダ

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが tim_ti1 エッジのみをカウントしている場合は SMS=“0001”を、tim_ti2 エッジのみをカウントしている場合は SMS=“0010”を、tim_ti1 と tim_ti2 の両方のエッジをカウントしている場合は SMS=“0011”を書き込みます。

tim_ti1 と tim_ti2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。必要なときには、入力フィルタもプログラミングできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 tim_ti1 と tim_ti2 は、直交エンコーダとのインタフェースに使用されます。表 261 を参照してください。カウンタのクロック供給は、tim_ti1fp1 または tim_ti2fp2（入力フィルタおよび極性選択の後は tim_ti1 と tim_ti2、フィルタされず反転されない場合は tim_ti1fp1=tim_ti1、フィルタされず反転されない場合は tim_ti2fp2=tim_ti2）での有効な遷移ごとに行われます。ただし、カウンタは有効である（TIMx_CR1 レジスタの CEN ビットに“1”が書き込まれている）ことが前提です。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが tim_ti1 のみ、tim_ti2 のみ、または tim_ti1 と tim_ti2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（tim_ti1 または tim_ti2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、繰り返しカウンタ、トリガ出力の機能は通常どおりに機能を続けます。エンコーダモードと外部クロックモード 2 は互換性がないので、同時に選択することはできません。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（tim_ti1 と tim_ti2 は同時に切り替わらないと想定しています）。

表 261. カウント方向とエンコーダ信号（CC1P = CC2P = 0）

アクティブエッジ	SMS[3:0]	他方の信号のレベル（tim_ti2 の場合は tim_ti1fp1、tim_ti1 の場合は tim_ti2fp2）	tim_ti1fp1 信号		tim_ti2fp2 信号	
			立ち上がり	立ち下がり	立ち上がり	立ち下がり
tim_ti1 のみカウント x1 モード	1110	ハイ	ダウン	アップ	カウントなし	カウントなし
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti2 のみカウント x1 モード	1111	ハイ	カウントなし	カウントなし	アップ	ダウン
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti1 のみカウント x2 モード	0001	ハイ	ダウン	アップ	カウントなし	カウントなし
		ロー	アップ	ダウン	カウントなし	カウントなし
tim_ti2 のみカウント x2 モード	0010	ハイ	カウントなし	カウントなし	アップ	ダウン
		ロー	カウントなし	カウントなし	ダウン	アップ
tim_ti1 と tim_ti2 の両方をカウント x4 モード	0011	ハイ	ダウン	アップ	アップ	ダウン
		ロー	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部トリガ入力に接続して、カウンタのリセットをトリガできます。

図 329 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=“01”（TIMx_CCMR1 レジスタ、tim_ti1fp1 は tim_ti1 に配置）
- CC2S=“01”（TIMx_CCMR2 レジスタ、tim_ti1fp2 は tim_ti2 に配置）
- CC1P=“0”、CC1NP =“0”（TIMx_CCER レジスタ、tim_ti1fp1 非反転、tim_ti1fp1=tim_ti1）
- CC2P=“0”、CC2NP =“0”（TIMx_CCER レジスタ、tim_ti1fp2 非反転、tim_ti1fp2=tim_ti2）
- SMS=“011”（TIMx_SMCR　レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=“1”（TIMx_CR1 レジスタ、カウンタ有効）

図 329. エンコーダインタフェースモードにおけるカウンタの動作例

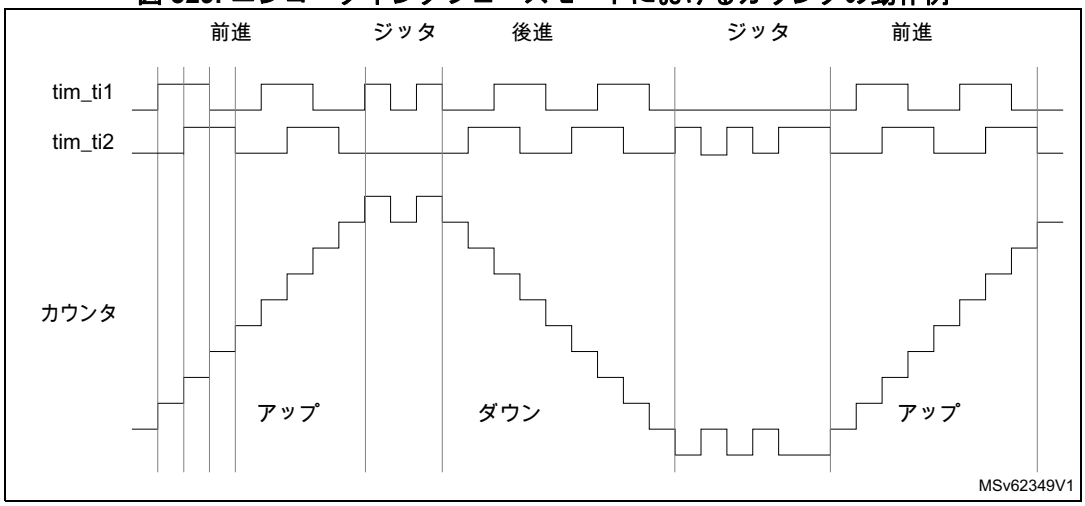
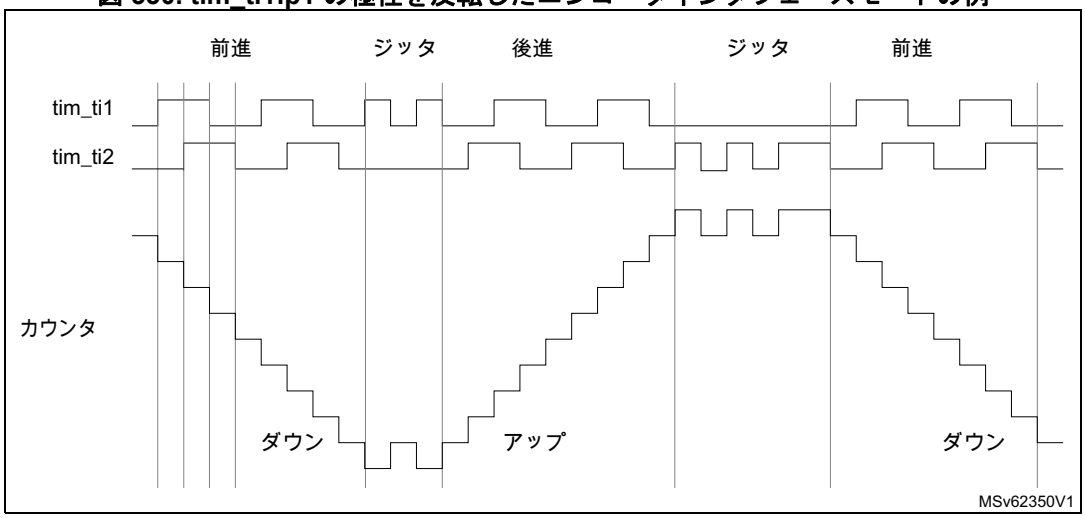


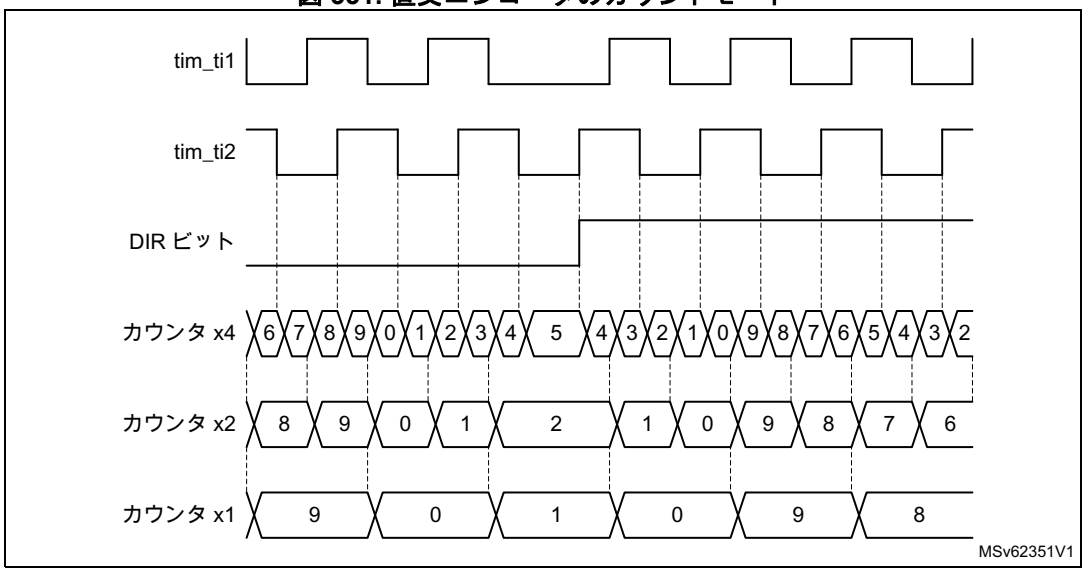
図 330 に、tim_ti1fp1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P="1"）。

図 330. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例



以下の図 331 は、さまざまなカウントモードでの速度反転中のタイマカウンタ値を示しています。

図 331. 直交エンコーダのカウントモード



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、DMA リクエストを通じて値を読み出すことも可能です。

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

クロックプラス方向エンコーダモード

直交エンコーダモードのほか、タイマは他のタイプのエンコーダもサポートしています。

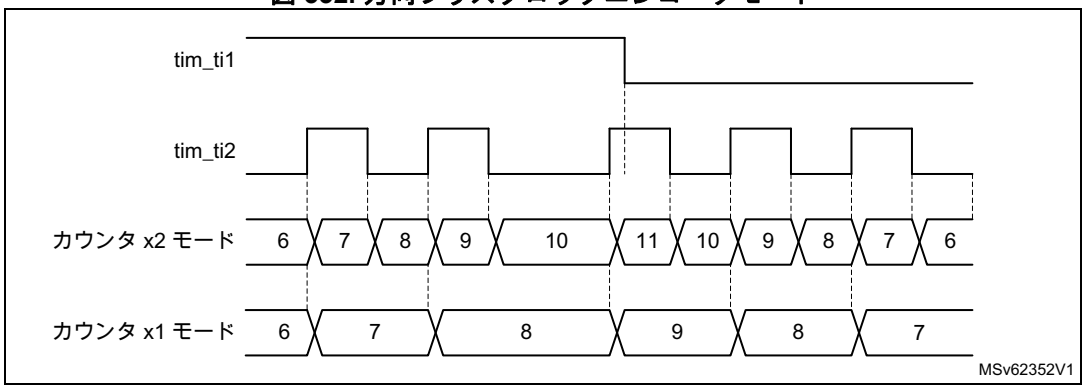
図 332 に示す「クロックプラス方向」モードでは、tim_ti2 の単一ラインでクロックが提供され、tim_ti1 入力を使用して方向が強制されます。

このモードは TIMx_SMCR レジスタの SMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1010 : X2 モード。カウンタはクロックの立ち上がりと立ち下がりの両エッジで更新されます。
- 1011 : X1 モード。カウンタは 次の CC2P ビット値に従って単一のクロックエッジで更新されます。CC2P = 0 は立ち上がりエッジ検出に対応し、CC2P = 1 は立ち下がりエッジ検出に対応します。

tim_ti1 の方向信号の極性は、次の CC1P ビットで設定されます。0 は正の極性（tim_ti1 がハイのときアップカウントで、tim_ti1 がローのときダウンカウント）に対応し、CC1P = 1 は負の極性（tim_ti1 がローのときアップカウント）に対応します。

図 332. 方向プラスクロックエンコーダモード



方向クロックエンコーダモード

図 333 の「方向クロック」モードでは、クロックは 2 本のラインで提供され、1 本のアップカウントクロックラインと 1 本のダウンカウントクロックラインを持つというように、方向に応じて一度に 1 つが提供されます。

このモードは TIMx_SMCR レジスタの SMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1100 : x2 モード。カウンタは、2 つのクロックラインのいずれかの立ち上がりと立ち下がりの両エッジで更新されます。CC1P および CC2P ビットは、クロックアイドル状態用のコーディングです。CCxP = 0 はハイレベルアイドル状態に対応し（下記図 333 を参照）、CCxP = 1 はローレベルアイドル状態に対応します（下記図 334 を参照）。
- 1101 : x1 モード。カウンタは、CC1P および CC2P ビット値に従って、単一のクロックエッジで更新されます。CCxP = 0 は立ち下がり検出とハイレベルアイドル状態に対応し（下記図 333 を参照）、CCxP = 1 は立ち上がり検出とローレベルアイドル状態に対応します（下記図 334 を参照）。

図 333. 方向クロックエンコーダモード（CC1P = CC2P = 0）

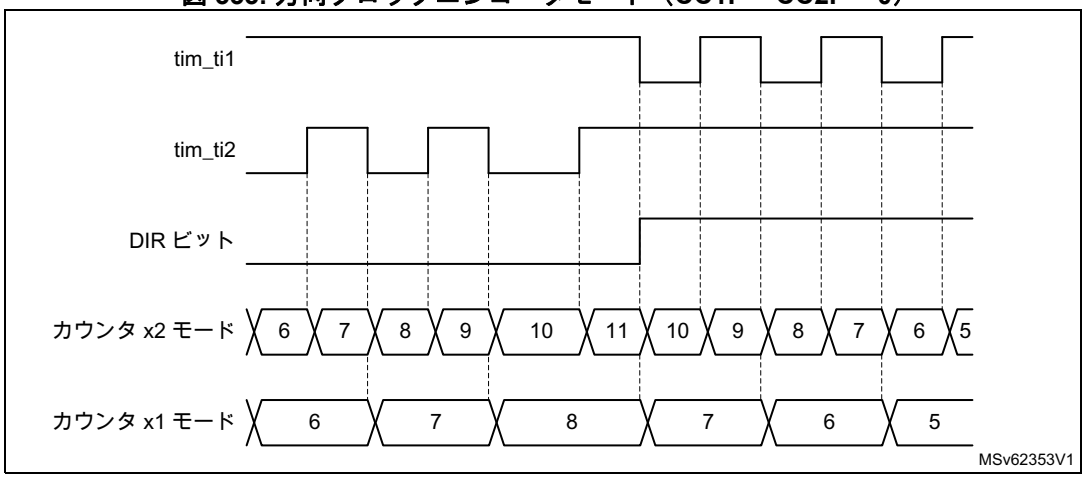
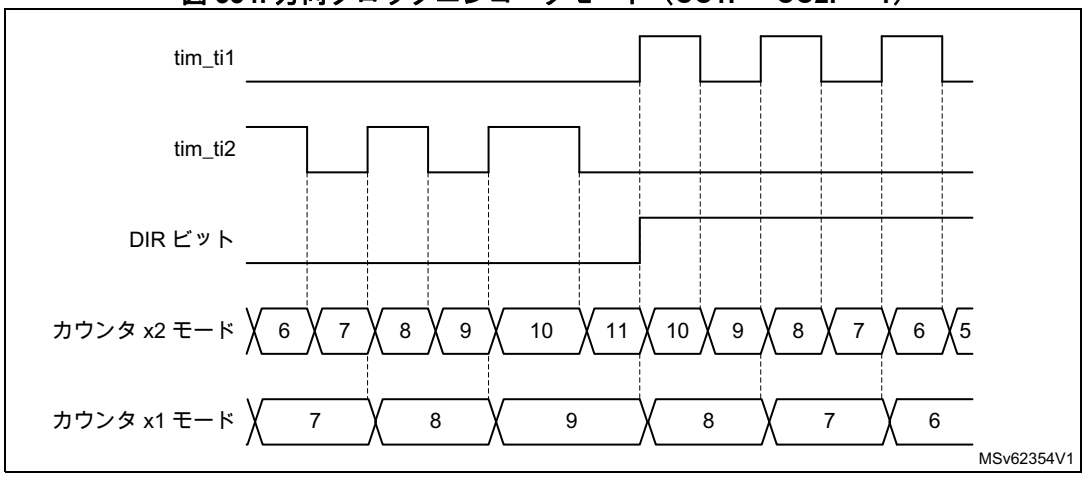


図 334. 方向クロックエンコーダモード（CC1P = CC2P = 1）



以下の表 262 に、任意の入力遷移に対して方向クロックモードがどのように動作するかを詳しく説明します。

表 262. カウント方向 対 エンコーダ信号および極性設定

方向クロックモード	SMS[3:0]	他方の信号のレベル（tim_ti2 の場合は tim_ti1fp1、tim_ti1 の場合は tim_ti2fp2）	tim_ti1fp1 信号		tim_ti2fp2 信号	
			立ち上がり	立ち下がり	立ち上がり	立ち下がり
x2 モード CCxP=0	1100	ハイ	ダウン	ダウン	アップ	アップ
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
x2 モード CCxP=1	1100	ハイ	カウントなし	カウントなし	カウントなし	カウントなし
		ロー	ダウン	ダウン	アップ	アップ
x1 モード CCxP=0	1101	ハイ	カウントなし	ダウン	カウントなし	アップ
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
x1 モード CCxP=1	1101	ハイ	カウントなし	カウントなし	カウントなし	カウントなし
		ロー	ダウン	カウントなし	アップ	カウントなし

インデックス入力

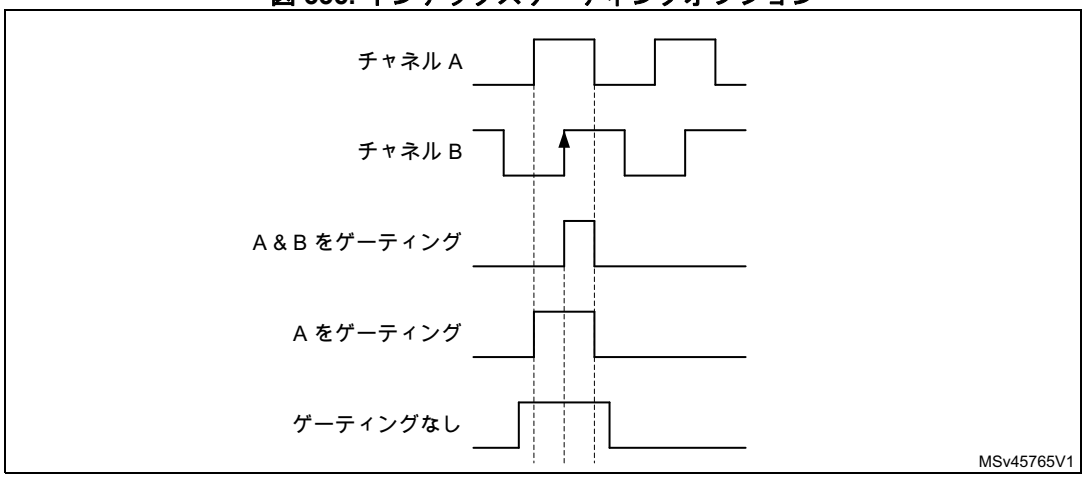
カウンタは、絶対基準位置を示すエンコーダからのインデックス信号によってリセットできます。インデックス信号は、tim_etr_in 入力に接続する必要があります。これは、デジタル入力フィルタを使用してフィルタリングできます。

インデックス機能は、TIMX_ECR レジスタの IE ビットで有効にされます。IE ビットは、SMS[3:0] ビットフィールドが次の値であるときに、エンコーダモードでのみセットする必要があります。0001、0010、011、1010、1011、1100、1101、1110、1111。

以下の図 335 に示すように、インデックスパルス調整のいくつかのオプションを備えた市販のエンコーダが提案されています。

- A および B でゲート処理：パルス幅は 1 つのチャネル周期の 1/4 であり、A および B の両方のエッジにそろえられます。
- A でゲート処理（または B でゲート処理）：パルス幅は 1 つのチャネル周期の 1/2 であり、チャネル A（または チャネル B）の 2 つのエッジにそろえられます。
- ゲート処理なし：パルス幅は最大 1 チャネル周期であり、どのエッジにもそろえられません。

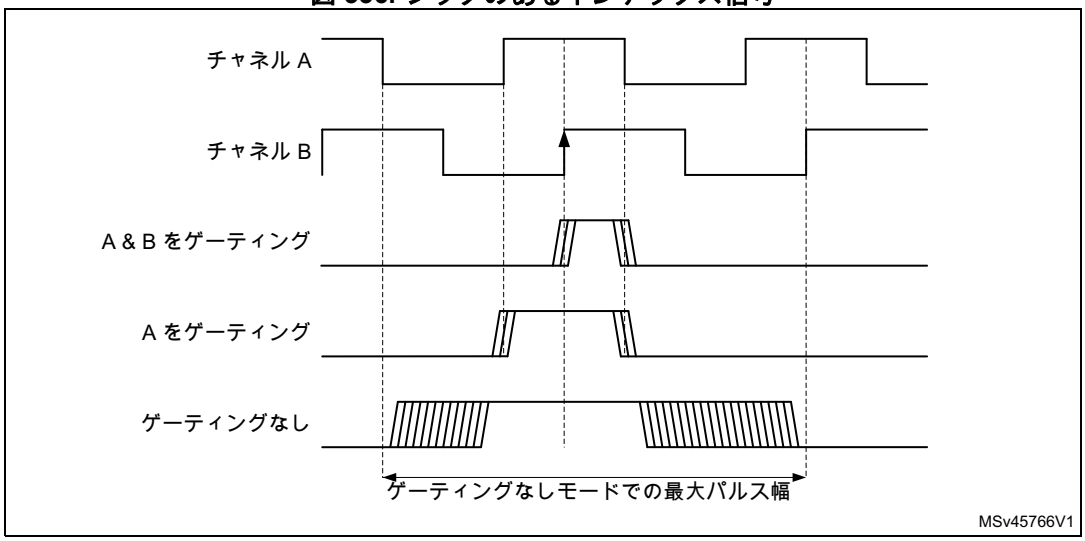
図 335. インデックスゲーティングオプション



以下の図 336 に示すように、ゲーティングモードに関係なく、回路はインデックス信号のジッタを許容します。

ゲート処理なしモードでは、信号はエンコーダの 2 周期より厳密に小さくなければなりません。パルス幅がエンコーダの 2 周期以上の場合、カウンタは複数回リセットされます。

図 336. ジッタのあるインデックス信号



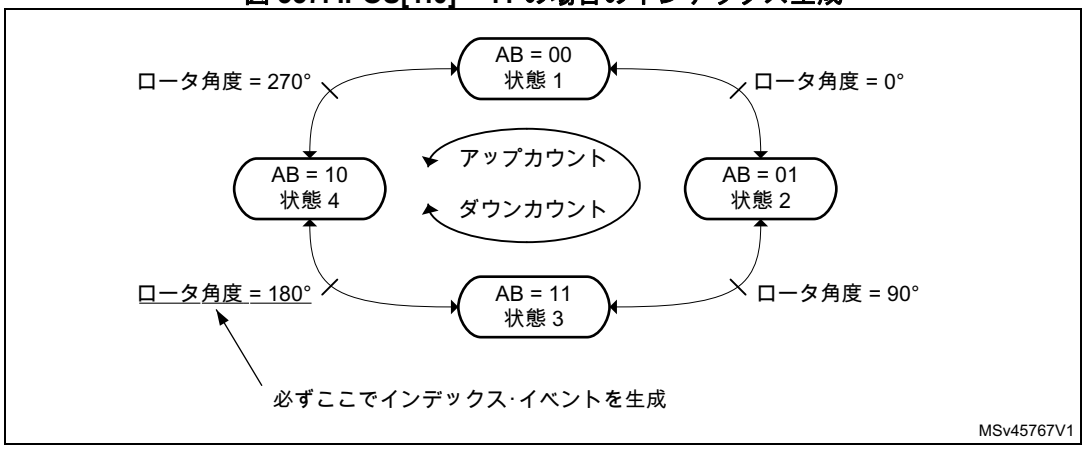
タイマは、特定のプログラミングを必要とせずに、3 つのゲーティングオプションを同様にサポートします。TIMx_ECR レジスタの IPOS[1:0] ビットフィールドを使用して、インデックスを同期する必要があるエンコーダ状態（チャネル A とチャネル B の状態の組み合わせ）を定義するだけです。

インデックス検出イベントは、速度の反転中に対称動作を確保するために、カウント方向に応じて異なる動作をします。

- カウンタはアップカウント中（DIR ビット = 0）にリセットされます。
- カウンタはダウンカウント時に、TIMx_ARR に設定されます。

これにより、カウント方向に関係なく、まったく同じ機械的角度位置でインデックスを生成できます。以下の図 337 は、単純な例（機械的回転ごとに 4 つのエッジを提供するエンコーダ）について、生成されるインデックスの位置を示しています。

図 337. IPOS[1:0] = 11 の場合のインデックス生成



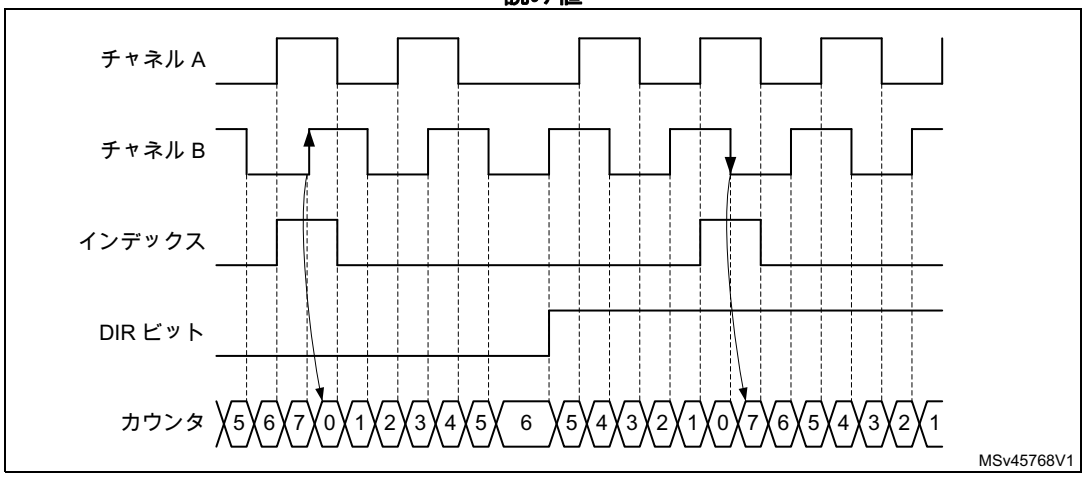
以下の図 338 に、IPOS[1:0] = 11 の場合の波形と対応する値を示します。カウンタの値が強制される瞬間は、カウント方向に応じて自動的に調整されることを示しています。

- アップカウント（DIR ビット= 0）の場合、エンコーダの状態が「11」（ChA = 1、ChB = 1）のとき、カウンタは 0 にセットされます。
- ダウンカウント（DIR ビット= 1）の場合、「11」状態を終了するとき、カウンタは TIMx_ARR にセットされます。

インデックス検出イベント時に割込みを発行できます。

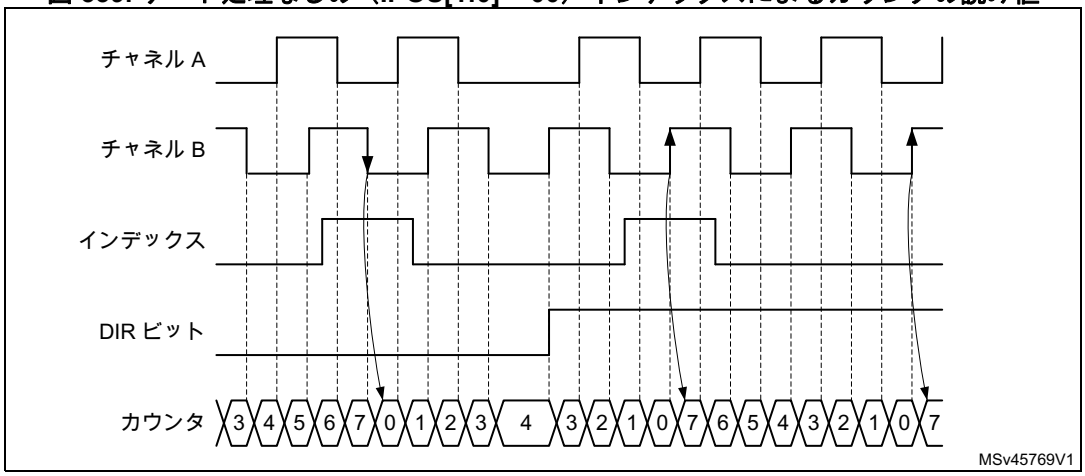
矢印は、どの遷移でインデックスイベント割込みが生成されるかを示しています。

図 338. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値



以下の図 339 に、ゲート処理なしモードの場合の波形と対応する値を示します。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図 339. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値



以下の図 340 は、さまざまなパルスアライメントシナリオでの「A & B でゲート処理」モードの処理方法を示しています。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図 340. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値

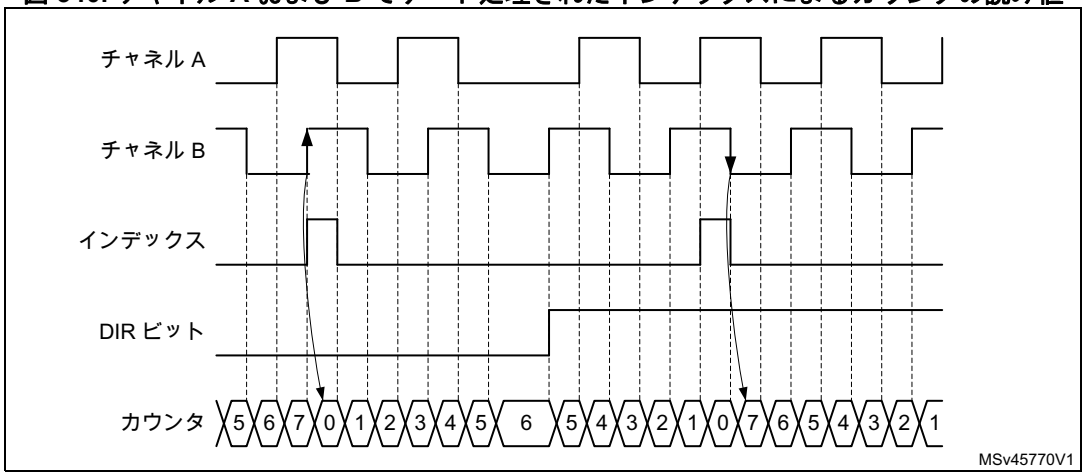


図 341 と 図 342 には、後続のインデックスパルスが、エンコーダクロック周期の 4 分の 1 よりも狭いことがあるケースが詳しく示されています。

図 341. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11）

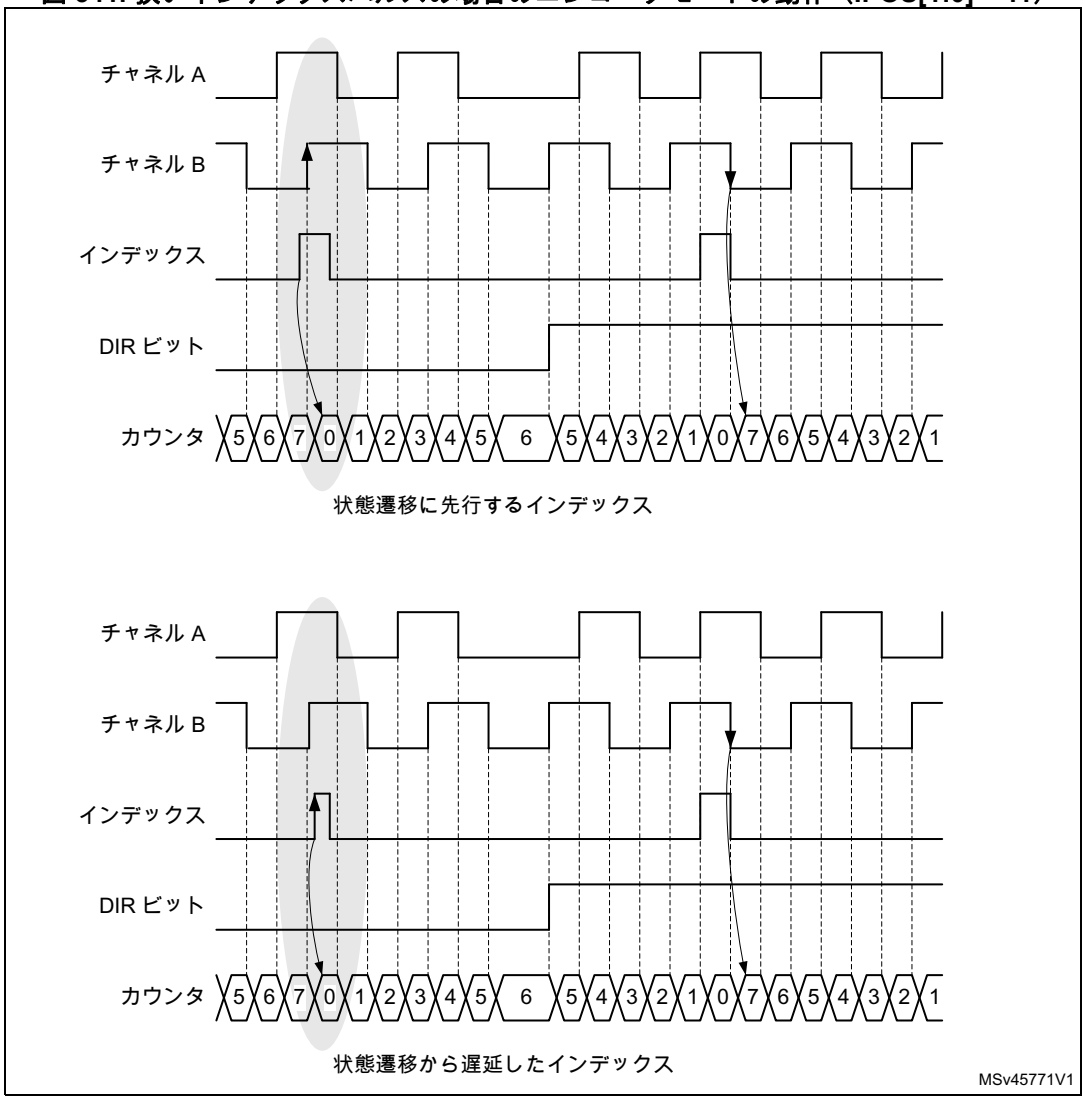
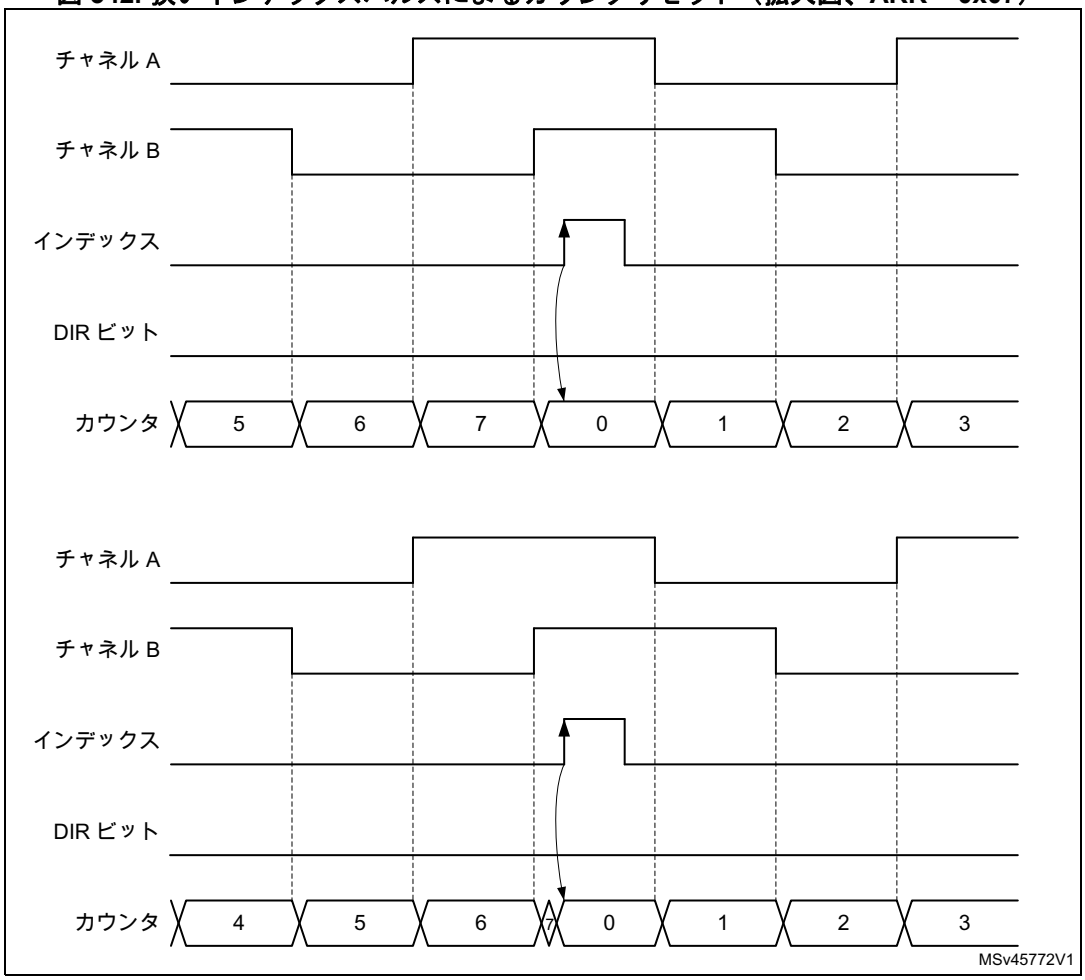
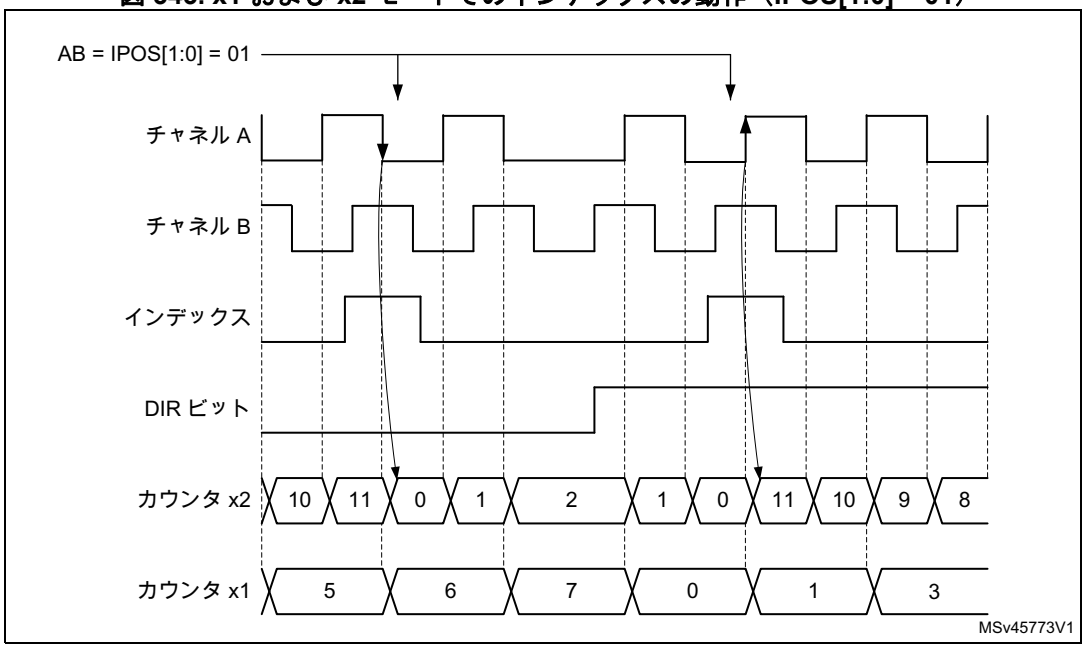


図 342. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07）



下記の図 343 は、x1 および x2 モードでインデックスがどのように管理されるかを示します。

図 343. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01）



インデックスの方向感受性

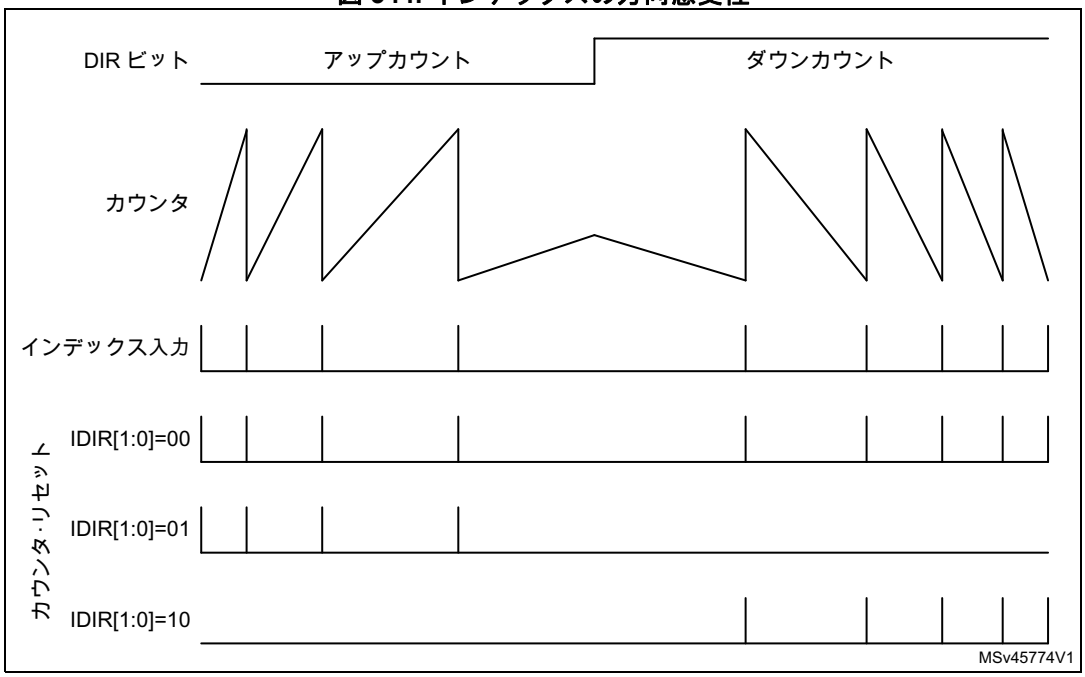
TIMx_ECR レジスタの IDIR[1:0] ビットフィールドにより、選択したカウント方向でのみインデックスをアクティブにできます。

以下の図 344 は、IDIR[1:0] 値に応じた、インデックスとカウンタのリセットイベントの関係を示しています。

注： **IDR[1:0] ビットフィールドは、IE ビットがリセットされているとき（インデクスモード無効時）に書き込む必要があります。**

注： **インデックスの方向感受性は、クロックプラス方向モードではサポートされていません。SMS[3:0] = 1010 または 1011 の場合、IDIR[1:0] は 00 にセットしなければなりません。**

図 344. インデックスの方向感受性

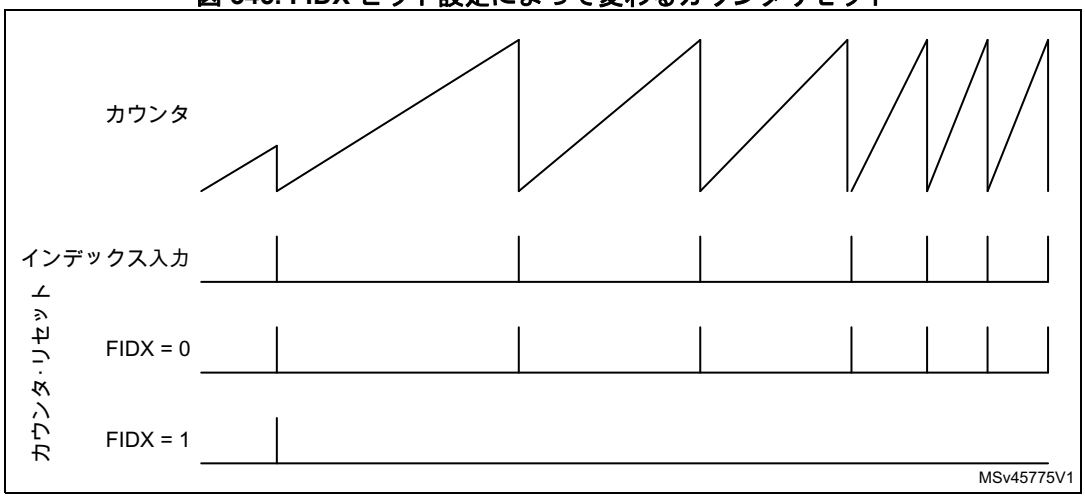


特殊な最初のインデックスイベント管理

TIMx_ECR レジスタの FIDX ビットによって、以下の図 345 に示すように、インデックスを 1 回だけ取得できます。最初のインデックスが到着すると、その後のインデックスは無視されます。必要に応じて、FIDX ビット に 0 を書き込み、再度 1 に設定することにより、回路を再びインデックスを取得できる状態にすることができます。

注： **FIDX = 1 のとき、位置 0（インデックスがアクティブ）で方向が変わると、インデックスを 2 回発行できます（IDXF フラグが設定されます）。**

図 345. FIDX ビット設定によって変わるカウンタリセット



非直交モードでのインデックス管理

以下の図 346 および図 347 では、SMS[3:0] ビットフィールドが 1010、1011、1100、1101 に等しい場合に、方向クロックモードおよびクロックプラス方向モードでインデックスがどのように管理されるかを詳しく説明します。

これらのモードの両方で、インデックスの検出は次のように IPOS[0] ビットで設定されます。

- IPOS[0] = 0：インデックスはクロックのローレベルで検出されます。
- IPOS[0] = 1：インデックスはクロックのハイレベルで検出されます。

IPOS[1] ビットは意味がありません。

図 346. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1）

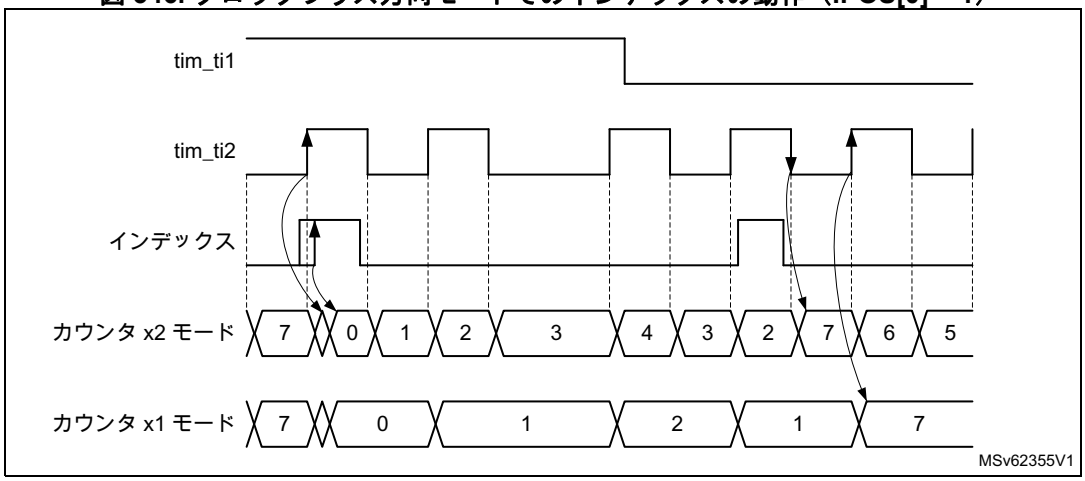
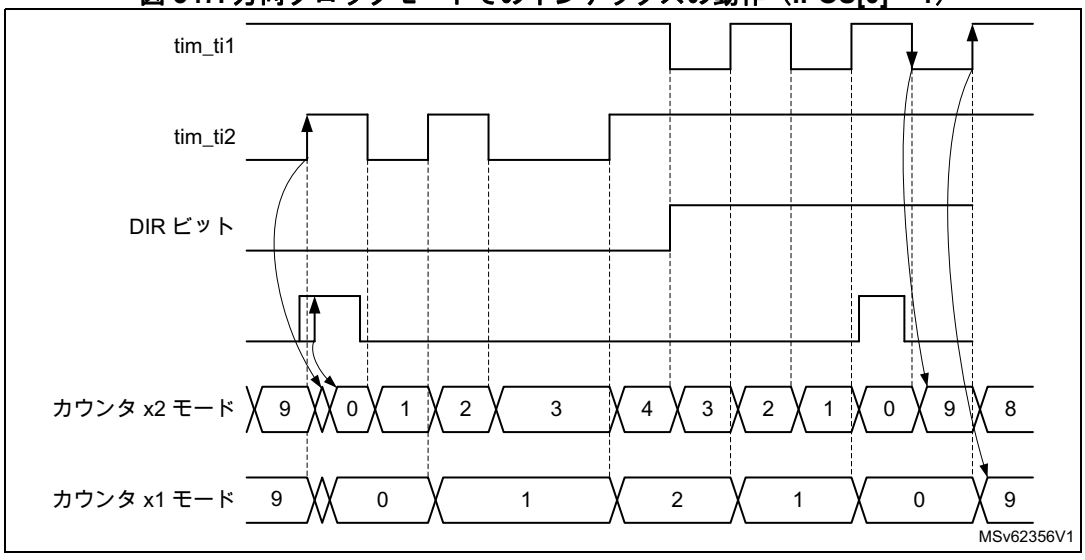


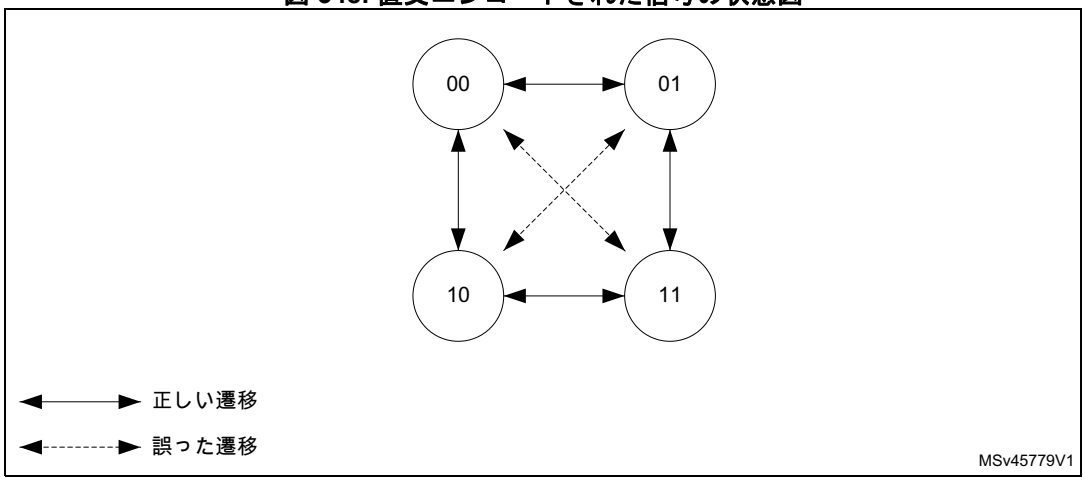
図 347. 方向クロックモードでのインデックスの動作（IPOS[0] = 1）



エンコーダのエラー管理

2 つの直交信号が使用できるエンコーダ構成の場合、遷移エラーを検出できます。2 つの入力の読み値は、以下の図 348 で、状態図として表すことができる 2 ビットのグレーコードに対応しています。一度に 1 つのビットだけが変化することが予測されています。間違った遷移があると、TIMx_SR ステータス·レジスタの TERRF 割込みフラグがセットされます。TIMx_DIER レジスタの TERRIE ビットがセットされている場合、遷移エラー割込みが生成されます。

図 348. 直交エンコードされた信号の状態図



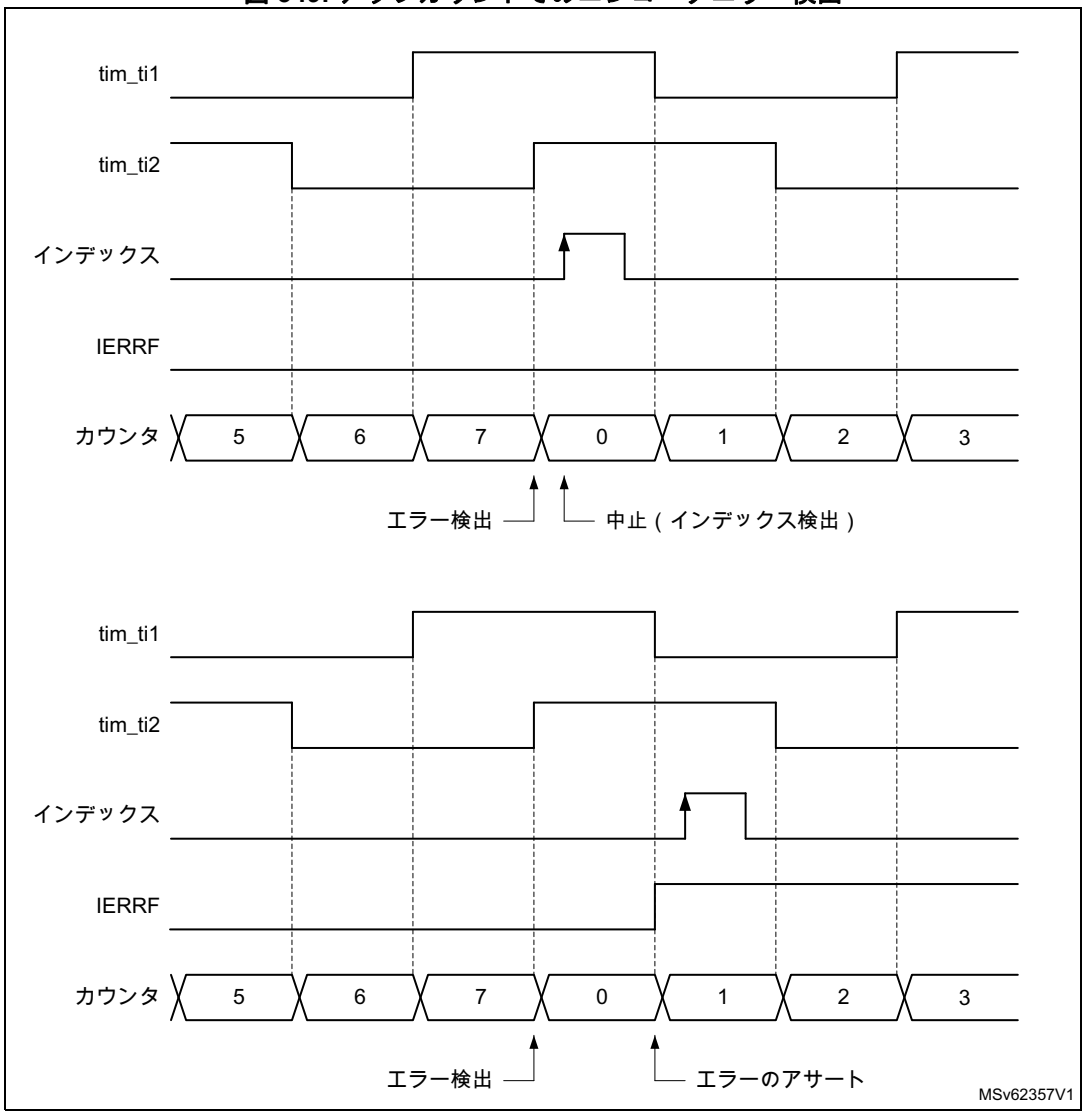
インデックス信号を持つエンコーダの場合、1 回転あたりのパルスが過剰になる異常動作を検出できます。1 回転あたり N 個のパルスを生成するエンコーダでは、1 回転あたり 4xN カウントが得られます。インデックス信号によって、4xN クロック周期ごとにカウンタがリセットされます。

カウンタ値がインデックス·イベントなしに TIMx_ARR から 0 にインクリメントされるか、0 から TIMxARR 値にデクリメントされると、インデックス位置エラーとして報告されます。

オーバーフロー閾値は、TIMx_ARR レジスタを使用してプログラムします。1,000 ラインのエンコーダでは、カウンタ値が 0 ~ 3999 になります（4x 読み取りモードの場合）。オーバーフロー検出閾値は、TIMx_ARR = 3999 + 1 = 4000 を設定してプログラムする必要があります。

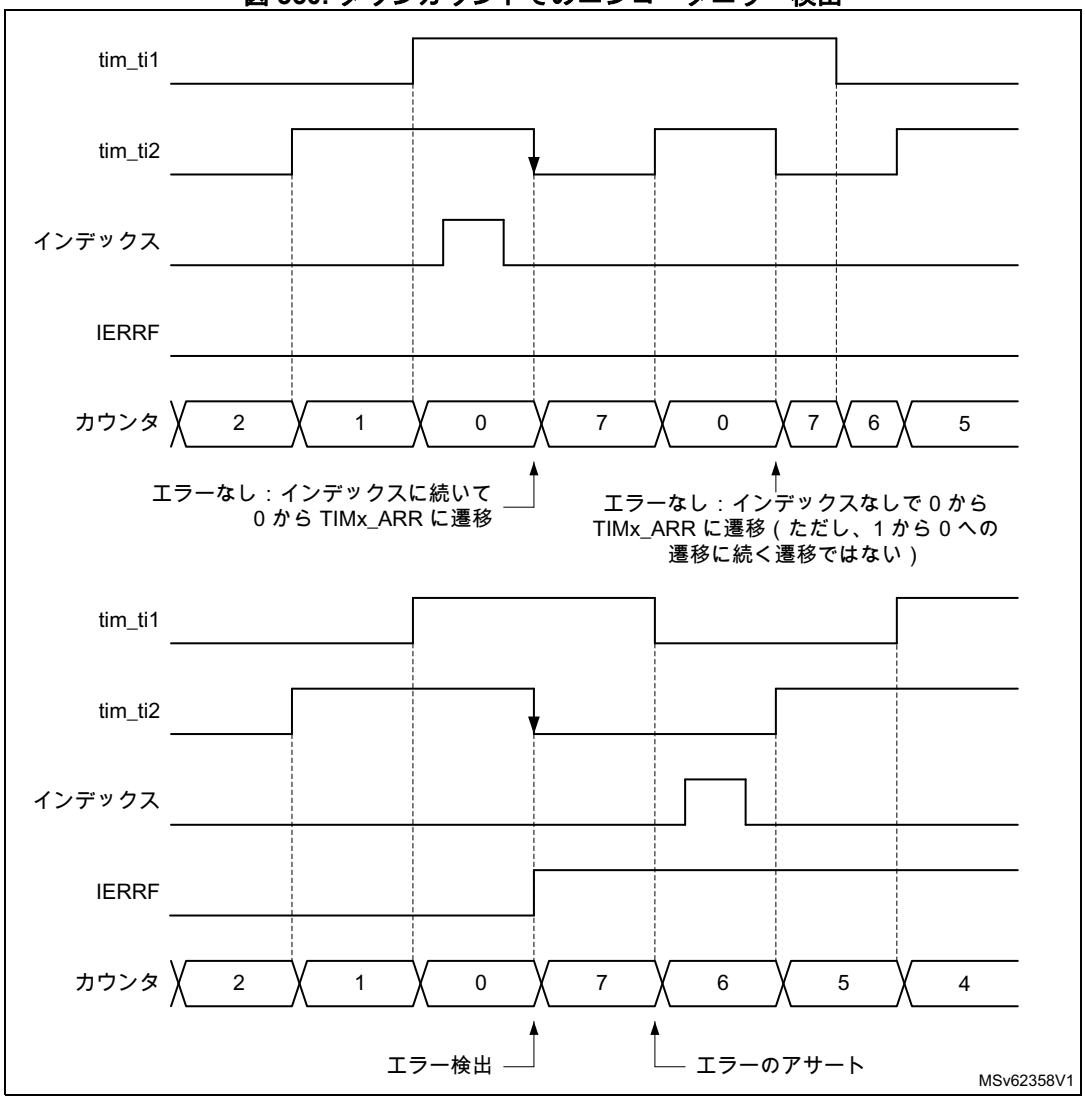
アップカウントでは、エラーアサーションは 0 から 1 への遷移まで遅延されます。これは、以下の図 349 に示すように、A および B ゲート処理モードでの狭いインデックスパルスへの対処です。

図 349. アップカウントでのエンコーダエラー検出



ダウンカウントモードでは、検出は 1 から 0 への事前遷移によって条件付けられます。これは、以下の図 350 に示すように、A および B ゲート処理モードでの狭いインデックスパルスに対処し、インデックス検出直後にエンコーダが TIMx_ARR と 0 の間でディザリングする場合の誤ったエラー検出を回避するためです。

図 350. ダウンカウントでのエンコーダエラー検出



インデックスエラーによって、TIMx_SR ステータスレジスタの IERRF 割込みフラグがセットされます。TIMx_DIER レジスタの IERRIE ビットがセットされている場合、インデックス･エラー割込みが生成されます。

エンコーダの機能的割込み

エンコーダモードでは、以下の割込みも使用できます。

- 方向変更：エンコーダ･モードでのカウント方向の変更によって、TIMx_CR1 レジスタの DIR ビットが反転します。方向変更によって、TIMx_SR ステータス･レジスタの DIRF 割込みフラグがセットされます。TIMx_DIER レジスタの DIRIE ビットがセットされている場合、方向変更割込みが生成されます。
- インデックス･イベント：インデックス･イベントによって、TIMx_SR ステータス･レジスタの IDXF 割込みフラグがセットされます。TIMx_DIER レジスタの IDXIE ビットがセットされている場合、インデックス割込みが生成されます。

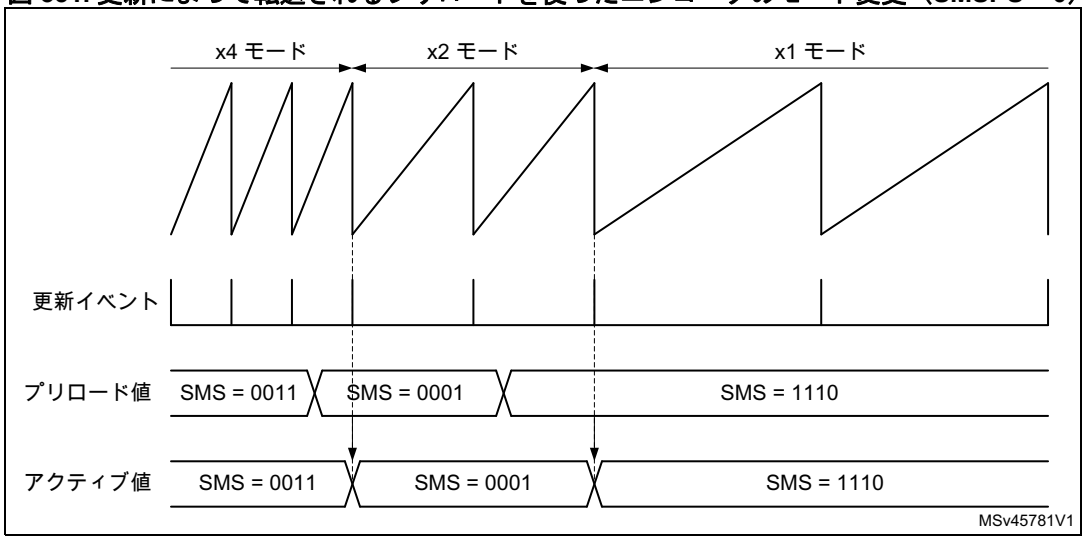
実行時エンコーダ･モード更新のためのスレーブ･モード選択プリロード

実行中に、あるエンコーダモードを別のモードに切り替える必要があることがあります。以下の図 351 に示すように、これは通常、高速時に更新割込みレートを低下させるために、x4 モードから x2 に、x1 に切り替えることにより、実行されます。

この目的のために、SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCR レジスタの SMSPE イネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、TIMx_SMCR レジスタの SMSPS ビットで選択できます。

- SMSPS = 0：転送は、アップカウント時にカウンタがオーバーフローするとき、およびダウンカウント時にアンダーフローするときに発生する更新イベント（UEV）によってトリガされます。このモードは、インデックスが無効な（ビット IE = 0 の）ときにのみ使用する必要があります。
- SMSPS = 1：転送は、インデックスイベントによってトリガされます。

図 351. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）



エンコーダクロック出力

エンコーダ･モードの動作原理は、十分な数のクロック･エッジ取得と正確な測定を行うために比較的長い積分時間を必要とするため、低速における高分解能の速度測定に最適とはいえません。

低速でのより良い対策方法は、エッジ間のクロック周期測定です。これは、スレーブタイマを使用して達成できます。タイマによって、tim_trgo 出力にエンコーダクロック情報を出力できます。その後、スレーブタイマはエンコーダクロックエッジごとに周期測定を実行し、速度情報を提供できます。

このモードは TIMx_CR2 レジスタの MMS[3:0] ビットフィールドを 1000 に設定することによって有効になります。これは、以下の SMS[3:0] 値に対して有効です。0001、0010、0011、1010、1011、1100、1101、1110、1111。その他の SMS[3:0] コードは許されず、予想外の動作につながることがあります。

28.3.26 方向ビットの出力

tim_oc3n および tim_oc4 出力信号に、タイマから方向信号を出力できます（TIMx_CR1 レジスタの DIR ビットのコピー）。これを達成するには、TIMx_CCMR2 レジスタの OC3M[3:0] または OC4M[3:0] ビットフィールドを 1011 にセットします。

この機能を使用すると、エンコーダモードでカウント方向（または回転方向）を監視すること、またはセンタアライン PWM モードでアップ／ダウンフェーズを示す信号を得ることができます。

28.3.27 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特殊なケースでは、バックグラウンドタスク（カウンタの読出し）と中断（更新の中断）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

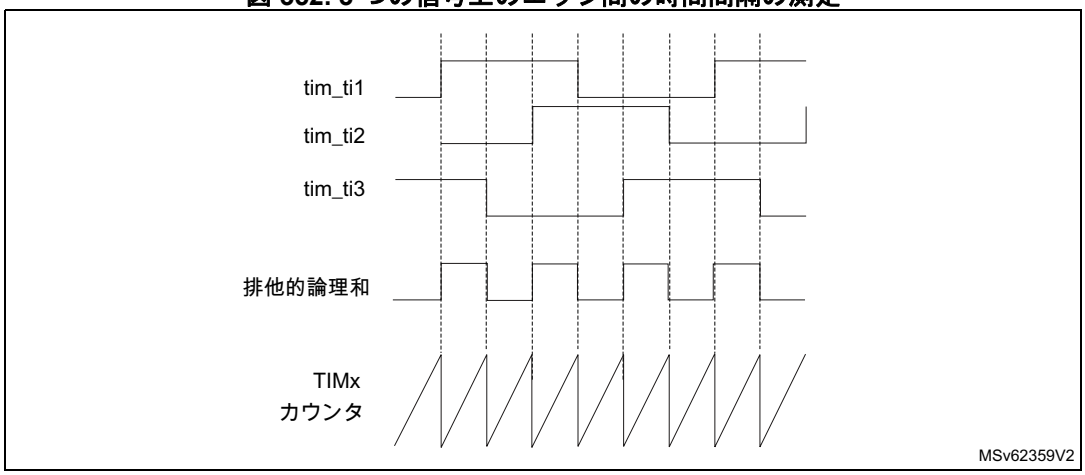
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

28.3.28 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピン tim_ti1、tim_ti2、および tim_ti3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図 352 に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 352. 3 つの信号上のエッジ間の時間間隔の測定



28.3.29 ホールセンサとのインタフェース

これは、高機能制御タイマを使用して PWM 信号を生成し、モータと図 353 で「インタフェースタイマ」と記されている別のタイマ TIMx を駆動することによって実現します。「インタフェースタイマ」は、XOR を通じて tim_ti1 入力チャネル（TIMx_CR2 レジスタの TI1S ビットをセットすることで選択できます）に接続された 3 つのタイマ入力ピン（tim_ti1、tim_ti2、および tim_ti3）をキャプチャします。

スレーブモードコントローラはリセットモードに設定され、スレーブ入力は tim_ti1f_ed です。したがって、3 つの入力のいずれかが反転するごとに、カウンタは 0 からカウントをリスタートします。これが、ホール入力の変化によってトリガされるタイムベースとなります。

「インタフェースタイマ」上で、キャプチャ／比較チャネル 1 がキャプチャモードで設定され、キャプチャ信号は tim_trc です（図 296：1082 ページのキャプチャ／比較チャネル（例：チャネル 1 入力ステージ）を参照）。キャプチャされた値は、入力の 2 回の変化の間の経過時間に対応し、モータの速度情報を与えます。

「インタフェースタイマ」を出力モードで使用して、（COM イベントをトリガすることによって）高機能制御タイマのチャネルの設定を変更するパルスを生成できます。高機能制御タイマは、モータを駆動する PWM 信号を生成するために使用されます。このためには、プログラミングした遅延の後に正パルスが生成されるように（出力比較モードまたは PWM モードで）インタフェースタイマチャネルをプログラミングする必要があります。このパルスは、tim_trgo 出力を通じて高機能制御タイマに送信されます。

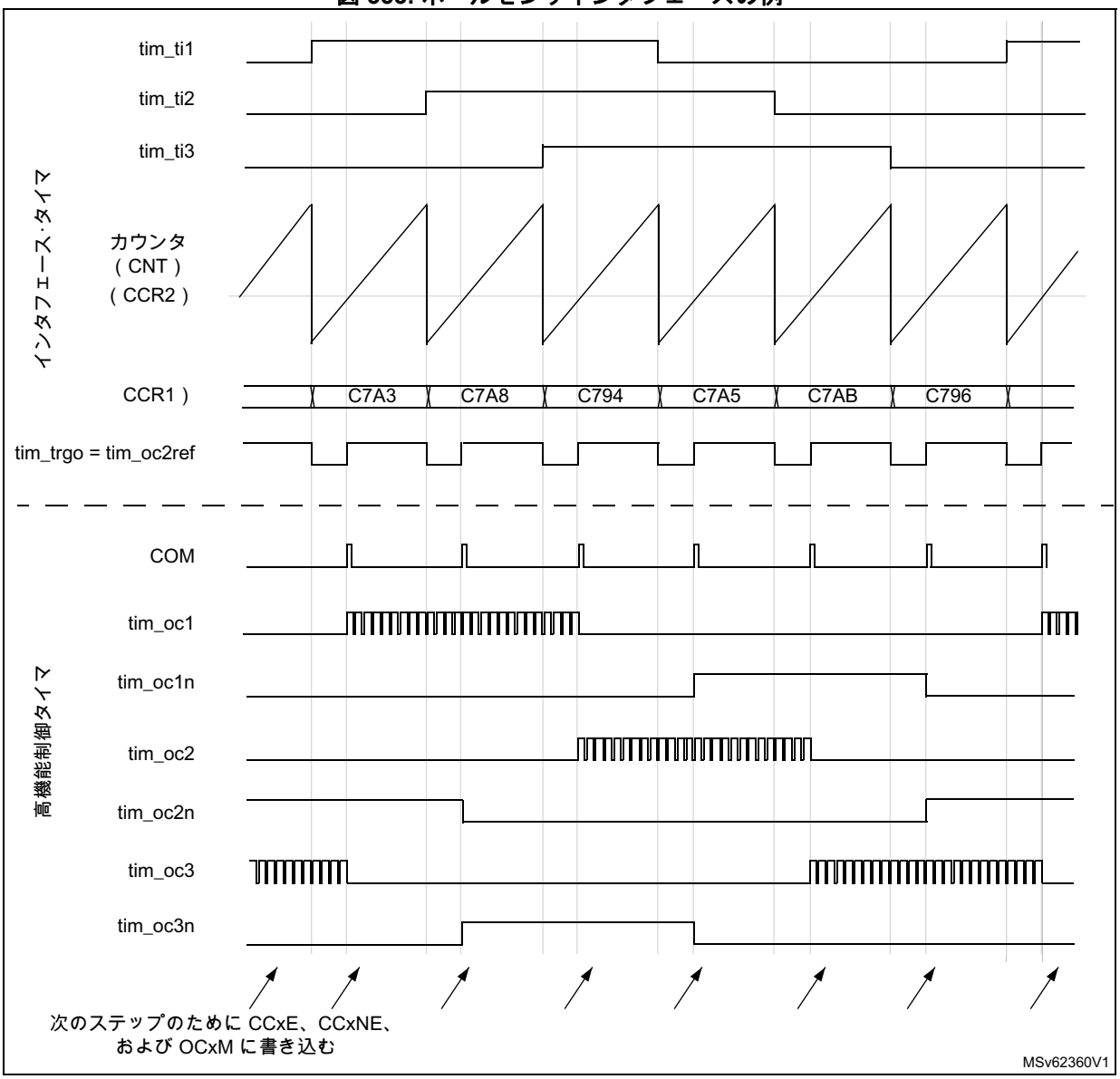
例：TIMx タイマの 1 つに接続されているホール入力が変化するたびに、プログラミングした遅延の後に高機能制御タイマの PWM 設定を変更するとします。

- 3 つのタイマ入力を tim_ti1 入力チャネルに OR 接続します。このためには、TIMx_CR2 レジスタの TI1S ビットに“1”を書き込みます。
- タイムベースをプログラムします。このためには、TIMx_ARR に最大値を書き込み、tim_ti1 の変化でカウンタがクリアされるようにします。最大カウンタ時間がセンサの 2 回の変化の間の時間より長くなるように、プリスケーラを設定します。
- チャネル 1 をキャプチャモード（tim_trc 選択）にプログラムします。すなわち、TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。必要な場合は、デジタルフィルタをプログラムすることもできます。
- チャネル 2 を PWM 2 モードにプログラミングし、希望の遅延を指定します。このためには、TIMx_CCMR1 レジスタの OC2M ビットに“111”を、CC2S ビットに“00”を書き込みます。
- tim_trgo 上のトリガ出力として tim_oc2ref を選択します。このためには、TIMx_CR2 レジスタの MMS ビットに“101”を書き込みます。

高機能制御タイマで、トリガ入力として適切な tim_itrx 入力を選択する必要があり、タイマが PWM 信号を生成するようにプログラミングし、キャプチャ /比較制御信号がプリロードされ（TIMx_CR2 レジスタの CCPC=1）、COM イベントがトリガ入力によって制御されなければなりません（TIMx_CR2 レジスタの CCUS=1）。PWM 制御ビット（CCxE、OCxM）は、COM イベント後に次のステップのために書き込まれます（これは、tim_oc2ref の立ち上がりエッジによって生成される割込みサブルーチンで行うことができます）。

図 353 に、この例を示します。

図 353. ホールセンサインタフェースの例



28.3.30 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。詳細については、セクション 29.4.23：タイマの同期を参照してください。これらのタイマは、いくつかのモードで同期状態にすることができます。すなわち、リセットモード、ゲートモード、トリガモード、リセット + トリガ モード、およびゲート + リセットモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

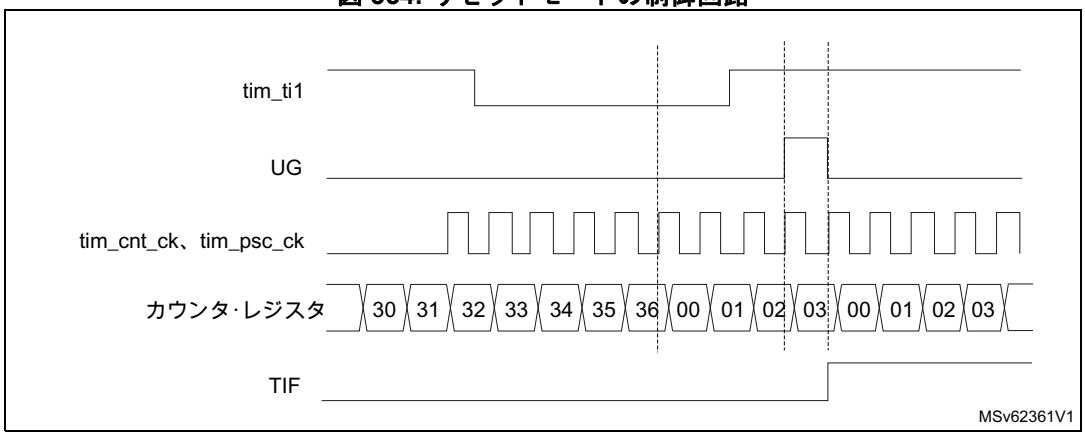
次の例では、tim_ti1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

- tim_ti1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタに CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1 の立ち上がりエッジまで通常の動作を行います。tim_ti1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。tim_ti1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1 入力の同期回路によるものです。

図 354. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

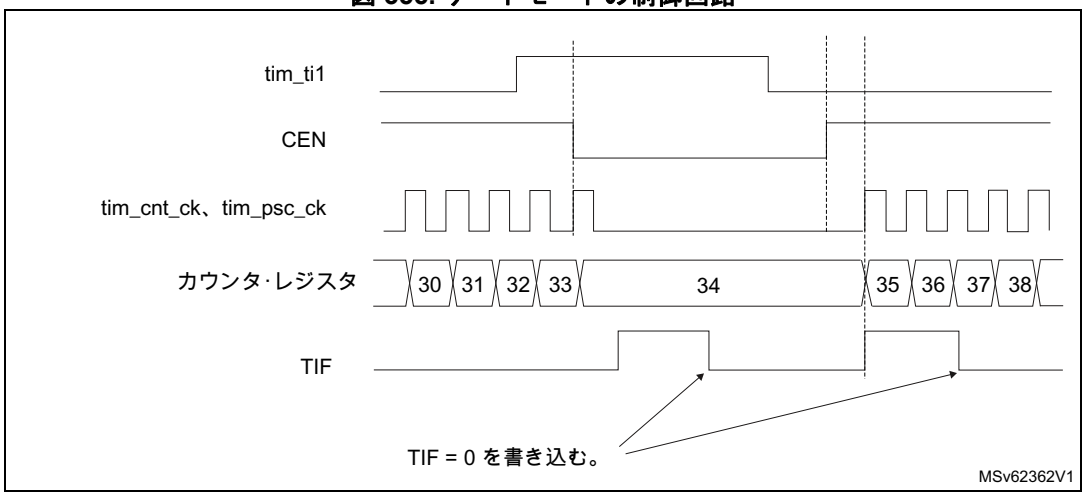
次の例では、アップカウンタは tim_ti1 入力がローのときだけカウントします。

- tim_ti1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタでCC1P= 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1 がローになると内部クロックでカウントを開始して、tim_ti1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

tim_ti1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1 入力の再同期回路によるものです。

図 355. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

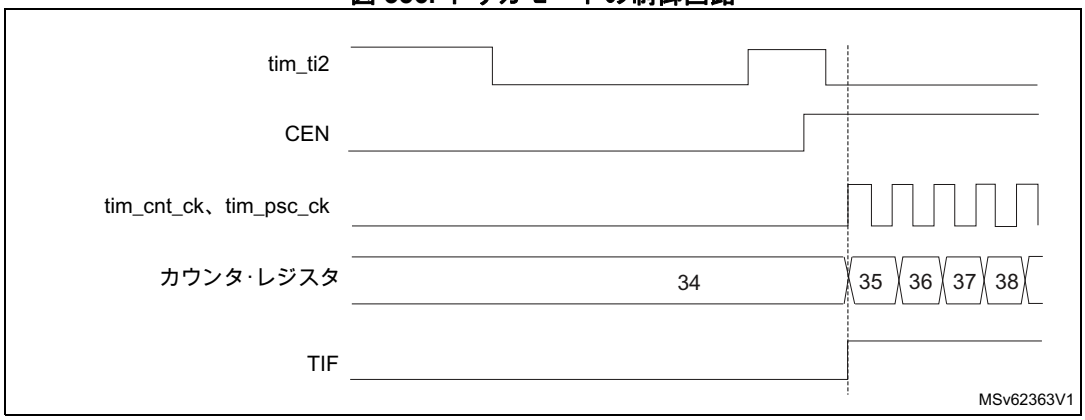
次の例では、アップカウンタは、tim_ti2 入力の立ち上がりエッジに応じて開始します。

- tim_ti2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
- TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして tim_ti2 を選択します。

tim_ti2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2 入力の再同期回路によるものです。

図 356. トリガモードの制御回路



スレーブモード：リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルス·モードで使用します。

スレーブ·モード：ゲートモードとリセットモードの組み合わせ

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードによって、範囲外の PWM 信号（最大期待値を超えるデューティサイクル）を検出できます。

スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、tim_etr_in 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、またはトリガモード）。TIMx_SMCR レジスタの TS ビットを通じて tim_trgi として tim_etr_in を選択しないようにしてください。

次の例では、アップカウンタは、tim_ti1 の立ち上がりエッジが発生すると、tim_etr_in 信号の立ち上がりエッジのたびにインクリメントされます。

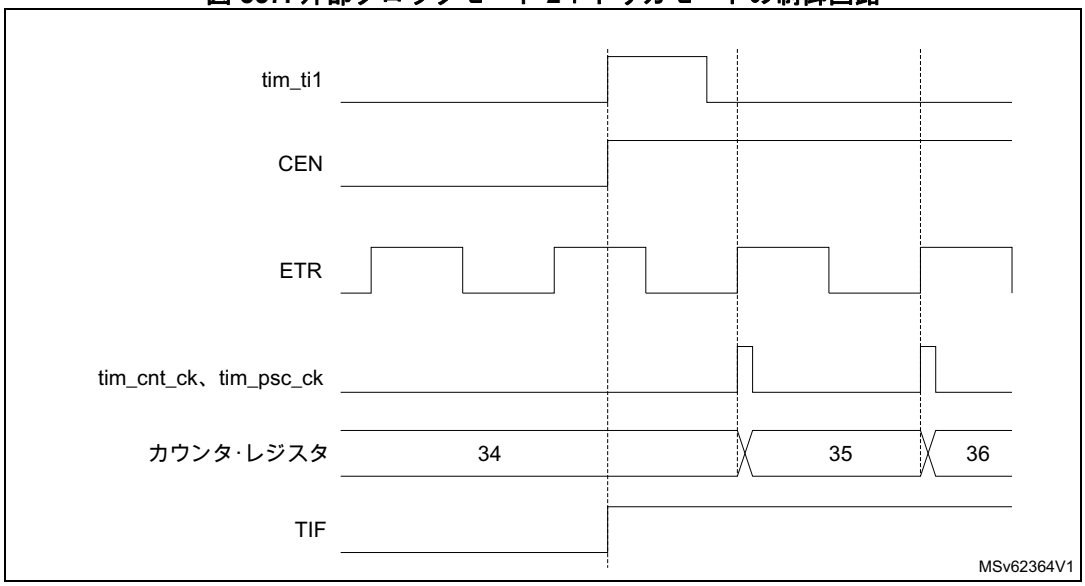
1. TIMx_SMCR レジスタで次のようにプログラミングすることによって､外部トリガ入力回路を構成します。
   - ETF = 0000：フィルタなし
   - ETPS=00：プリスケーラ無効
   - ETP=0：tim_etr_in の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   - IC1F=0000：フィルタなし。
   - キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。

3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。

tim_ti1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、その後 tim_etr_in の立ち上がりエッジでカウントします。

tim_etr_in 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_etrp 入力の再同期回路によるものです。

図 357. 外部クロックモード 2＋トリガモードの制御回路



注： **tim_trgo または tim_trgo2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

28.3.31 ADC トリガ

タイマでは、リセット·イベント、イネーブル·イベント、比較イベントなどのさまざまな内部信号で ADC トリガ·イベントを生成できます。以下のような内部エッジ検出から発行されたパルスを生成することもできます。

- OC4ref の立ち上がりおよび立ち下がりエッジ
- OC5ref の立ち上がりエッジまたは OC6ref の立ち下がりエッジ

トリガは、ADC にリダイレクトされる tim_trgo2 内部ラインで発行されます。使用できるイベントは全部で 16 個あり、TIMx_CR2 レジスタの MMS2[3:0] ビットを使用して選択できます。

3 相モータ駆動のアプリケーションの例については、1100 ページの図 311 を参照してください。

注： **tim_trgo または tim_trgo2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

注： **ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。**

28.3.32 DMA バースト･モード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効にします。
5. DMA チャネルを有効にします。

この例は、各CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

28.3.33 TIM1/TIM8/TIM20DMA リクエスト

下表に示すように、TIM1/TIM8/TIM20 は DMA リクエストを生成できます。

表 263. DMA リクエスト

DMA リクエスト信号	DMA 略称	DMA リクエスト	有効化制御ビット
tim_upd_dma	TIM_UP	更新	UDE
tim_cc1_dma	TIM_CH1	キャプチャ／比較 1	CC1DE
tim_cc2_dma	TIM_CH2	キャプチャ／比較 2	CC2DE
tim_cc3_dma	TIM_CH3	キャプチャ／比較 3	CC3DE
tim_cc4_dma	TIM_CH4	キャプチャ／比較 4	CC4DE
tim_com_dma	TIM_COM	転流（COM）	COMDE
tim_trg_dma	TIM_TRIG	トリガ	TDE

28.3.34 デバッグ･モード

マイクロコントローラがデバッグ･モードになると（FPU 搭載 Cortex®-M4 コアは停止状態）、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

安全のため、カウンタが停止すると、出力は無効になります（MOE ビットのリセット時と同じ）。通常強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

28.4 TIM1/TIM8/TIM20 低電力モード

表 264. 低消費電力モードが TIM1/TIM8/TIM20 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスは SLEEP モードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

28.5 TIM1/TIM8/TIM20 割込み

表 265 に示すように、TIM1/TIM8/TIM20 は複数の割込みを生成できます。

表 265. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
TIM_UP	更新	UIF	UIE	UIF に 0 を書き込む	可能	不可
TIM_CC	キャプチャ／比較 1	CC1IF	CC1IE	CC1IF に 0 を書き込む	可能	不可
	キャプチャ／比較 2	CC2IF	CC2IE	CC2IF に 0 を書き込む	可能	不可
	キャプチャ／比較 3	CC3IF	CC3IE	CC3IF に 0 を書き込む	可能	不可
	キャプチャ／比較 4	CC4IF	CC4IE	CC4IF に 0 を書き込む	可能	不可
TIM_TRG_COM	転流（COM）	COMIF	COMIE	COMIF に 0 を書き込む	可能	不可
	トリガ	TIF	TIE	TIF に 0 を書き込む	可能	不可
TIM_DIR_IDX	インデックス	IDXF	IDXIE	IDXF に 0 を書き込む	可能	不可
	方向	DIRF	DIRIE	DIRF に 0 を書き込む	可能	不可
TIM_BRK	ブレーク	BIF	BIE	BIF に 0 を書き込む	可能	不可
	Break2	B2IF		B2IF に 0 を書き込む	可能	不可
	システム･ブレーク	SBIF		SBIF に 0 を書き込む	可能	不可
TIM_IERR	インデックスエラー	IERRF	IERRIE	IERRF に 0 を書き込む	可能	不可
TIM_TER	遷移エラー	TERRF	TERRIE	TERRF に 0 を書き込む	可能	不可

28.6 TIM1/TIM8/TIM20 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

28.6.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 1、8、20）

アドレス･オフセット：0x000

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DITHEN**：ディザリング･イネーブル

0：ディザリングは無効です。
1：ディザリングは有効です。

注： DITHEN ビットは、CEN ビットがリセットされているときのみ変更できます。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。
1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_etr_in、tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{tim_ker_ck}$
01：$t_{DTS}=2*t_{tim_ker_ck}$
10：$t_{DTS}=4*t_{tim_ker_ck}$
11：予約済み - この値をプログラミングしないでください。

ビット 7 **ARPE**：自動再ロード･プリロード･イネーブル

0：TIMx_ARR レジスタはバッファされません。
1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。
01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。
11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット 3 **OPM**：ワンパルス･モード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエスト･ソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディスエーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタ･イネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

28.6.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 1、8、20）

アドレス･オフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5
						rw		rw	rw	rw	rw		rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OIS4N	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **MMS2[3:0]**：マスタモード選択 2

これらのビットにより、選択される同期（tim_trgo2）について、ADC に情報を送信できるようになります。組み合わせは、次のとおりです。

0000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（tim_trgo2）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo2 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力（tim_trgo2）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、tim_trgo2 には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo2）として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（tim_trgo2）。

0100：**比較** - tim_oc1refc 信号がトリガ出力（tim_trgo2）として使用されます。

0101：**比較** - tim_oc2refc 信号がトリガ出力（tim_trgo2）として使用されます。

0110：**比較** - tim_oc3refc 信号がトリガ出力（tim_trgo2）として使用されます。

0111：**比較** - tim_oc4refc 信号がトリガ出力（tim_trgo2）として使用されます。

1000：**比較** - tim_oc5refc 信号がトリガ出力（tim_trgo2）として使用されます。

1001：**比較** - tim_oc6refc 信号がトリガ出力（tim_trgo2）として使用されます。

1010：**パルス比較** - tim_oc4refc の立ち上がりエッジまたは立ち下がりエッジによって、tim_trgo2 にパルスが生成されます。

1011：**パルス比較** - tim_oc6refc の立ち上がりエッジまたは立ち下がりエッジによって、tim_trgo2 にパルスが生成されます。

1100：**パルス比較** - tim_oc4refc または tim_oc6refc の立ち上がりエッジによって、tim_trgo2 にパルスが生成されます。

1101：**パルス比較** - tim_oc4refc の立ち上がりエッジまたは tim_oc6refc の立ち下がりエッジによって、tim_trgo2 にパルスが生成されます。

1110：**パルス比較** - tim_oc5refc または tim_oc6refc の立ち上がりエッジによって、tim_trgo2 にパルスが生成されます。

1111：**パルス比較** - tim_oc5refc の立ち上がりエッジまたは tim_oc6refc の立ち下がりエッジによって、tim_trgo2 にパルスが生成されます。

注： スレーブ･タイマまたは ADC のクロックは、マスタ･タイマからイベントを受信する前に有効化する必要があり、マスタ･タイマからトリガを受信している間は動作中に変更しないでください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **OIS6**：出力アイドル状態 6（tim_oc6 出力）
OIS1 ビットの説明を参照してください。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **OIS5**：出力アイドル状態 5（tim_oc5 出力）
OIS1 ビットの説明を参照してください。

ビット 15 **OIS4N**：出力アイドル状態 4（tim_oc4n 出力）
OIS1N ビットの説明を参照してください。

ビット 14 **OIS4**：出力アイドル状態 4（tim_oc4 出力）
OIS1 ビットの説明を参照してください。

ビット 13 **OIS3N**：出力アイドル状態 3（tim_oc3n 出力）
OIS1N ビットの説明を参照してください。

ビット 12 **OIS3**：出力アイドル状態 3（tim_oc3n 出力）
OIS1 ビットの説明を参照してください。

ビット 11 **OIS2N**：出力アイドル状態 2（tim_oc2n 出力）
OIS1N ビットの説明を参照してください。

ビット 10 **OIS2**：出力アイドル状態 2（tim_oc2 出力）
OIS1 ビットの説明を参照してください。

ビット 9 **OIS1N**：出力アイドル状態 1（tim_oc1n 出力）
0：MOE=0 のとき、デッドタイム後に tim_oc1n=0
1：MOE=0 のとき、デッドタイム後に tim_oc1n=1
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 8 **OIS1**：出力アイドル状態 1（tim_oc1 出力）
0：MOE=0 のとき、tim_oc1=0（デッドタイム後に）
1：MOE=0 のとき、tim_oc1=1（デッドタイム後に）
注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 7 **TI1S**：tim_ti1 選択
0：tim_ti1_in[15:0] マルチプレクサ出力が tim_ti1 入力に接続されます。
1：tim_ti1_in[15:0]、tim_ti2_in[15:0]、および tim_ti3_in[15:0] マルチプレクサ出力が XOR 演算されて tim_ti1 入力に接続されます。

ビット 25、6:4 **MMS[3:0]**：マスタ･モード選択

これらのビットにより、同期を目的としてマスタ･モードでスレーブ･タイマに送信する情報を選択できます（tim_trgo）。組み合わせは、次のとおりです。

0000：**リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタ･イネーブル信号 CNT_EN がトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタ･イネーブル信号は、ゲート･モードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタ･イネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブ･モードが選択されている場合を除き、tim_trgo には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します（tim_trgo）。

0100：**比較** - tim_oc1refc 信号がトリガ出力（tim_trgo）として使用されます。

0101：**比較** - tim_oc2refc 信号がトリガ出力（tim_trgo）として使用されます。

0110：**比較** - tim_oc3refc 信号がトリガ出力（tim_trgo）として使用されます。

0111：**比較** - tim_oc4refc 信号がトリガ出力（tim_trgo）として使用されます。

1000：**エンコーダ･クロック出力** - エンコーダ･クロック信号がトリガ出力（tim_trgo）として使用されます。このコードは次の SMS[3:0] 値に対して有効です。0001、0010、0011、1010、1011、1100、1101、1110、1111。その他の SMS[3:0] コードは許されず、予想外の動作につながることがあります。

その他のコードは予約済み

注： スレーブタイマまたは ADC のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされている場合（CCPC=1）、COMG ビットのセットまたは tim_trgi の立ち上がりエッジによって更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM の各ビットがプリロードされます。書込み後のこれらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUS ビットの状態に応じて、COMG ビットのセットまたは tim_trgi の立ち上がりエッジ検出で更新）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

28.6.3 TIMx スレーブ･モード制御レジスタ（TIMx_SMCR）（x = 1、8、20）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
						rw	rw			rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **SMSPS**：SMS プリロード･ソース

このビットは、プリロードからアクティブへの SMS[3:0] ビットフィールドの転送をトリガするイベントのいずれかを選択します。

0：転送は、タイマの更新イベントによってトリガされます。

1：転送は、インデックス･イベントによってトリガされます。

ビット 24 **SMSPE**：SMS プリロード･イネーブル

このビットは、SMS[3:0] ビットフィールドをプリロードするかどうかを選択します。

0：SMS[3:0] ビットフィールドはプリロードされません。

1：SMS[3:0] プリロードは有効です。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 21:20 **TS[4:3]**：トリガ選択 - ビット 4:3

TS[2:0] 説明を参照 - ビット 6:4

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ETP**：外部トリガ極性

このビットは、tim_etr_in と tim_$\overline{\text{etr_in}}$ のいずれがトリガ動作に使用されるかを選択します。

0：tim_etr_in は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：tim_etr_in は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE**：外部クロック･イネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、tim_etrf 信号のアクティブエッジによってクロック供給されます。

注： **ECE ビットをセットすることは、tim_trgi が tim_etrf に接続された状態で外部クロック･モード 1 を選択することと同じ効果があります（SMS = 111、TS = 00111）。**

外部クロック･モード 2 とスレーブ モードであるリセット･モード、ゲート･モード、またはトリガ･モードを同時に使用できます。ただし、この場合、tim_trgi を tim_etrf に接続することはできません（TS ビットが 00111 でないことが必要）。

外部クロック･モード 1 と外部クロック･モード 2 が同時に有効な場合、外部クロック入力は tim_etrf です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 tim_etrp の周波数は、TIMxCLK 周波数の 1/4 までに制限されます。プリスケーラを有効にすると、tim_etrp 周波数を低減できます。これは、高速な外部クロックを tim_etr_in に入力するときに役立ちます。

00：プリスケーラオフ

01：tim_etr_in 周波数は 2 分周されます。

10：tim_etr_in 周波数は 4 分周されます。

11：tim_etr_in 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、tim_etrp 信号をサンプルする周波数と、tim_etrp に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ／スレーブ･モード

0：影響なし。

1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgo を通じて）完全な同期状態が得られるまで遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS[2:0]**：トリガ選択

このビットフィールドは TS[4:3] ビットと結合されます。

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000 : 内部トリガ 0（tim_itr0）
00001 : 内部トリガ 1（tim_itr1）
00010 : 内部トリガ 2（tim_itr2）
00011 : 内部トリガ 3（tim_itr3）
00100 : tim_ti1 エッジ検出回路（tim_ti1f_ed）
00101 : フィルタタイマ入力 1（tim_ti1fp1）
00110 : フィルタタイマ入力 2（tim_ti2fp2）
00111 : 外部トリガ入力（tim_etrf）
01000 : 内部トリガ 4（tim_itr4）
01001 : 内部トリガ 5（tim_itr5）
01010 : 内部トリガ 6（tim_itr6）
01011 : 内部トリガ 7（tim_itr7）
01100 : 内部トリガ 8（tim_itr8）
01101 : 内部トリガ 9（tim_itr9）
01110 : 内部トリガ 10（tim_itr10）
01111 : 内部トリガ 11（tim_itr11）
10000 : 内部トリガ 12（tim_itr12）
10001 : 内部トリガ 13（tim_itr13）
10010 : 内部トリガ 14（tim_itr14）
10011 : 内部トリガ 15（tim_itr15）
その他：予約済みです。

各タイマにおける tim_itrx の意味の詳細については、表 251 : TIMx 内部トリガ接続を参照してください。

注： 設定を変更するときに誤ったエッジが検出されないように、これらのビットは、使用されていないとき（SMS = 000 のときなど）にのみ変更する必要があります。

ビット 3 **OCCS**：OCREF クリア選択

このビットは、OCREF クリアソースを選択するために使用されます。

0：tim_ocref_clr_int は tim_ocref_clr 入力に接続されます。
1：tim_ocref_clr_int は tim_etrf に接続されます。

ビット 16、2:0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：直交エンコーダ･モード 1、x2 モード - カウンタは、tim_ti2fp2 のレベルに応じて、tim_ti1fp1 のエッジでカウント･アップ／ダウンします。

0010：直交エンコーダ･モード 2、x2 モード - カウンタは、tim_ti1fp1 のレベルに応じて、tim_ti2fp2 のエッジでカウント･アップ／ダウンします。

0011：直交エンコーダ･モード 3、x4 モード - カウンタは、他の入力のレベルに応じて、tim_ti1fp1 と tim_ti2fp2 の両方のエッジでカウント･アップ/ダウンします。

0100：リセット･モード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲート･モード - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ･モード - カウンタは、トリガ tim_trgi の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロック･モード 1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセット･モードとトリガ･モードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲート･モードとリセット･モードの組み合わせ - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

1010：エンコーダ･モード：クロックプラス方向、x2 モード。

1011：エンコーダ･モード：クロックプラス方向、x1 モード。tim_ti2fp2 エッジ検出は CC2P によって設定されます。

1100：エンコーダ･モード：方向クロック、x2 モード。

1101：エンコーダ･モード：方向クロック、x1 モード。tim_ti1fp1 と tim_ti2fp2 エッジ検出は CC1P と CC2P によって設定されます。

1110：直交エンコーダ･モード：x1 モード、tim_ti1fp1 エッジのみでカウント、エッジ検出は CC1P によって設定されます。

1111：直交エンコーダ･モード：x1 モード、tim_ti2fp2 エッジのみでカウント、エッジ検出は CC2P によって設定されます。

注： トリガ入力として tim_ti1f_ed が選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_ed は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgo または tim_trgo2 の信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

28.6.4 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 1、8、20）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.
								rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TERRIE**：遷移エラー割込みイネーブル
0：遷移エラー割込みは無効です。
1：遷移エラー割込みは有効です。

ビット 22 **IERRIE**：インデックス･エラー割込みイネーブル
0：インデックス･エラー割込みは無効です。
1：インデックス･エラー割込みは有効です。

ビット 21 **DIRIE**：方向変更割込みイネーブル
0：方向変更割込みは無効です。
1：方向変更割込みは有効です。

ビット 20 **IDXIE**：インデックス割込みイネーブル
0：インデックス割込みは無効です。
1：インデックス変更割込みは有効です。

ビット 19:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエスト･イネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエスト･イネーブル
0：COM DMA リクエストは無効です。
1：COM DMA リクエストは有効です。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエスト･イネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット 6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 **COMIE**：COM 割込みイネーブル
0：COM 割込み無効
1：COM 割込み有効

ビット 4 **CC4IE**：キャプチャ／比較 4 割込みイネーブル
0：CC4 割込みは無効です。
1：CC4 割込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割込みイネーブル
0：CC3 割込みは無効です。
1：CC3 割込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル
0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

28.6.5 TIMx ステータスレジスタ（TIMx_SR）（x = 1、8、20）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	CC6IF	CC5IF
								rc_w0	rc_w0	rc_w0	rc_w0			rc_w0	rc_w0
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TERRF**：遷移エラー割込みフラグ
このフラグは、エンコーダモードで遷移エラーが検出されたときに、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：エンコーダ遷移エラーは検出されていません。
1：エンコーダ遷移エラーが検出されています。

ビット 22 **IERRF**：インデックス･エラー割込みフラグ
このフラグは、インデックスエラーが検出されたときに、ハードウェアによってセットされます。これは、ソフトウェアで“0”を書き込むことによってクリアされます。
0：インデックス･エラーは検出されていません。
1：インデックス･エラーが検出されています。

ビット 21 **DIRF**：方向変更割込みフラグ

このフラグは、エンコーダモードで方向が変わった（TIMx_CR の DIR ビット値が変わった）ときに、ハードウェアによってセットされます。これは、ソフトウェアで“0”を書き込むことによってクリアされます。

0：方向の変更なし

1：方向の変更

ビット 20 **IDXF**：インデックス割込みフラグ

このフラグは、インデックスイベントが検出されたときに、ハードウェアによってセットされます。これは、ソフトウェアで“0”を書き込むことによってクリアされます。

0：インデックス･イベントは発生していません。

1：インデックス･イベントが発生しています。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC6IF**：比較 6 割込みフラグ

CC1IF の説明を参照してください。

注： チャネル 6 は出力としてのみ設定できます。

ビット 16 **CC5IF**：比較 5 割込みフラグ

CC1IF の説明を参照してください。

注： チャネル 5 は出力としてのみ設定できます。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **SBIF**：システム･ブレーク割込みフラグ

このフラグは、システムブレーク入力がアクティブになると、ハードウェアによってセットされます。システムブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

PWM 動作をリスタートするには、このフラグをリセットする必要があります。

0：ブレークイベントは発生していません。

1：システムブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバーキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 **B2IF**：ブレーク 2 割込みフラグ

このフラグは、ブレーク 2 入力がアクティブになると、ハードウェアによってセットされます。ブレーク 2 入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク 2 入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 7 **BIF**：ブレーク割込みフラグ

このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0 : ブレークイベントは発生していません。

1 : ブレーク入力でアクティブレベルが検出されました。TIMx_DIER レジスタの BIE=1 の場合、割込みが生成されます。

ビット 6 **TIF**：トリガ割込みフラグ

このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4 **CC4IF**：キャプチャ／比較 4 割込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタ･オーバーフロー時（アップカウント･モードおよびアップ／ダウンカウント･モードの場合）、またはアンダーフロー時（ダウンカウント･モードの場合）に CC1IF ビットはハイになります。センターアライン･モードでのフラグの設定には 3 つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–繰り返しカウンタ値に関するオーバーフローまたはアンダーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

–TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（セクション 28.6.3：TIMx スレーブ･モード制御レジスタ（TIMx_SMCR）（x = 1、8、20）を参照）。

28.6.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 1、8、20）

アドレスオフセット：0x014

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **B2G**：ブレーク 2 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレーク 2 イベントが生成されます。MOE ビットがクリアされ、B2IF フラグがセットされます。有効な場合は、関連する割込みが発生します。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 **COMG**：キャプチャ／比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4 **CC4G**：キャプチャ／比較 4 生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0 :影響なし。

1 :カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

28.6.7 TIMx キャプチャ／比較モード･レジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 1、8、20）

アドレス･オフセット：0x018

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード：

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00 : CC2 チャネルは出力として設定されます。

01 : CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。

10 : CC2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。

11 : CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、tim_ti1 入力をサンプリングする周波数と、tim_ti1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000 : フィルタなし、f_{DTS} でサンプリング

0001 : $f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010 : $f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011 : $f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100 : $f_{SAMPLING}= f_{DTS}/2$、N = 6

0101 : $f_{SAMPLING}= f_{DTS}/2$、N = 8

0110 : $f_{SAMPLING}= f_{DTS}/4$、N = 6

0111 : $f_{SAMPLING}= f_{DTS}/4$、N = 8

1000 : $f_{SAMPLING}= f_{DTS}/8$、N = 6

1001 : $f_{SAMPLING}= f_{DTS}/8$、N = 8

1010 : $f_{SAMPLING}= f_{DTS}/16$、N = 5

1011 : $f_{SAMPLING}= f_{DTS}/16$、N = 6

1100 : $f_{SAMPLING}= f_{DTS}/16$、N = 8

1101 : $f_{SAMPLING}= f_{DTS}/32$、N = 5

1110 : $f_{SAMPLING}= f_{DTS}/32$、N = 6

1111 : $f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00 : CC1 チャネルは出力として設定されます。

01 : CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

10 : CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。

11 : CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

28.6.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 1、8、20）

アドレス･オフセット：0x018

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		OC1 CE	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC2CE：**出力比較 2 クリアイネーブル

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。

10：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。

11：CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7 **OC1CE：**出力比較 1 クリアイネーブル

0：tim_oc1ref は tim_ocref_clr_int 信号の影響を受けません。

1：tim_oc1ref は tim_ocref_clr_int 信号のハイレベルが検出されるとクリアされます（tim_ocref_clr 入力または tim_etrf 入力）。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、tim_oc1 および tim_oc1n が導き出される出力基準信号 tim_oc1ref の動作を定義します。tim_oc1ref はアクティブハイですが、tim_oc1 および tim_oc1n のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：比較が一致した場合、チャネル 1 をアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にハイになります。

0010：比較が一致した場合、チャネル 1 をインアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、tim_oc1ref は反転します。

0100：強制非アクティブレベル - tim_oc1ref は強制的にローになります。

0101：強制アクティブレベル - tim_oc1ref は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（tim_oc1ref=“0”）に、そうでない場合はアクティブ（tim_oc1ref=“1”）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - tim_oc1ref は、PWM モード 1 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理和（OR）です。

1101：組み合わせ PWM モード 2 - tim_oc1ref は、PWM モード 2 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理積（AND）です。

1110：非対称 PWM モード 1 - tim_oc1ref は、PWM モード 1 と同様に動作します。tim_oc1refc は、カウンタがカウント･アップするときに tim_oc1ref を出力し、カウント･ダウンするときに tim_oc2ref を出力します。

1111：非対称 PWM モード 2 - tim_oc1ref は、PWM モード 2 と同様に動作します。tim_oc1refc は、カウンタがカウント･アップするときに tim_oc1ref を出力し、カウント･ダウンするときに tim_oc2ref を出力します。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

注： PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタのCCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0 :TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1 :TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S="00"（チャネルは出力に設定）のときには、変更できません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0 :CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1 :トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

10：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。

11：CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

28.6.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 1、8、20）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 3 と出力比較モードでのチャネル 4）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F[3:0]**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC[1:0]**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti4 に配置されます。

10：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti3 に配置されます。

11：CC4 チャネルは入力として設定され、tim_ic4 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7:4 **IC3F[3:0]**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC[1:0]**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti3 に配置されます。

10：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti4 に配置されます。

11：CC3 チャネルは入力として設定され、tim_ic3 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

28.6.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 1、8、20）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 3 と出力比較モードでのチャネル 4）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4 CE	OC4M [2:0]			OC4 PE	OC4 FE	CC4S [1:0]		OC3 CE	OC3M [2:0]			OC3 PE	OC3 FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 24、14:12 **OC4M[3:0]**：出力比較 4 モード

OC3M[3:0] ビットの説明を参照してください。

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti4 に配置されます。

10：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti3 に配置されます。

11：CC4 チャネルは入力として設定され、tim_ic4 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル

ビット 16、6:4 **OC3M[3:0]**：出力比較 3 モード

これらのビットは、tim_oc3 および tim_oc3n が導き出される出力基準信号 tim_oc3ref の動作を定義します。tim_oc3ref はアクティブハイですが、tim_oc3 および tim_oc3n のアクティブレベルは CC3P および CC3NP ビットに依存します。

0000 : 停止 - 出力比較レジスタ TIMx_CCR3 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001 : 比較が一致した場合、チャネル 3 をアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 3（TIMx_CCR3）と一致すると、tim_oc3ref 信号は強制的にハイになります。

0010 : 比較が一致した場合、チャネル 3 をインアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 3（TIMx_CCR3）と一致すると、tim_oc3ref 信号は強制的にローになります。

0011 : 反転 - TIMx_CNT = TIMx_CCR3 のとき、tim_oc3ref は反転します。

0100 : 強制非アクティブレベル - tim_oc3ref は強制的にローになります。

0101 : 強制アクティブレベル - tim_oc3ref は強制的にハイになります。

0110 : PWM モード 1 - カウント･アップ時、チャネル 3 は、TIMx_CNT < TIMx_CCR3 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 3 は、TIMx_CNT> TIMx_CCR3 の場合はインアクティブ（tim_oc3ref=“0”）に、そうでない場合はアクティブ（tim_oc3ref=“1”）になります。

0111 : PWM モード 2 - カウントアップ時、チャネル 3 は、TIMx_CNT < TIMx_CCR3 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 3 は、TIMx_CNT> TIMx_CCR3 の場合はアクティブに、そうでない場合は非アクティブになります。

1000 : 再トリガ可能な OPM モード 1 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001 : 再トリガ可能な OPM モード 2 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010 : 比較によるパルス：TIMxECR の PWPRSC[2:0] ビットフィールドと PW[7:0] ビットフィールドのプログラミング内容に従って、CCR3 一致イベント時にパルスが tim_oc3ref に生成されます。

1011 : 方向の出力。tim_oc3ref 信号は DIR ビットのコピーによって上書きされます。

1100 : 組み合わせ PWM モード 1 - tim_oc3ref は、PWM モード 1 と同様に動作します。tim_oc3refc は tim_oc3ref と tim_oc4ref の論理和（OR）です。

1101 : 組み合わせ PWM モード 2 - tim_oc3ref は、PWM モード 2 と同様に動作します。tim_oc3refc は tim_oc3ref と tim_oc4ref の論理積（AND）です。

1110 : 非対称 PWM モード 1 - tim_oc3ref は、PWM モード 1 と同様に動作します。tim_oc3refc は、カウンタがカウント･アップするときに tim_oc3ref を出力し、カウント･ダウンするときに tim_oc4ref を出力します。

1111 : 非対称 PWM モード 2 - tim_oc3ref は、PWM モード 2 と同様に動作します。tim_oc3refc は、カウンタがカウント･アップするときに tim_oc3ref を出力し、カウント･ダウンするときに tim_oc4ref を出力します。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

注： PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタのCCPC ビットがセットされた場合、OC3M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00 :CC3 チャネルは出力として設定されます。

01 :CC3 チャネルは入力として設定され、tim_ic3 は tim_ti3 に配置されます。

10 :CC3 チャネルは入力として設定され、tim_ic3 は tim_ti4 に配置されます。

11 :CC3 チャネルは入力として設定され、tim_ic3 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

28.6.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8、20）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E
										rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	CC4NE	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CC6P**：キャプチャ /比較 6 出力極性
CC1P の説明を参照してください。

ビット 20 **CC6E**：キャプチャ /比較 6 出力イネーブル
CC1E の説明を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CC5P**：キャプチャ /比較 5 出力極性
CC1P の説明を参照してください。

ビット 16 **CC5E**：キャプチャ /比較 5 出力イネーブル
CC1E の説明を参照してください。

ビット 15 **CC4NP**：キャプチャ /比較 4 相補出力極性
CC1NP の説明を参照してください。

ビット 14 **CC4NE**：キャプチャ /比較 4 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 13 **CC4P**：キャプチャ /比較 4 出力極性
CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ /比較 4 出力イネーブル
CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ /比較 3 相補出力極性
CC1NP の説明を参照してください。

ビット 10 **CC3NE**：キャプチャ /比較 3 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 9 **CC3P**：キャプチャ /比較 3 出力極性
CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ /比較 3 出力イネーブル
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ /比較 2 相補出力極性
CC1NP の説明を参照してください。

ビット 6 **CC2NE**：キャプチャ /比較 2 相補出力イネーブル
CC1NE の説明を参照してください。

ビット 5 **CC2P**：キャプチャ /比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ /比較 2 出力イネーブル
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ /比較 1 相補出力極性
CC1 チャネルが出力として設定されている場合：
0：tim_oc1n はアクティブハイです。
1：tim_oc1n はアクティブローです。
CC1 チャネルが入力として設定されている場合：
このビットは、tim_ti1fp1とtim_ti2fp1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされ、（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力として設定）になった直後は書込みできません。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル
0：オフ - tim_oc1n はアクティブではありません。tim_oc1n のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E の各ビットによって決まります。
1：オン - tim_oc1n 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NE アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 1 **CC1P**：キャプチャ /比較 1 出力極性

0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）

1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）

CC1 チャネルを入力として設定している場合は、CC1NP/CC1P の両ビットで、トリガまたはキャプチャ動作における TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1：非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP = 1、CC1P = 0：この設定は予約済みです。使用できません。

注： このビットは、LOCK レベル 2 または 3 をプログラムした時点で書込みできなくなります（TIMx_BDTR レジスタの LOCK ビット）。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ /比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 266 を参照してください。

注： 相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1E アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

表 266. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	tim_ocx 出力状態	tim_ocxn 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0、tim_ocxn=0	
		0	0	1	出力無効（タイマによって駆動されない： ハイインピーダンス） tim_ocx=0	tim_ocxref + 極性 tim_ocxn = tim_ocxref xor CCxNP
		0	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocxn=0
		X	1	1	OCREF + 極性 + デッドタイム	OCREF に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） tim_ocx=CCxP	tim_ocxref + 極性 tim_ocxn = tim_ocxref xor CCxNP
		1	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	オフ状態（インアクティブ状態で出力有効） tim_ocxn=CCxNP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイ･インピーダンス）	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期に：tim_ocx=CCxP、tim_ocxn=CCxNP（tim_brk または tim_brk2 がトリガされた場合）。 クロックが存在する場合、（tim_brk がトリガされた場合にのみ有効）：アクティブな状態で OISx と OISxN が OCX と tim_ocxn にそれぞれ対応しないとみなされる場合（ハーフブリッジ設定でスイッチを駆動した場合に短絡の原因となる）、デッドタイム後に tim_ocx=OISx および tim_ocxn=OISxN となります。 **注： tim_brk2 は OSSI = OSSR = 1 の場合にのみ使用できます。**	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO が制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP はクリアされたままでなければなりません。

注： 相補 tim_ocx および tim_ocxn チャネルに接続されている外部入出力ピンの状態は、tim_ocx および tim_ocxn チャネルの状態と、GPIO レジスタに依存します。

28.6.12 TIMx カウンタ（TIMx_CNT）（x = 1、8、20）

アドレスオフセット：0x024

リセット値：0x0000 0000

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

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

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMxCR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分は得られません。

28.6.13 TIMx プリスケーラ（TIMx_PSC）（x = 1、8、20）

アドレスオフセット：0x028

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（$f_{tim_cnt_ck}$）は $f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

28.6.14 TIMx自動再ロード･レジスタ（TIMx_ARR）（x = 1、8、20）

アドレスオフセット：0x02C

リセット値：0x0000 FFFF

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

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **ARR[19:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 28.3.3 : 1062 ページのタイムベース･ユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、自動再ロード値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

28.6.15 TIMx 繰り返しカウンタ･レジスタ（TIMx_RCR）（x = 1、8、20）

アドレスオフセット：0x030

リセット値：0x0000

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

ビット 15:0 **REP[15:0]**：繰り返しカウンタの再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰り返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返しカウンタは、繰り返し更新イベント UEV でのみ、REP 値で再ロードされるので、TIMx_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）は次のことを意味します。

–エッジアラインモードでは、PWM 周期の数

–センターアラインモードでは、PWM の 1/2 周期の数

28.6.16 TIMx キャプチャ／比較レジスタ 1 （TIMx_CCR1）（x = 1、8、20）

アドレス･オフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR1[19:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc1 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1 チャネルが入力として設定されている場合：CR1 は、最後の入力キャプチャ 1 イベント（tim_ic1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

28.6.17 TIMx キャプチャ／比較レジスタ 2 （TIMx_CCR2）（x = 1、8、20）

アドレス･オフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR2[19:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc2 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR2[15:0] に比較値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2 チャネルが入力として設定されている場合：CCR2 は、最後の入力キャプチャ 2 イベント（tim_ic2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] にキャプチャ値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

28.6.18 TIMx キャプチャ／比較レジスタ 3 （TIMx_CCR3）（x = 1、8、20）

アドレス･オフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR3[19:0]**：キャプチャ /比較値

CC3 チャネルが出力として設定されている場合：CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc3 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR3[15:0] に比較値を保持します。CCR3[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:4] に整数部を保持します。CCR3[3:0] ビットフィールドにはディザリング部分が含まれます。

CC3 チャネルが入力として設定されている場合：CCR3 は、最後の入力キャプチャ 3 イベント（tim_ic3）によって転送されたカウンタ値です。TIMx_CCR3 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR3[15:0] にキャプチャ値を保持します。CCR3[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:4] にキャプチャを保持します。CCR3[3:0] ビットはリセットされています。

28.6.19 TIMx キャプチャ／比較レジスタ 4 （TIMx_CCR4）（x = 1、8、20）

アドレス･オフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR4[19:0]**：キャプチャ /比較値

CC4 チャネルが出力として設定されている場合：CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR2 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、tim_oc4 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR4[15:0] に比較値を保持します。CCR4[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:4] に整数部を保持します。CCR4[3:0] ビットフィールドにはディザリング部分が含まれます。

CC4 チャネルが入力として設定されている場合：CCR4 は、最後の入力キャプチャ 4 イベント（tim_ic4）によって転送されたカウンタ値です。TIMx_CCR4 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR4[15:0] にキャプチャ値を保持します。CCR4[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:4] にキャプチャを保持します。CCR4[3:0] ビットはリセットされています。

28.6.20 TIMx ブレークおよびデッドタイム･レジスタ（TIMx_BDTR）（x = 1、8、20）

アドレスオフセット：0x044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	BK2BID	BKBID	BK2 DSRM	BK DSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]			
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： **ビット BKBID/BK2BID/BK2P、BK2E、BK2F[3:0]、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] は、LOCK 設定に応じて書込みがロックされるので、TIMx_BDTR レジスタへの最初のアクセス時に、これらすべてを設定しなければならないことがあります。**

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **BK2BID**：ブレーク 2 双方向

BKBID の説明を参照してください

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 tim_brk は入力モードです。

1：ブレーク入力 tim_brk は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： **このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。**

注： **このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。**

ビット 27 **BK2DSRM**：ブレーク 2 解除

BKDSRM の説明を参照してください

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 tim_brk が設定されます。

1：ブレーク入力 tim_brk は解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： **このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。**

ビット 25 **BK2P**：ブレーク 2 極性

0：ブレーク入力 tim_brk2 はアクティブローです。

1：ブレーク入力 tim_brk2 はアクティブハイです。

注： **このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。**

注： **このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。**

ビット 24 **BK2E**：ブレーク 2 イネーブル

このビットは、完全なブレーク 2 保護を有効にします（図 316：ブレークおよびブレーク 2 回路の概要のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク2 機能は無効です。

1：ブレーク2 機能は有効です。

注： BRKIN2 は OSSR = OSSI = 1 の場合にのみ使用してください。

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 23:20 **BK2F[3:0]**：ブレーク 2 フィルタ

このビットフィールドは、tim_brk2 入力をサンプリングする周波数と、tim_brk2 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、tim_brk2 は非同期として動作します。

0001：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6

0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8

0110：$f_{SAMPLING} = f_{DTS}/4$、N = 6

0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8

1000：$f_{SAMPLING} = f_{DTS}/8$、N = 6

1001：$f_{SAMPLING} = f_{DTS}/8$、N = 8

1010：$f_{SAMPLING} = f_{DTS}/16$、N = 5

1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6

1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8

1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5

1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6

1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk 入力をサンプリングする周波数と、tim_brk に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、tim_brk は非同期で動作します。

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、ブレーク入力の 1 つ（tim_brk または tim_brk2）がアクティブになると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：ブレーク 2 イベントへの対応。OC および OCN 出力は無効です。
ブレークイベントへの対応、または MOE が 0 に書き込まれた場合：OC および OCN 出力が無効か、OSSI ビットによって強制的にアイドル状態になります。

1：OC および OCN 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。
詳細については、OC/OCN イネーブルの説明を参照してください（セクション 28.6.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8、20））。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアでセットできるほか、次回の更新イベントで自動的にセットできます（ブレーク入力 tim_brk と tim_brk2 のどちらもアクティブでないこと）。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 tim_brk はアクティブ･ローです。

1：ブレーク入力 tim_brk はアクティブ･ハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレーク･イネーブル

このビットは、完全なブレーク保護を有効にします（図 316：ブレークおよびブレーク 2 回路の概要のように bk_acth および BKIN ソースにそれぞれ接続されたすべてのソースを含む）。

0：ブレーク機能は無効です。

1：ブレーク機能は有効です。

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 28.6.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8、20））。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、CCxE=1 または CCxNE=1 になると、OC/OCN 出力は、インアクティブレベルで有効になります（出力は引き続きタイマで制御される）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドル･モードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルでのブレークイベントやソフトウェアの書込みによって使用されます。

詳細については、OC/OCN イネーブルの説明を参照してください（セクション 28.6.11：TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 1、8、20））。

0：インアクティブのとき、OC/OCN 出力は無効です（タイマは出力の制御を解除し、ハイインピーダンス状態を強制する GPIO ロジックによって引き継がれます）。

1：インアクティブのとき、OC/OCN 出力はまず強制的にインアクティブレベルにされ、次にデッドタイム後に強制的にアイドルレベルにされます。タイマは出力の制御を保持します。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCK オフ － どのビットも書込み保護されません。

01：LOCK レベル 1 － TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx ビットと OISxN ビット、および TIMx_BDTR レジスタの BKBID/BK2BID/BKE/BKP/AOE の各ビットには書込みできなくなります。

10：LOCK レベル 2 － LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 － LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMRx レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIMx_BDTR レジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

28.6.21 TIMx キャプチャ／比較レジスタ 5 （TIMx_CCR5）（x = 1、8、20）

アドレス･オフセット：0x048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5[19:16]			
rw	rw	rw										rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR5 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **GC5C3**：チャネル 5 およびチャネル 3 のグループ化

チャネル 3 出力におけるひずみ：

0：tim_oc5ref の tim_oc3refc への影響はありません。

1：tim_oc3refc は、tim_oc3ref と tim_oc5ref の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR2 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 30 **GC5C2**：チャネル 5 およびチャネル 2 のグループ化

チャネル 2 出力におけるひずみ：

0：tim_oc5ref の tim_oc2refc への影響はありません。

1：tim_oc2refc は、tim_oc2ref と tim_oc5ref の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 29 **GC5C1**：チャネル 5 およびチャネル 1 のグループ化

チャネル 1 出力におけるひずみ：

0：oc5ref の oc1refc への影響はありません。

1：oc1refc は、oc1ref と oc5ref の論理 AND です。

このビットは、直ちに有効にするか、プリロードしておいて更新イベント後に考慮にすることができます（TIMxCCMR1 でプリロード機能を選択している場合）。

注： このひずみを組み合わせ PWM 信号に適用することもできます。

ビット 28:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR5[19:0]**：キャプチャ／比較 5 値

CCR5 は、実際のキャプチャ／比較 5 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC5PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 5 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc5 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR5[15:0] に比較値を保持します。CCR5[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR5[19:4] に整数部を保持します。CCR5[3:0] ビットフィールドにはディザリング部分が含まれます。

28.6.22 TIMx キャプチャ／比較レジスタ 6 （TIMx_CCR6）（x = 1、8、20）

アドレス･オフセット：0x04C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR6 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR6[19:0]**：キャプチャ／比較 6 値

CCR6 は、実際のキャプチャ／比較 6 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC6PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 6 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc6 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR6[15:0] に比較値を保持します。CCR6[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR6[19:4] に整数部を保持します。CCR6[3:0] ビットフィールドにはディザリング部分が含まれます。

28.6.23 TIMx キャプチャ／比較モード･レジスタ 3（TIMx_CCMR3）（x = 1、8、20）

アドレス･オフセット：0x050

リセット値：0x0000 0000

上記の CCMR1 レジスタの説明を参照してください。チャネル 5 および 6 は出力としてのみ設定できます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6 CE	OC6M [2:0]			OC6 PE	OC6FE	Res.	Res.	OC5 CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC6CE**：出力比較 6 クリアイネーブル

ビット 24、14:12 **OC6M[3:0]**：出力比較 6 モード

ビット 11 **OC6PE**：出力比較 6 プリロードイネーブル

ビット 10 **OC6FE**：出力比較 6 高速イネーブル

ビット 9:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **OC5CE :** 出力比較 5 クリアイネーブル

ビット 16、6:4 **OC5M[3:0]**：出力比較 5 モード

ビット 3 **OC5PE**：出力比較 5 プリロードイネーブル

ビット 2 **OC5FE**：出力比較 5 高速イネーブル

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

28.6.24 TIMx タイマ･デッドタイム･レジスタ 2 （TIMx_DTR2）（x = 1、8、20）

アドレス･オフセット：0x054

リセット値：0x0000 0000

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

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DTPE**：デッドタイム･プリロード･イネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 16 **DTAE**：デッドタイム非対称イネーブル

0：立ち上がりおよび立ち下がりエッジでのデッドタイムは同じで、DTG[7:0] レジスタで定義します。

1：立ち上がりエッジのデッドタイムは DTG[7:0] レジスタで定義され、立ち下がりエッジのデッドタイムは DTGF[7:0] ビットで定義されます。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジ･ジェネレータのセットアップ

このビットフィールドでは、立ち下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$･

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$･

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$･

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$･

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

28.6.25 TIMx タイマ･エンコーダ制御レジスタ（TIMx_ECR）（x = 1、8、20）

アドレスオフセット：0x058

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **PWPRSC[2:0]**：パルス幅プリスケーラ

このビットフィールドは、パルス発生回路用のクロックプリスケーラを次のように設定します。

$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$

ビット 23:16 **PW[7:0]**：パルス幅

このビットフィールドは、パルスの長さを次のように定義します。

$t_{PW} = PW[7:0] \times t_{PWG}$

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **IPOS[1:0]**：インデックス位置決め

直交エンコーダモード（SMS[3:0] = 0001、0010、0011、1110、1111）で、このビットは、インデックスイベントがカウンタをリセットする AB 入力の設定を示します。

00：AB = 00 のときにインデックスがカウンタをリセットします。
01：AB = 01 のときにインデックスがカウンタをリセットします。
10：AB = 10 のときにインデックスがカウンタをリセットします。
11：AB = 11 のときにインデックスがカウンタをリセットします。

方向クロックモードまたはクロックプラス方向モード（SMS[3:0] = 1010、1011、1100、1101）で、これらのビットは、インデックスイベントがカウンタをリセットするレベルを示します。双方向クロックモードでは、これは両方のクロック入力に適用されます。

x0：クロックが 0 のときにインデックスがカウンタをリセットします。
x1：クロックが 1 のときにインデックスがカウンタをリセットします。

注： IPOS[1] ビットは意味がありません。

ビット 5 **FIDX**：最初のインデックス

このビットは、最初のインデックスのみが考慮されるかどうかを示します。

0：インデックスは常にアクティブです。
1：最初のインデックスのみがカウンタをリセットします。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:1 **IDIR[1:0]**：インデックスの方向

このビットは、どちらの方向のときにインデックスイベントがカウンタをリセットするかを示します。

00：方向に関係なく、インデックスがカウンタをリセットします。
01：アップカウントのときのみ、インデックスがカウンタをリセットします。
10：ダウンカウントのときのみ、インデックスがカウンタをリセットします。
11：予約済みです。

注： IDR[1:0] ビットフィールドは、IE ビットがリセット状態のとき（インデクス無効時）に書き込む必要があります。

ビット 0 IEインデックス･イネーブル

このビットは、インデックスイベントがカウンタをリセットするかどうかを示します。

0：インデックスは無効です。

1：インデックスは有効です。

28.6.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 1、8、20）

アドレスオフセット：0x05C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：tim_ti4[15:0] 入力を選択します。

0000 :tim_ti4_in0TIMx_CH4

0001 :tim_ti4_in1

.....

1111 :tim_ti4_in15

相互接続一覧については、セクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：tim_ti3[15:0] 入力を選択します。

0000 :tim_ti3_in0TIMx_CH2

0001 :tim_ti3_in1

.....

1111 :tim_ti3_in15

相互接続一覧については、セクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：tim_ti2[15:0] 入力を選択します。

0000 :tim_ti2_in0TIMx_CH2

0001 :tim_ti2_in1

.....

1111 :tim_ti2_in15

相互接続一覧については、セクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：tim_ti1[15:0] 入力を選択します。
0000 :tim_ti1_in0TIMx_CH1
0001 :tim_ti1_in1
.....
1111 :tim_ti1_in15
相互接続一覧については、セクション 28.3.2 : TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

28.6.27 TIMxオルタネート機能オプション･レジスタ 1 （TIMx_AF1）（x = 1、8、20）

アドレス･オフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETRSEL[1:0]		BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：etr_in ソース選択
これらのビットは、etr_in 入力ソースを選択します。
0000 :tim_etr0TIMx_ETR 入力
0001 :tim_etr1
.....
1111 :tim_etr15
製品固有の実装については セクション 28.3.2 : TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

注： これらのビットは､LOCK レベル 1 がプログラムされている場合(TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 13 **BKCMP4P**：tim_brk_cmp4 入力極性
このビットは、tim_brk_cmp4 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。
0：tim_brk_cmp4 入力の極性を反転しません（BKP = 0 ならばアクティブ･ロー、BKP = 1 ならばアクティブ･ハイ）
1 : tim_brk_cmp4 入力の極性を反転します（BKP = 0 ならばアクティブ･ハイ、BKP = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 12 **BKCMP3P**：tim_brk_cmp3 入力極性
このビットは、tim_brk_cmp3 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。
0：tim_brk_cmp3 入力の極性を反転しません（BKP = 0 ならばアクティブ･ロー、BKP = 1 ならばアクティブ･ハイ）
1 : tim_brk_cmp3 入力の極性を反転します（BKP = 0 ならばアクティブ･ハイ、BKP = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 11 **BKCMP2P**：tim_brk_cmp2 入力極性

このビットは、tim_brk_cmp2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2 入力の極性を反転しません（BKP = 0 ならばアクティブ･ロー、BKP = 1 ならばアクティブ･ハイ）

1：tim_brk_cmp2 入力の極性を反転します（BKP = 0 ならばアクティブ･ハイ、BKP = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：tim_brk_cmp1 入力極性

このビットは、tim_brk_cmp1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1 入力の極性を反転しません（BKP = 0 ならばアクティブ･ロー、BKP = 1 ならばアクティブ･ハイ）

1：tim_brk_cmp1 入力の極性を反転します（BKP = 0 ならばアクティブ･ハイ、BKP = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：TIMx_BKIN 入力の極性

このビットは、TIMx_BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：0：TIMx_BKIN 入力の極性を反転しません（BKP = 0 ならばアクティブ･ロー、BKP = 1 ならばアクティブ･ハイ）

1：1：TIMx_BKIN 入力の極性を反転します（BKP = 0 ならばアクティブ･ハイ、BKP = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKCMP8E**：tim_brk_cmp8 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp8 を有効化します。tim_brk_cmp8 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp8 入力は無効です。

1：tim_brk_cmp8 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7 **BKCMP7E**：tim_brk_cmp7 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp7 を有効化します。tim_brk_cmp7 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp7 入力は無効です。

1：tim_brk_cmp7 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 6 **BKCMP6E**：tim_brk_cmp6 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp6 を有効化します。tim_brk_cmp6 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp6 入力は無効です。

1：tim_brk_cmp6 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 5 **BKCMP5E**：tim_brk_cmp5 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp5 を有効化します。tim_brk_cmp5 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp5 入力は無効です。

1：tim_brk_cmp5 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 4 **BKCMP4E**：tim_brk_cmp4 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp4 を有効化します。tim_brk_cmp4 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp4 入力は無効です。

1：tim_brk_cmp4 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 3 **BKCMP3E**：tim_brk_cmp3 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp3 を有効化します。tim_brk_cmp3 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp3 入力は無効です。

1：tim_brk_cmp3 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 2 **BKCMP2E**：tim_brk_cmp2 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp2 を有効化します。tim_brk_cmp2 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp2 入力は無効です。

1：tim_brk_cmp2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：tim_brk_cmp1 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp1 を有効化します。tim_brk_cmp1 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp1 入力は無効です。

1：tim_brk_cmp1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：TIMx_BKIN 入力イネーブル

このビットは、タイマの tim_brk 入力に対して TIMx_BKIN オルタネート機能入力を有効化します。TIMx_BKIN 入力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：TIMx_BKIN 入力は無効です。

1：TIMx_BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 製品固有の実装についてはセクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

28.6.28 TIMxオルタネート機能レジスタ 2 （TIMx_AF2）（x = 1、8、20）

アドレス·オフセット：0x064

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK2CMP4P	BK2CMP3P	BK2CMP2P	BK2CMP1P	BK2INP	BK2CMP8E	BK2CMP7E	BK2CMP6E	BK2CMP5E	BK2CMP4E	BK2CMP3E	BK2CMP2E	BK2CMP1E	BK2INE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **OCRSEL[2:0]**：ocref_clr ソース選択

これらのビットは、ocref_clr 入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

.....

111：tim_ocref_clr7

製品固有の情報については セクション 28.3.2：TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

注： これらのビットは､LOCK レベル 1 がプログラムされている場合(TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **BK2CMP4P**：tim_brk2_cmp4 入力極性

このビットは、tim_brk2_cmp4 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp4 入力の極性を反転しません（BK2P = 0 ならばアクティブ·ロー、BK2P = 1 ならばアクティブ·ハイ）

1：tim_brk2_cmp4 入力の極性を反転します（BK2P = 0 ならばアクティブ·ハイ、BK2P = 1 ならばアクティブ·ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 12 **BK2CMP3P**：tim_brk2_cmp3 入力極性

このビットは、tim_brk2_cmp3 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp3 入力の極性を反転しません（BK2P = 0 ならばアクティブ·ロー、BK2P = 1 ならばアクティブ·ハイ）

1：tim_brk2_cmp3 入力の極性を反転します（BK2P = 0 ならばアクティブ·ハイ、BK2P = 1 ならばアクティブ·ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット)、変更できません。

ビット 11 **BK2CMP2P**：tim_brk2_cmp2 入力極性

このビットは、tim_brk2_cmp2 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp2 入力の極性を反転しません（BK2P = 0 ならばアクティブ･ロー、BK2P = 1 ならばアクティブ･ハイ）

1：tim_brk2_cmp2 入力の極性を反転します（BK2P = 0 ならばアクティブ･ハイ、BK2P = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BK2CMP1P**：tim_brk2_cmp1 入力極性

このビットは、tim_brk2_cmp1 入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：tim_brk2_cmp1 入力の極性を反転しません（BK2P = 0 ならばアクティブ･ロー、BK2P = 1 ならばアクティブ･ハイ）

1：tim_brk2_cmp1 入力の極性を反転します（BK2P = 0 ならばアクティブ･ハイ、BK2P = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BK2INP**：TIMx_BKIN2 入力の極性

このビットは、TIMx_BKIN2 オルタネート機能入力の感度を選択します。BK2P 極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN2 入力の極性を反転しません（BK2P = 0 ならばアクティブ･ロー、BK2P = 1 ならばアクティブ･ハイ）

1：TIMx_BKIN2 入力の極性を反転します（BK2P = 0 ならばアクティブ･ハイ、BK2P = 1 ならばアクティブ･ロー）

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BK2CMP8E**：tim_brk2_cmp8 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp8 を有効化します。tim_brk2_cmp8 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp8 入力は無効です。

1：tim_brk2_cmp8 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7 **BK2CMP7E**：tim_brk2_cmp7 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp7 を有効化します。tim_brk2_cmp7 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp7 入力は無効です。

1：tim_brk2_cmp7 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 6 **BK2CMP6E**：tim_brk2_cmp6 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp6 を有効化します。tim_brk2_cmp6 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp6 入力は無効です。

1：tim_brk2_cmp6 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 5 **BK2CMP5E**：tim_brk2_cmp5 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp5 を有効化します。tim_brk2_cmp5 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp5 入力は無効です。

1：tim_brk2_cmp5 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 4 **BK2CMP4E**：tim_brk2_cmp4 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp4 を有効化します。tim_brk2_cmp4 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp4 入力は無効です。

1：tim_brk2_cmp4 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 3 **BK2CMP3E**：tim_brk2_cmp3 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp3 を有効化します。tim_brk2_cmp3 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp3 入力は無効です。

1：tim_brk2_cmp3 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 2 **BK2CMP2E**：tim_brk2_cmp2 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp2 を有効化します。tim_brk2_cmp2 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp2 入力は無効です。

1：tim_brk2_cmp2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BK2CMP1E**：tim_brk2_cmp1 イネーブル

このビットは、タイマの tim_brk2 入力に対して tim_brk2_cmp1 を有効化します。tim_brk2_cmp1 出力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：tim_brk2_cmp1 入力は無効です。

1：tim_brk2_cmp1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BK2INE**：TIMx_BKIN2 入力イネーブル

このビットは、タイマの tim_brk2 入力に対して TIMx_BKIN2 オルタネート機能入力を有効化します。TIMx_BKIN2 入力は、ほかの tim_brk2 ソースとの「論理和」がとられます。

0：TIMx_BKIN2 入力は無効です。

1：TIMx_BKIN2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： 製品固有の実装についてはセクション 28.3.2:TIM1/TIM8/TIM20 ピンおよび内部信号を参照してください。

28.6.29 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 1、8、20）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、転送数など DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000 : 1 回転送

00001 : 2 回転送

00010 : 3 回転送

.....

11010 : 26 回転送

例：次の転送を考えます：DBL = 7 バイトかつ DBA = TIM2_CR1。

–DBL = 7 バイトおよび DBA = TIM2_CR1 が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1 アドレス）+ DBA +（DMA インデックス）、ここで DMA インデックス = DBL

この例では、（TIMx_CR1 アドレス）+ DBA に 7 バイトが加算され、データのコピー元とコピー先のアドレスが得られます。この場合、転送は、以下のアドレスを先頭とした 7 つのレジスタに対して行われます。（TIMx_CR1 アドレス）+ DBA

DMA データサイズの設定に応じて、いくつかのケースが想定されます。

–DMA データ･サイズをハーフワードで設定した場合、7 つのレジスタにそれぞれ 16 ビットのデータが転送されます。

–DMA データ･サイズをバイトで設定した場合も、データは 7 つのレジスタに転送されます（最初のレジスタには最初の MSB バイトが含まれ、2 番目のレジスタには最初の LSB バイトが含まれるなど、以降同様）。したがって、タイマへの転送では、DMA で転送するデータサイズも指定する必要があります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し/書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000 : TIMx_CR1

00001 : TIMx_CR2

00010 : TIMx_SMCR

.....

28.6.30 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 1、8、20）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

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

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、（TIMx_CR1 アドレス）+（DBA + DMA インデックス）x 4 のアドレスにあるレジスタへのアクセスとなります。

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

28.6.31 TIMx レジスタマップ

TIMx レジスタは、次の表のように、16 ビットでアドレス指定できるレジスタとしてマッピングされます。

表 267. TIMxレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD [1:0]		ARPE	CMS [1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x004	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	MMS2 [3:0]				Res.	OIS6	Res.	OIS5	OIS4N	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1S	MMS [2:0]			CCDS	CCUS	Res.	CCPC
	リセット値							0		0	0	0	0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0
0x008	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS [4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS [1:0]		ETF[3:0]			MSM	TS[2:0]				OCCS	SMS[2:0]		
	リセット値							0	0			0	0				0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値									0	0	0	0						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	CC6IF	CC5IF	Res.	Res.	SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値									0	0	0	0			0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																								0	0	0	0	0	0	0	0	0
0x018	**TIMx_CCMR1** 入力キャプチャ・モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2 PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	**TIMx_CCMR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	OC4CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4 PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3 PSC [1:0]		CC3S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC6P	CC6E	Res.	Res.	CC5P	CC5E	CC4NP	CC4NE	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値											0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	**TIMx_CNT**	UIFCPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 267. TIMxレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x030	**TIMx_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	**TIMx_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	**TIMx_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	**TIMx_CCR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	**TIMx_CCR4**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	**TIMx_BDTR**	Res.	Res.	BK2BID	BKBID	BK2DSRM	BKDSRM	BK2P	BK2E	BK2F [3:0]				BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x048	**TIMx_CCR5**	GC5C3	GC5C2	GC5C1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR5 [19:0]																			
	リセット値	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	**TIMx_CCR6**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR6 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	**TIMx_CCMR3**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC6M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC5M[3]	OC6CE	OC6M [2:0]			OC6PE	OC6FE	Res.	Res.	OC5CE	OC5M [2:0]			OC5PE	OC5FE	Res.	Res.
	リセット値								0								0	0	0	0	0	0	0			0	0	0	0	0	0		

表 267. TIMxレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x054	**TIMx_DTR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x058	**TIMx_ECR**	Res.	Res.	Res.	Res.	Res.	PWPR SC[2:0]			PW[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS [1:0]		FIDX	Res.	Res.	IDIR [1:0]		IE
	リセット値						0	0	0	0	0	0	0	0	0	0	0									0	0	0			0	0	0
0x05C	**TIMx_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]		Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x060	**TIMx_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res ETRSEL [3:0]				BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x064	**TIMx_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]			Res.	Res.	BK2CMP4P	BK2CMP3P	BK2CMP2P	BK2CMP1P	BK2INP	BK2CMP8E	BK2CMP7E	BK2CMP6E	BK2CMP5E	BK2CMP4E	BK2CMP3E	BK2CMP2E	BK2CMP1E	BK2INE
	リセット値														0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x068.. 0x3D8	予約済みです。	Res.																															
0x3DC	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIMx_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては 80ページのセクション 2.2 参照してください。

29 汎用タイマ（TIM2/TIM3/TIM4/TIM5）

29.1 TIM2/TIM3/TIM4/TIM5 の概要

この汎用タイマは、プログラマブルなプリスケーラによって駆動される 16 ビットまたは 32 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較と PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマは完全に独立していて、いかなるリソースも共有しません。これらのタイマは、セクション 29.4.23：タイマの同期に示すように、相互に同期させることができます。

29.2 TIM2/TIM3/TIM4/TIM5 の主な特徴

汎用 TIMx タイマの主な機能は、次のとおりです。

- 16 ビットまたは 32 ビットのアップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1 から 65535 の間で分周する 16 ビットプログラム可能プリスケーラ。
- 次の機能を持つ、最大 4 つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモードとセンターアラインモード）
  - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー／アンダーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

29.3 TIM2/TIM3/TIM4/TIM5 の実装

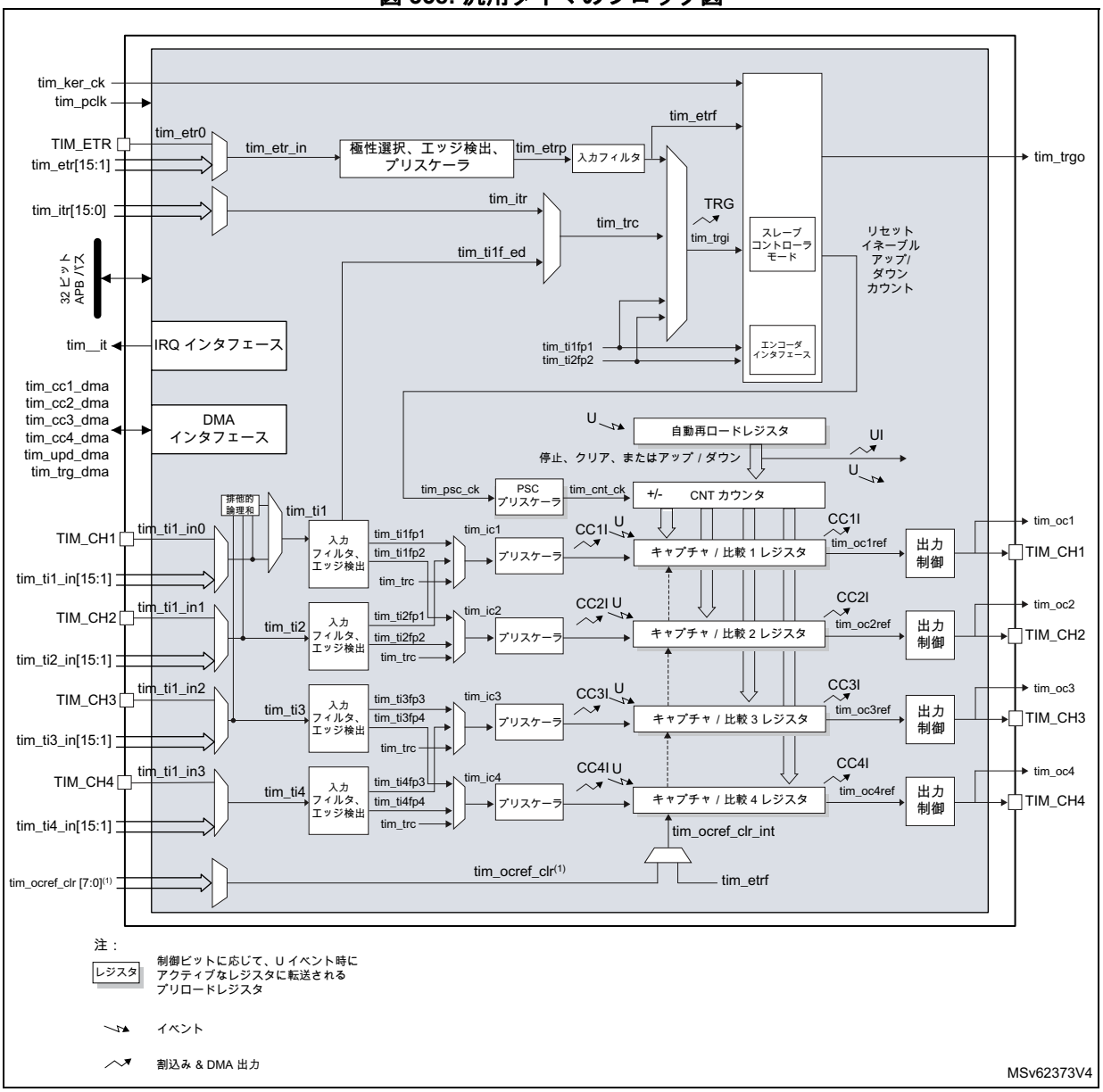
表 268. STM32G4 シリーズ汎用タイマ

タイマインスタンス	TIM2	TIM3	TIM4	TIM5
分解能	32 ビット	16 ビット	16 ビット	32 ビット
OCREF クリア選択ソース	あり tim_etrf tim_ocref_clr[7:0]	あり tim_etrf tim_ocref_clr[7:0]	なし tim_etrf -	なし tim_etrf -

29.4 TIM2/TIM3/TIM4/TIM5 機能詳細

29.4.1 ブロック図

図 358. 汎用タイマのブロック図



1. この機能は一部のタイマでは利用できません（セクション 29.3：TIM2/TIM3/TIM4/TIM5 の実装を参照）。

29.4.2 TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号

このセクションの表 269 および表 270 は、TIM の入力および出力の一覧です。

表 269. TIM の入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2 TIM_CH3 TIM_CH4	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、または PWM で使用できます。 TIM_CH1 と TIM_CH2 は、外部クロック（tim_ker_ck クロックの 1/4 未満）、外部トリガ、および直交エンコーダの入力としても使用できます。 TIM_CH1、TIM_CH2、および TIM_CH3 は、デジタルホール効果センサとのインタフェースに使用できます。
TIM_ETR	入力	外部トリガ入力。この入力 は、外部トリガまたは外部クロックソースとして使用できます。tim_etr_in プリスケーラが使用されている場合、この入力は tim_ker_ck よりも高い周波数のクロックを受信できます。

表 270. TIM 内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0] tim_ti3_in [15:0] tim_ti4_in [15:0]	入力	内部タイマ入力バス。tim_ti1_in[15:0] および tim_ti2_in[15:0] の入力は、キャプチャ用、外部クロック（tim_ker_ck クロックの 1/4 未満）、直交エンコーダ信号用として使用できます。
tim_etr[15:0]	入力	外部トリガ内部入力バス。これらの入力は、トリガ、外部クロックとして、またはハードウェアのサイクルごとのパルス幅制御に使用できます。tim_etr_in プリスケーラが使用されている場合、これらの入力は tim_ker_ck よりも高い周波数のクロックを受信できます。
tim_itr[15:0]	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ck クロックの 1/4 未満）として使用できます。
tim_trgo	出力	内部トリガ出力。このトリガは他のオンチップペリフェラルをトリガできます。
tim_ocref_clr[7:0]	入力	タイマの tim_ocref_clr 入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim_ocxref 信号をクリアするために使用できます。
tim_pclk	入力	タイマ APB クロック。
tim_ker_ck	入力	タイマ･カーネル･クロック。
tim_it	出力	キャプチャ／比較、更新、ブレークの各トリガ要求をまとめたグローバル･タイマ割込み。

表 270. TIM 内部入力／出力信号 （続き）

内部信号名	信号タイプ	説明
tim_cc1_dma tim_cc2_dma tim_cc3_dma tim_cc4_dma	出力	タイマキャプチャ／比較 1～4 DMA リクエスト。
tim_upd_dma	出力	タイマ更新 DMA リクエスト。
tim_trg_dma	出力	タイマトリガ DMA リクエスト。

表 271、表 272、表 273、および 表 274 に、tim_ti[4:1] 入力マルチプレクサに接続されるソースを列挙します。

表 271. tim_ti1 入力マルチプレクサへの相互接続

tim_ti1 入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti1_in0	TIM2_CH1	TIM3_CH1	TIM4_CH1	TIM5_CH1
tim_ti1_in1	comp1_out	comp1_out	comp1_out	LSI
tim_ti1_in2	comp2_out	comp2_out	comp2_out	LSE
tim_ti1_in3	comp3_out	comp3_out	comp3_out	RTC ウェイクアップ
tim_ti1_in4	comp4_out	comp4_out	comp4_out	comp1_out
tim_ti1_in5	comp5_out	comp5_out	comp5_out	comp2_out
tim_ti1_in6	予約済みです。	comp6_out	comp6_out	comp3_out
tim_ti1_in7		comp7_out	comp7_out	comp4_out
tim_ti1_in8		予約済みです。	予約済みです。	comp5_out
tim_ti1_in9				comp6_out
tim_ti1_in10				comp7_out
tim_ti1_in[15:11]	予約済みです。			

表 272. tim_ti2 入力マルチプレクサへの相互接続

tim_ti2 入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti2_in0	TIM2_CH2	TIM3_CH2	TIM4_CH2	TIM5_CH2
tim_ti2_in1	comp1_out	comp1_out	comp1_out	comp1_out
tim_ti2_in2	comp2_out	comp2_out	comp2_out	comp2_out
tim_ti2_in3	comp3_out	comp3_out	comp3_out	comp3_out
tim_ti2_in4	comp4_out	comp4_out	comp4_out	comp4_out
tim_ti2_in5	comp6_out	comp5_out	comp5_out	comp5_out

表 272. tim_ti2 入力マルチプレクサへの相互接続 （続き）

tim_ti2 入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti2_in6	予約済みです。	comp6_out	comp6_out	comp6_out
tim_ti2_in7		comp7_out	comp7_out	comp7_out
tim_ti2_in [15:8]	予約済みです。			

表 273. tim_ti3 入力マルチプレクサへの相互接続

tim_ti3 入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti3_in0	TIM2_CH3	TIM3_CH3	TIM4_CH3	TIM5_CH3
tim_ti3_in1	comp4_out	comp3_out	comp5_out	予約済みです。
tim_ti2_in [15:2]	予約済みです。			

表 274. tim_ti4 入力マルチプレクサへの相互接続

tim_ti4 入力	ソース			
	TIM2	TIM3	TIM4	TIM5
tim_ti4_in0	TIM2_CH4	TIM3_CH4	TIM4_CH4	TIM5_CH4
tim_ti4_in1	comp1_out	予約済みです。	comp6_out	予約済みです。
tim_ti4_in2	comp2_out		予約済みです。	
tim_ti4_in [15:3]	予約済みです。			

表 275 に、tim_itr 入力マルチプレクサに接続される内部ソースを列挙します。

表 275. TIMx 内部トリガ接続

TIMx	TIM2	TIM3	TIM4	TIM5
tim_itr0	tim1_trgo	tim1_trgo	tim1_trgo	tim1_trgo
tim_itr1	予約済みです。	tim2_trgo	tim2_trgo	tim2_trgo
tim_itr2	tim3_trgo	予約済みです。	tim3_trgo	tim3_trgo
tim_itr3	tim4_trgo	tim4_trgo	予約済みです。	tim4_trgo
tim_itr4	tim5_trgo	tim5_trgo	tim5_trgo	予約済みです。
tim_itr5	tim8_trgo	tim8_trgo	tim8_trgo	tim8_trgo
tim_itr6	tim15_trgo	tim15_trgo	tim15_trgo	tim15_trgo
tim_itr7	tim16_oc1	tim16_oc1	tim16_oc1	tim16_oc1
tim_itr8	tim17_oc1	tim17_oc1	tim17_oc1	tim17_oc1
tim_itr9	tim20_trgo	tim20_trgo	tim20_trgo	tim20_trgo
tim_itr10	hrtim_out_sync2	hrtim_out_sync2	hrtim_out_sync2	hrtim_out_sync2

表 275. TIMx 内部トリガ接続 （続き）

TIMx	TIM2	TIM3	TIM4	TIM5
tim_itr11	USB SOF SYNC	予約済みです。	予約済みです。	予約済みです。
tim_itr[15:12]	予約済みです。			

表 276 に、tim_etr 入力マルチプレクサに接続される内部ソースを列挙します。

表 276. tim_etr 入力マルチプレクサへの相互接続

タイマ外部トリガ入力信号	タイマ外部トリガ信号割り当て			
	TIM2	TIM3	TIM4	TIM5
tim_etr0	TIM2_ETR	TIM3_ETR	TIM4_ETR	TIM5_ETR
tim_etr1	comp1_out	comp1_out	comp1_out	comp1_out
tim_etr2	comp2_out	comp2_out	comp2_out	comp2_out
tim_etr3	comp3_out	comp3_out	comp3_out	comp3_out
tim_etr4	comp4_out	comp4_out	comp4_out	comp4_out
tim_etr5	comp5_out	comp5_out	comp5_out	comp5_out
tim_etr6	comp6_out	comp6_out	comp6_out	comp6_out
tim_etr7	comp7_out	comp7_out	comp7_out	comp7_out
tim_etr8	TIM3_ETR	TIM2_ETR	TIM3_ETR	TIM2_ETR
tim_etr9	TIM4_ETR	TIM4_ETR	TIM5_ETR	TIM3_ETR
tim_etr10	TIM5_ETR	予約済みです。	予約済みです。	予約済みです。
tim_etr11	LSE	adc2_awd1		
tim_etr12	予約済みです。	adc2_awd2		
tim_etr13		adc2_awd3		
tim_etr[15:14]	予約済みです。			

表 277 に、tim_ocref_clr 入力マルチプレクサに接続される内部ソースを列挙します。

表 277. tim_ocref_clr 入力マルチプレクサへの相互接続

タイマ tim_ocref_clr 信号	タイマ tim_ocref_clr 信号割当て			
	TIM2	TIM3	TIM4	TIM5
tim_ocref_clr0	comp1_out	comp1_out	予約済みです。	予約済みです。
tim_ocref_clr1	comp2_out	comp2_out		
tim_ocref_clr2	comp3_out	comp3_out		
tim_ocref_clr3	comp4_out	comp4_out		
tim_ocref_clr4	comp5_out	comp5_out		
tim_ocref_clr5	comp6_out	comp6_out		
tim_ocref_clr6	comp7_out	comp7_out		
tim_ocref_clr7	予約済みです。			

29.4.3 タイムベース･ユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビット／32 ビットカウンタです。カウンタはカウント･アップ、カウント･ダウン、またはカウント･アップダウンします。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）:
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力 tim_cnt_ck から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビット／32 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 359 と 図 360 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 359. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

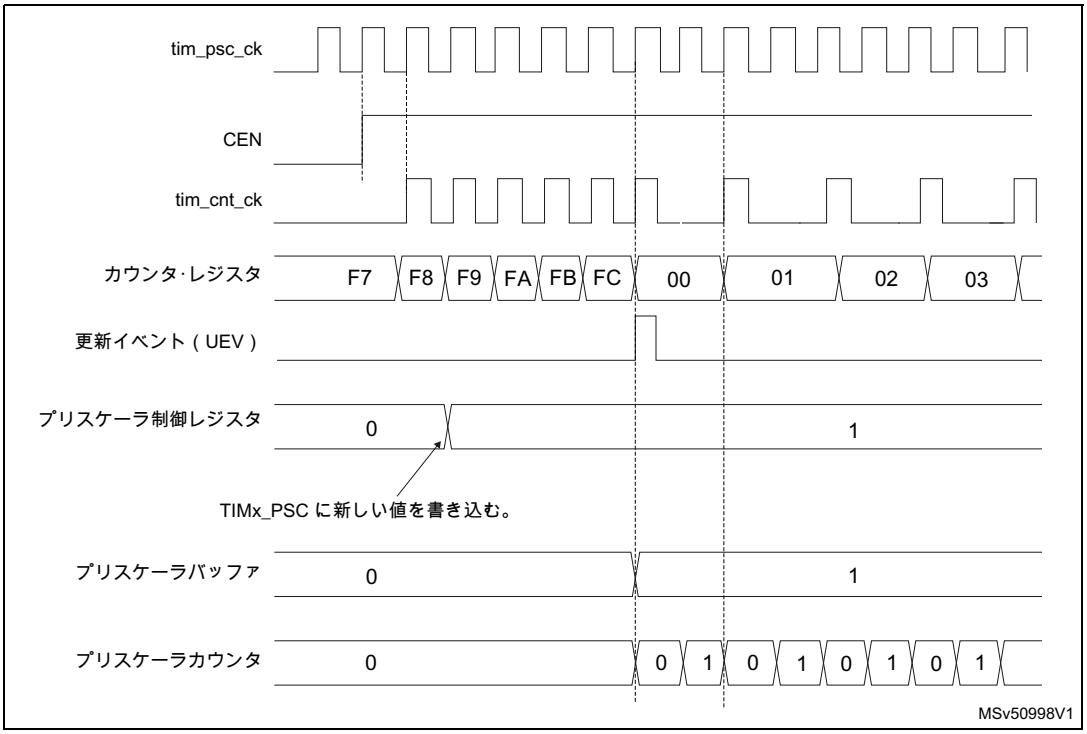
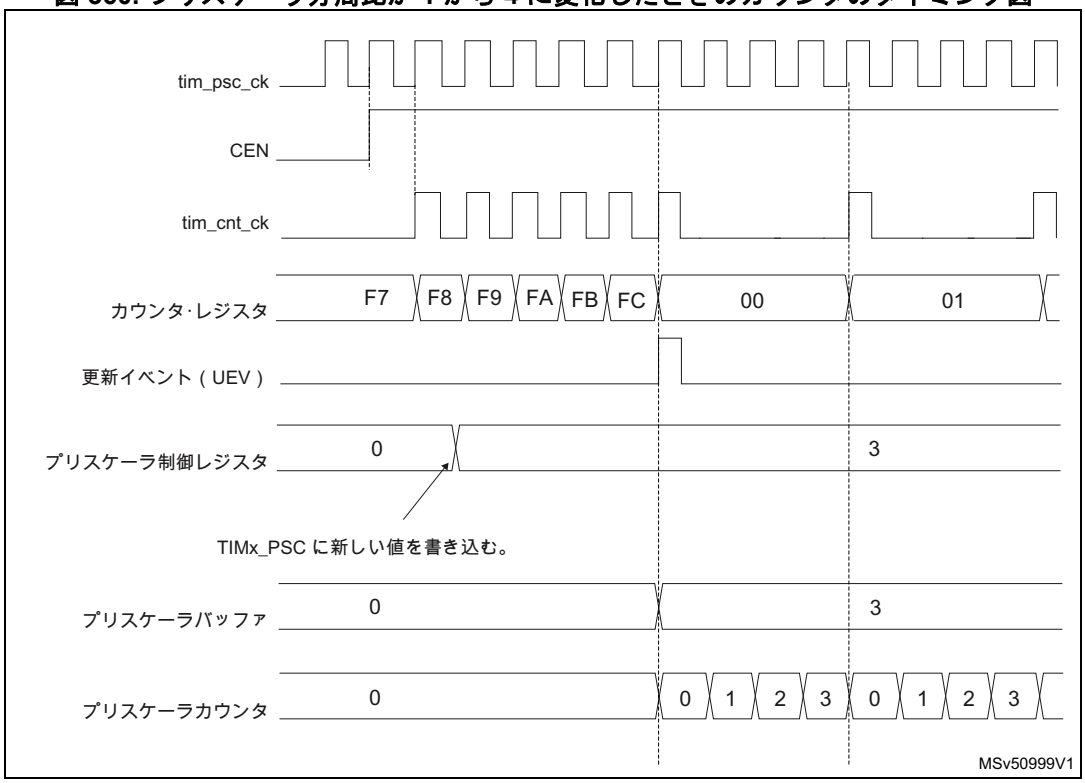


図 360. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



29.4.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることで生成できます。

UEV イベントは、TIMx_CR1 レジスタの UDIS ビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 361. 内部クロック分周比が 1 の場合のカウンタのタイミング図

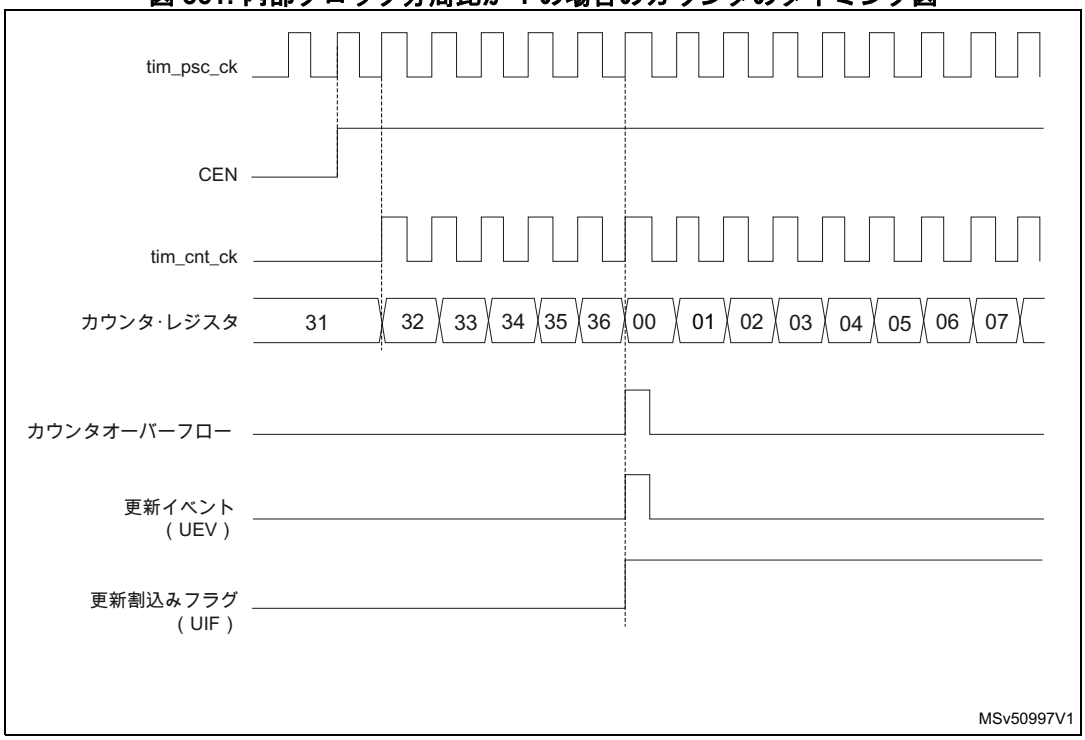


図 362. 内部クロック分周比が 2 の場合のカウンタのタイミング図

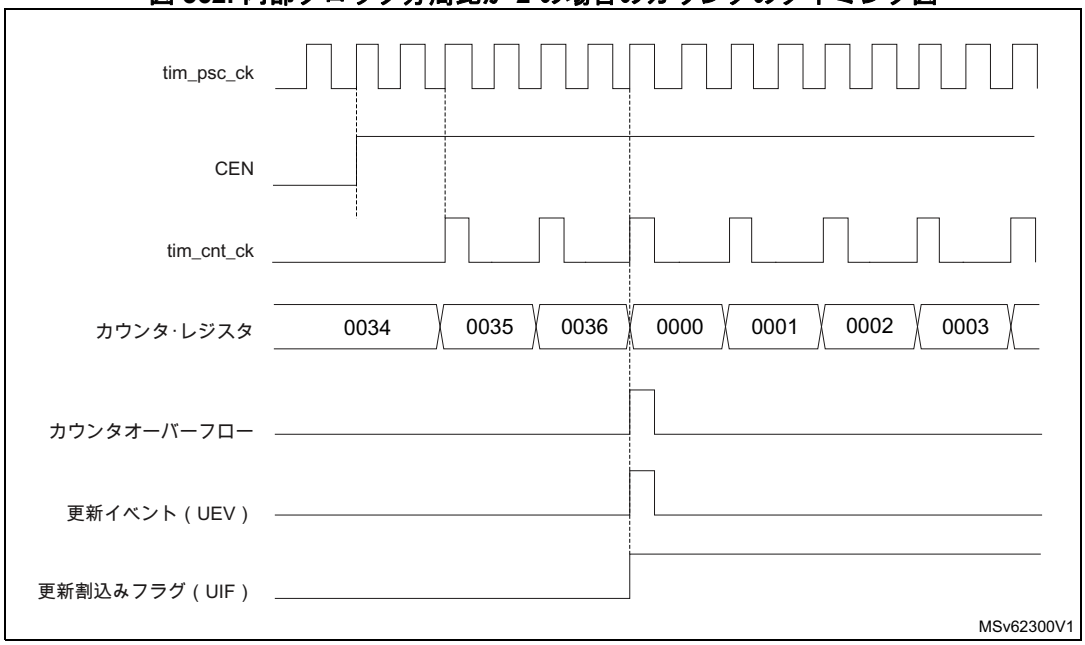


図 363. 内部クロック分周比が 4 の場合のカウンタのタイミング図

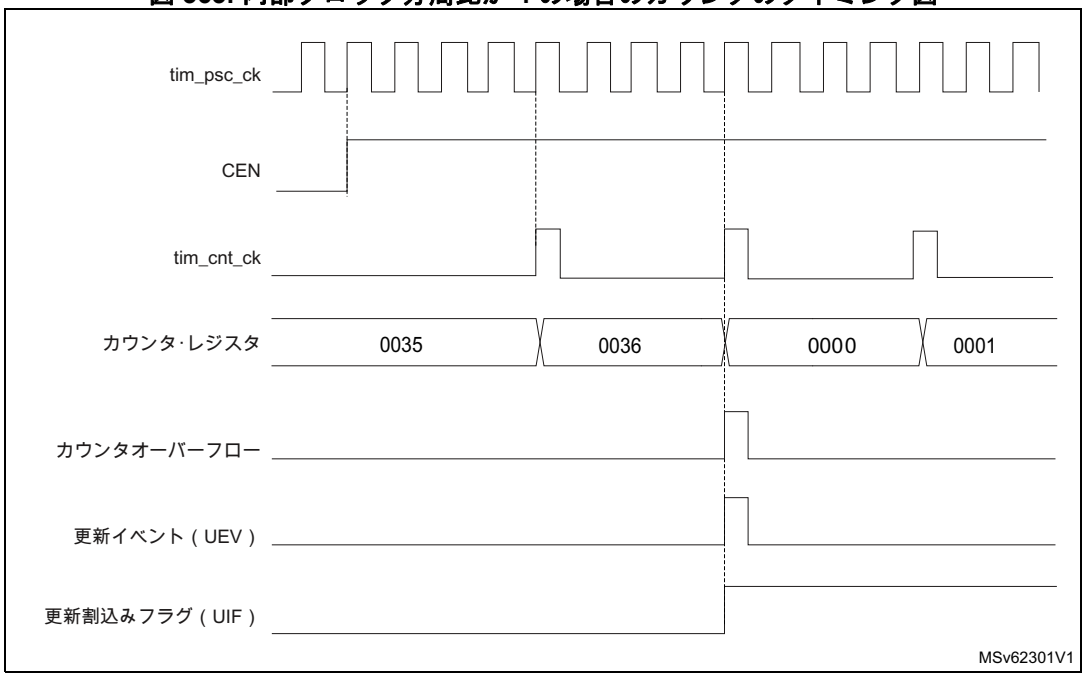


図 364. 内部クロック分周比が N の場合のカウンタのタイミング図

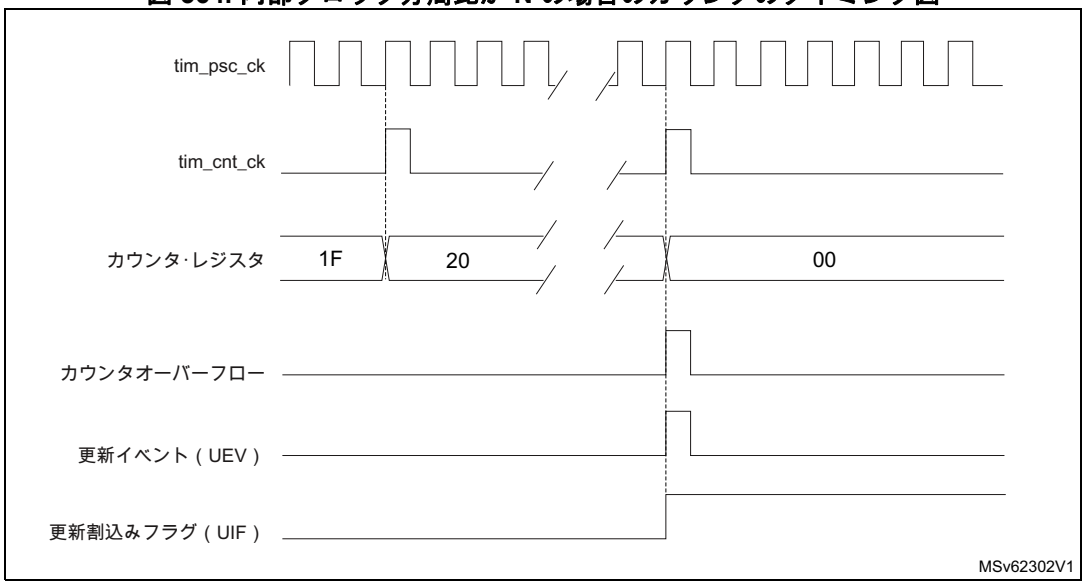


図 365. ARPE=0（TIMx_ARR はプリロードされない）の場合の更新イベント時のカウンタのタイミング図

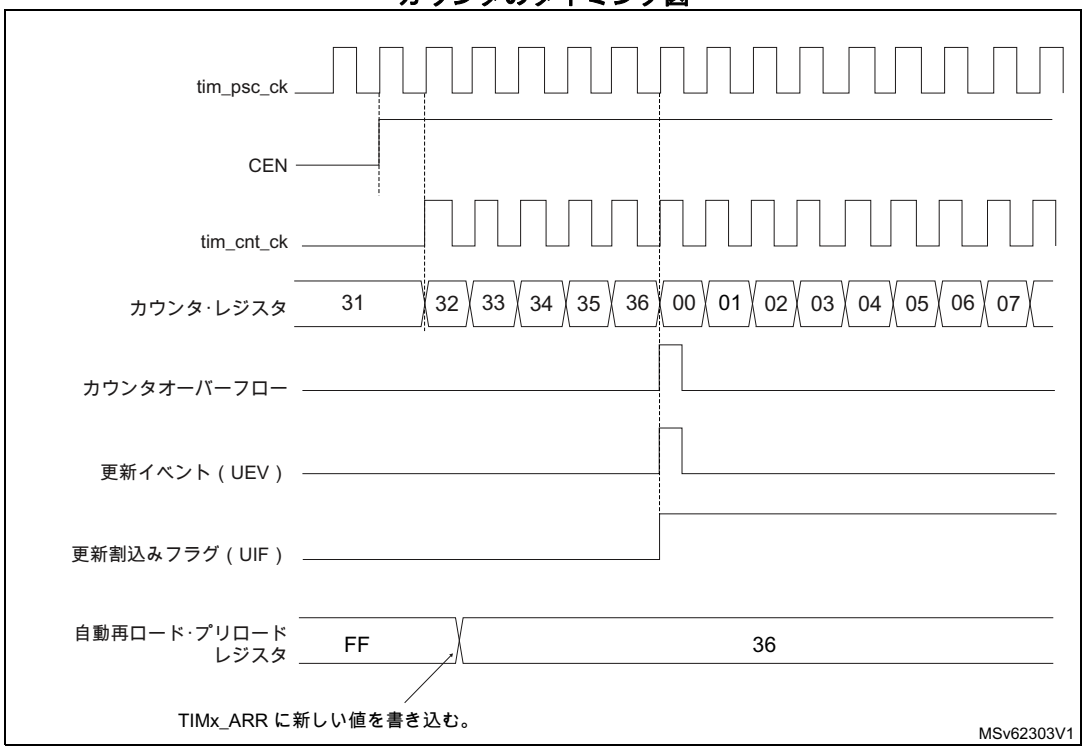
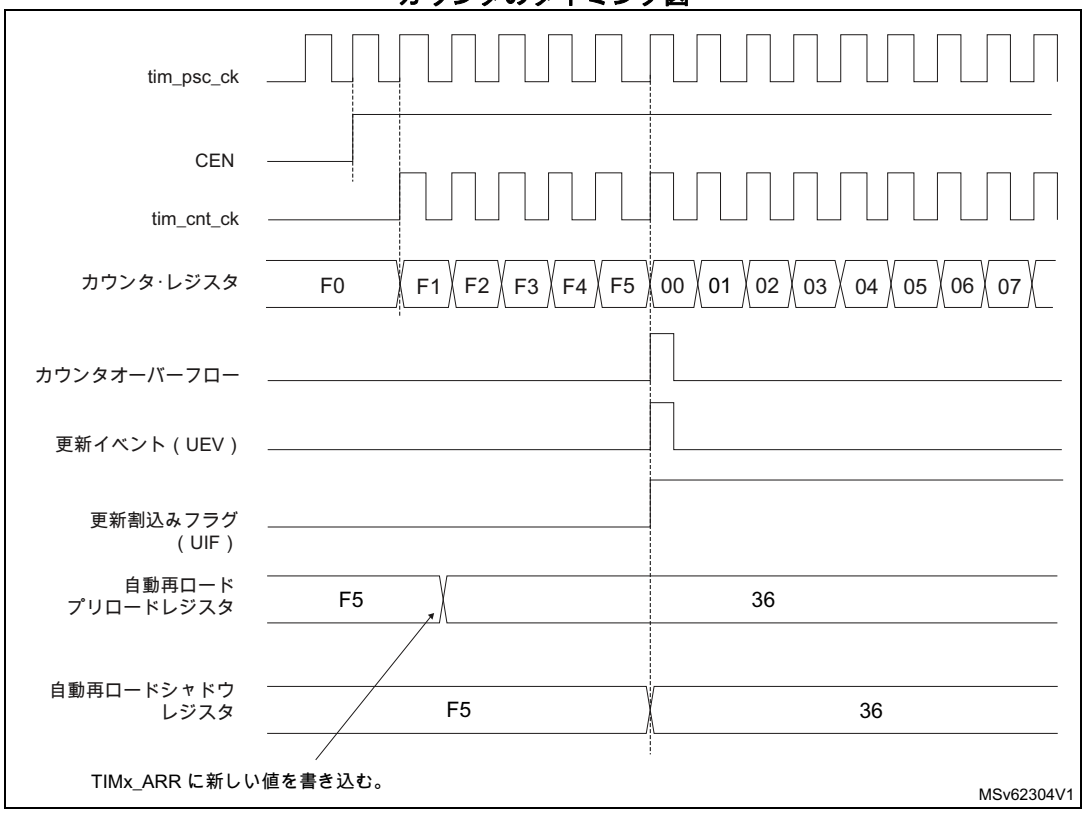


図 366. ARPE=1（TIMx_ARR はプリロードされる）のときの更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値（TIMx_ARR レジスタの内容）から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、（ソフトウェアで、または、スレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケーラのカウンタは 0 からリスタートします（しかし、プリスケーラ比は変化しません）。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 367. 内部クロック分周比が 1 の場合のカウンタのタイミング図

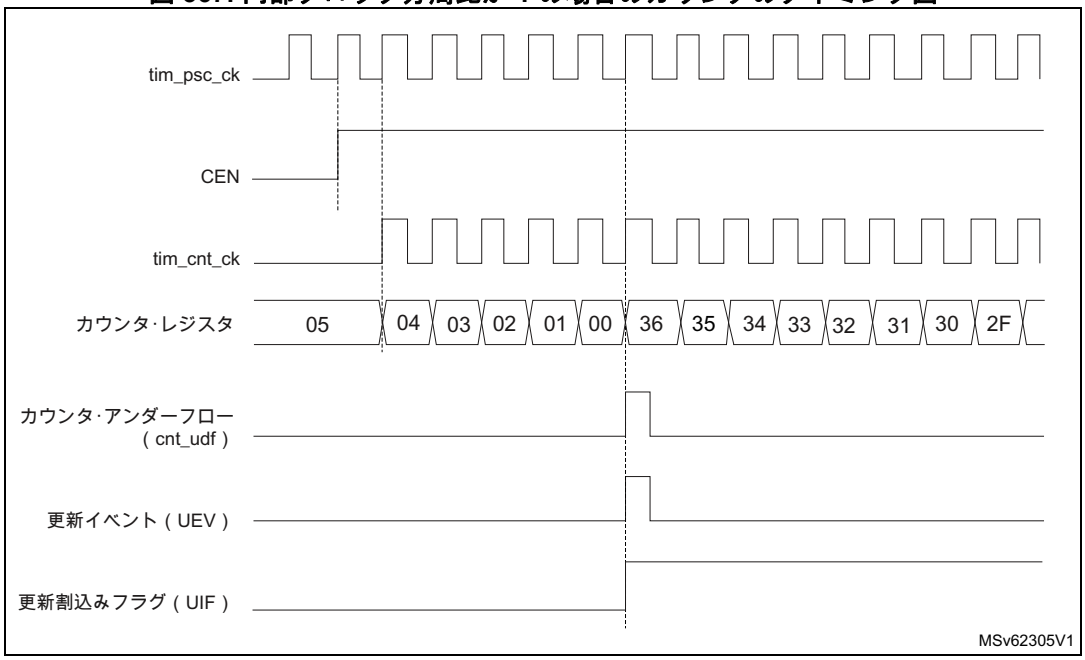


図 368. 内部クロック分周比が 2 の場合のカウンタのタイミング図

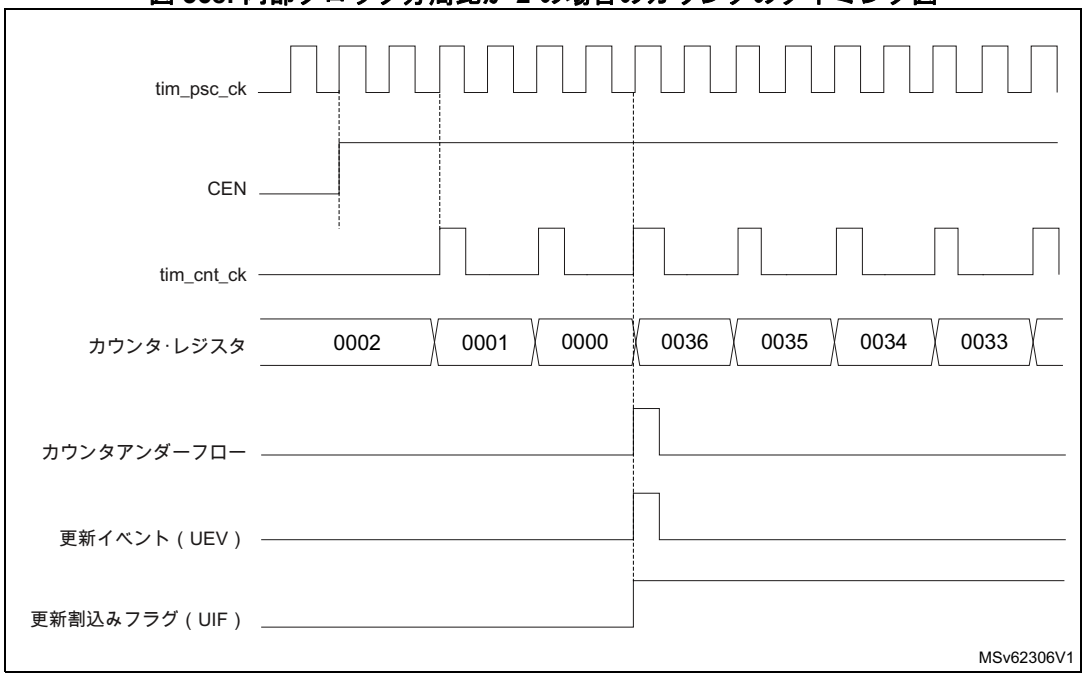


図 369. 内部クロック分周比が 4 の場合のカウンタのタイミング図

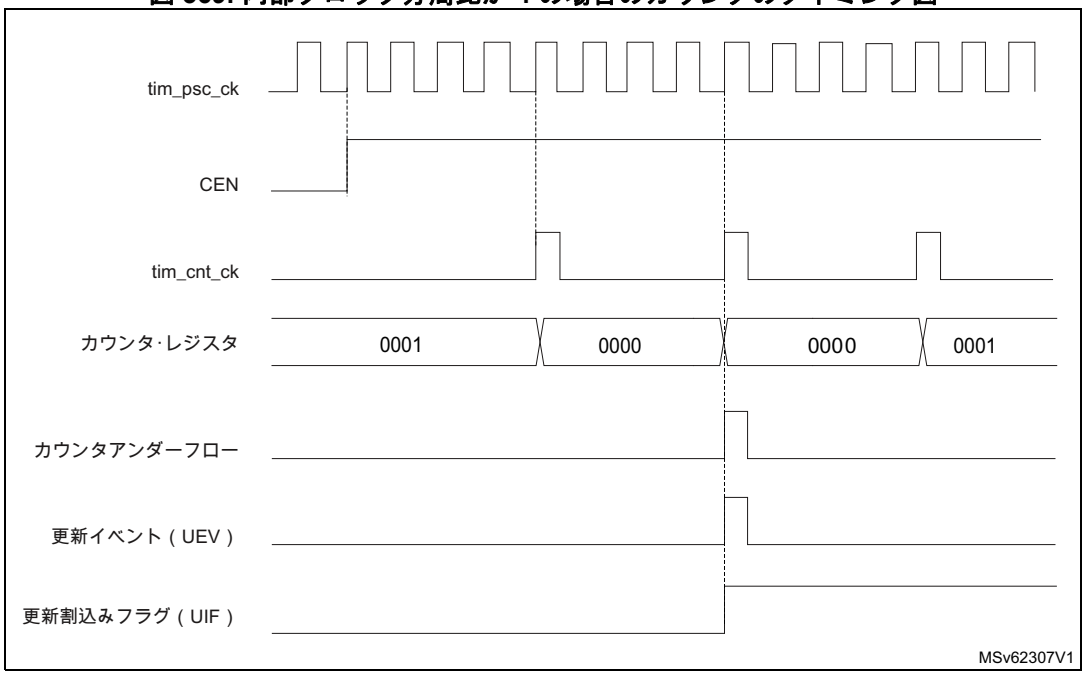


図 370. 内部クロック分周比が N の場合のカウンタのタイミング図

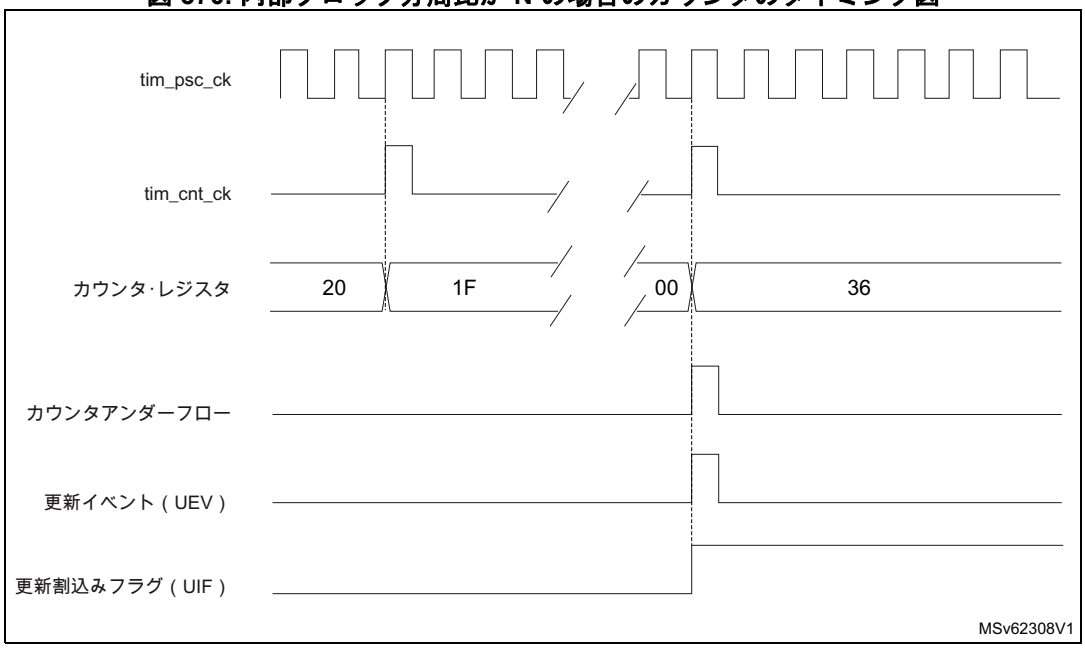
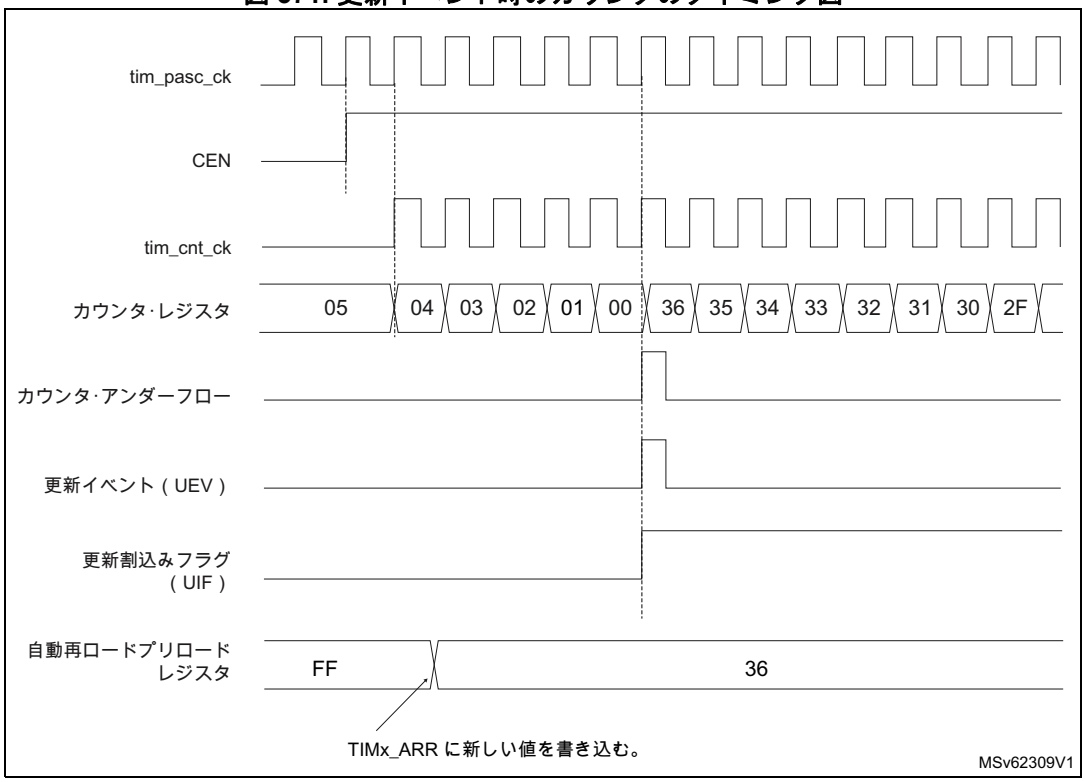


図 371. 更新イベント時のカウンタのタイミング図



センターアライン·モード（アップ／ダウン·カウント）

センターアラインモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）−1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャネルの出力比較割込みフラグは、カウンタがカウントダウンするとき（センターアラインモード 1、CMS=01）、カウンタがカウントアップするとき（センターアラインモード 2、CMS=10）、またはカウンタがカウントアップしてカウントダウンするとき（センターアラインモード 3、CMS=11）にセットされます。

このモードでは、方向ビット（TIMx_CR1 レジスタの DIR）に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、（ソフトウェアで、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（した

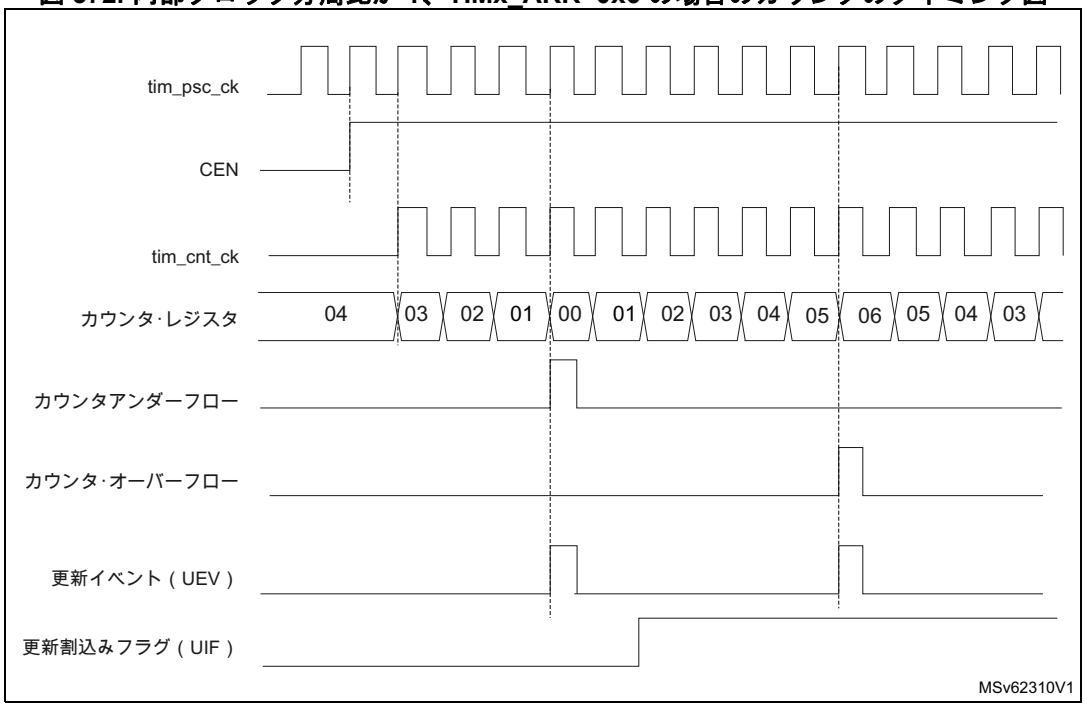
がって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値（TIMx_ARR レジスタの内容）で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります（カウンタに新しい値がロードされます）。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 372. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センターアラインモード 1 が使用されています（詳細については、セクション 29.5.1：1274 ページのTIMx 制御レジスタ 1（TIMx_CR1）（x = 2〜5）を参照）。

図 373. 内部クロック分周比が 2 の場合のカウンタのタイミング図

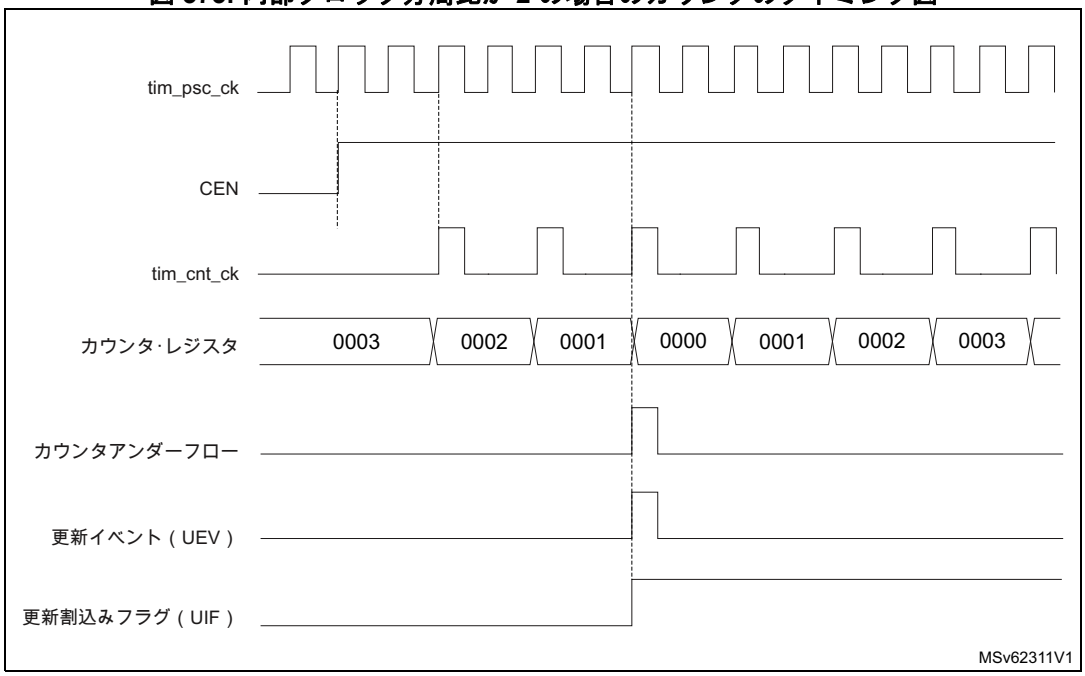
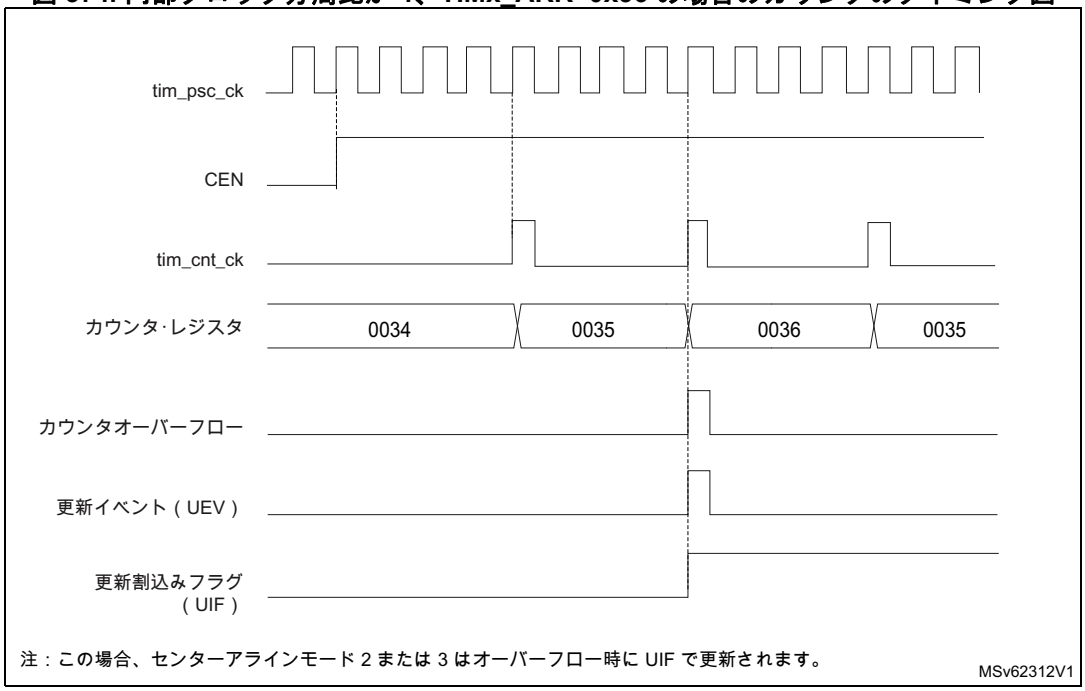


図 374. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 375. 内部クロック分周比が N の場合のカウンタのタイミング図

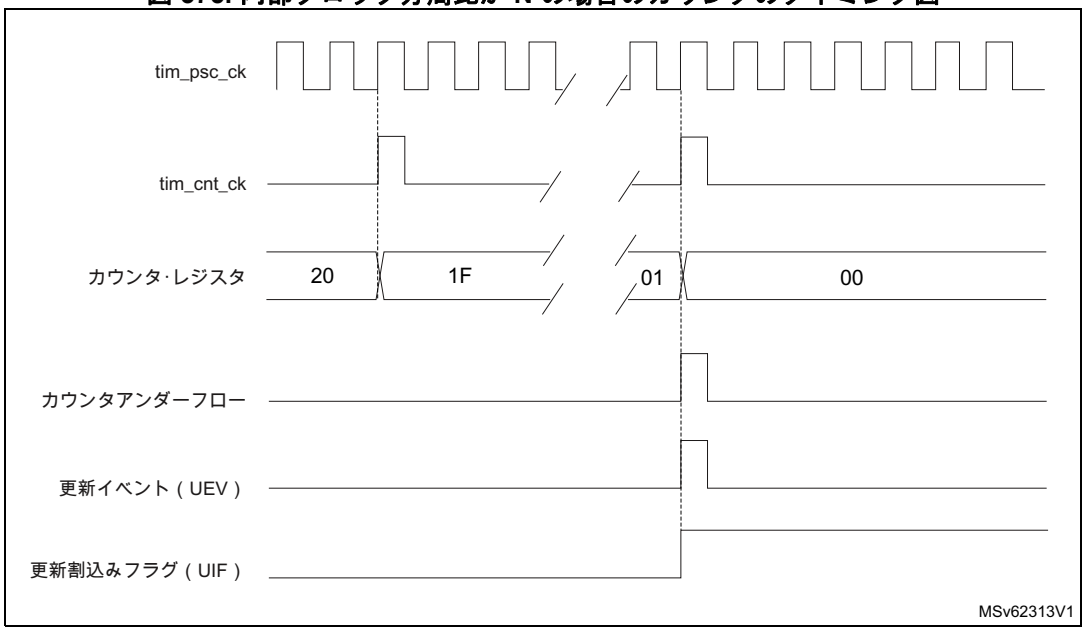


図 376. ARPE=1（カウンタアンダーフロー）の場合の更新イベント時、カウンタタイミング図

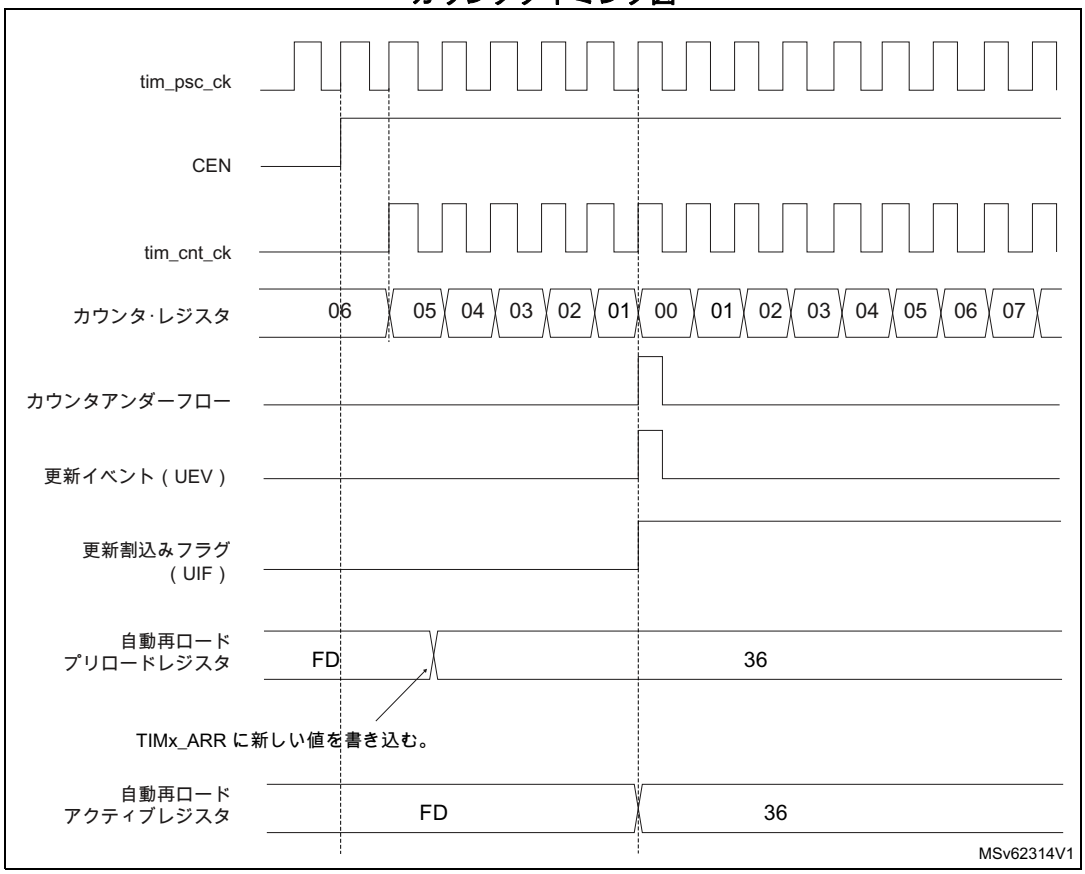
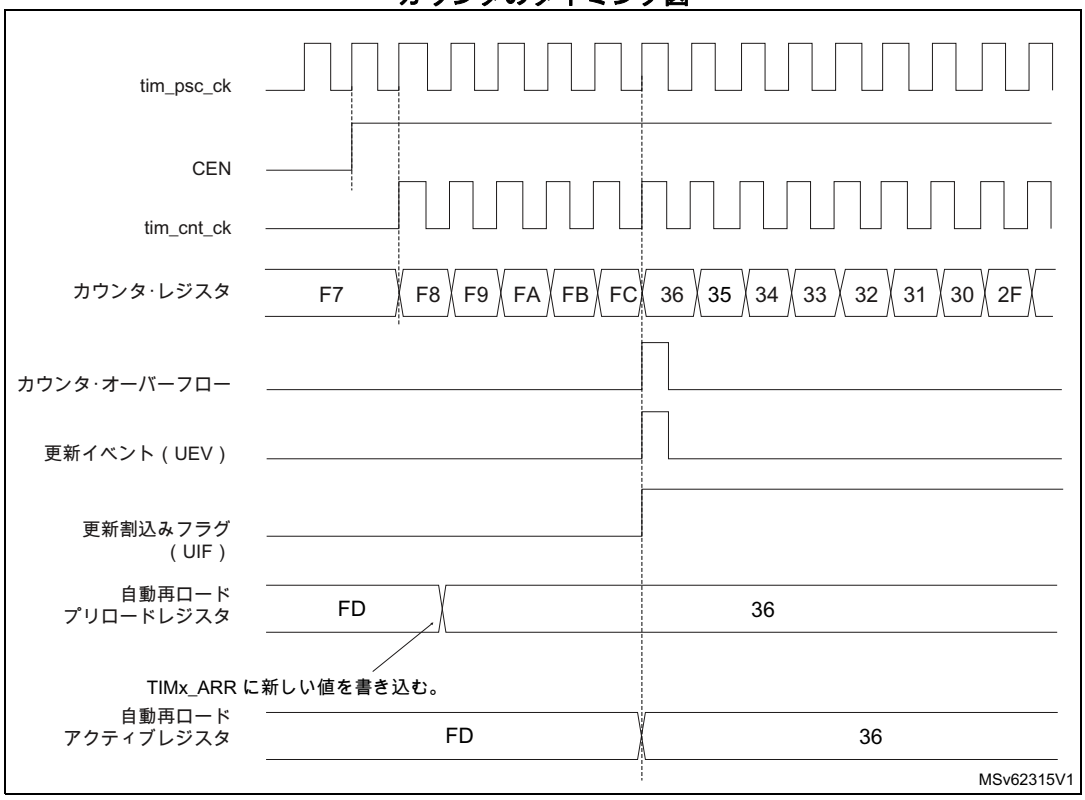


図 377. ARPE=1（カウンタオーバーフロー）の場合の更新イベント時のカウンタのタイミング図



29.4.5 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

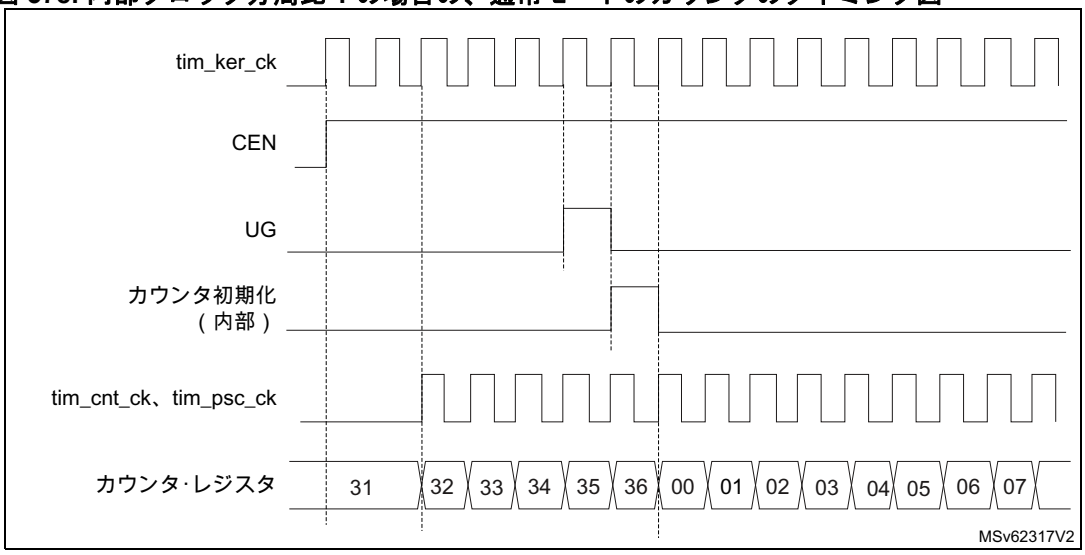
- 内部クロック（tim_ker_ck）
- 外部クロックモード 1：外部入力ピン（tim_ti1 または tim_ti2）
- 外部クロックモード 2：外部トリガ入力（tim_etr_in）
- 内部トリガ入力（tim_itr）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、タイマ 1 がタイマ 2 のプリスケーラとして機能するように設定できます。詳細については、：1267ページの別のタイマのプリスケーラとしたタイマの使用を参照してください。

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（TIMx_SMCR レジスタの SMS=000）、CEN、DIR ビット（TIMx_CR1 レジスタ）と UG ビット（TIMx_EGR レジスタ）が実際の制御ビットであり、ソフトウェアでのみ変更できます（自動的にクリアされたままの UG ビットを除きます）。CEN ビットに 1 が書き込まれるとすぐに、プリスケーラには内部クロック tim_ker_ck が供給されます。

図 378 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

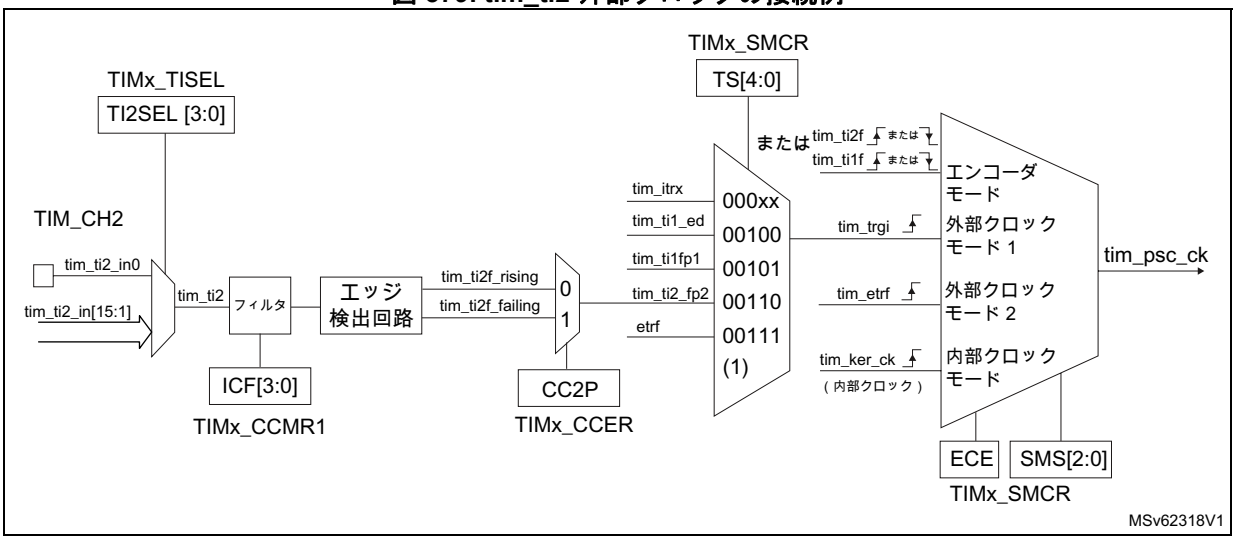
図 378. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 379. tim_ti2 外部クロックの接続例



1. 01000 から 11111 の範囲のコード：tim_itr[15:0]

たとえば、tim_ti2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

たとえば、tim_ti2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な tim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタに CC2S=01 を書き込むことによって、チャネル 2 が tim_ti2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。

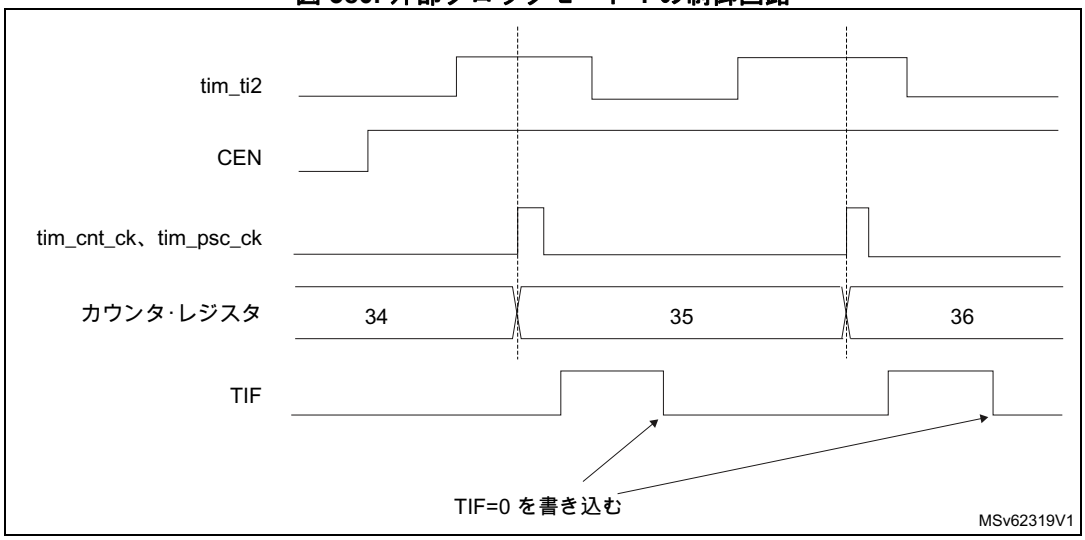
注： **キャプチャプリスケーラはトリガには使用されないので、設定は不要です。**

4. CC2P=0、CC2NP=0、および CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして tim_ti2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

tim_ti2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際のカウンタクロックまでの間には、tim_ti2 入力の再同期回路による遅延があります。

図 380. 外部クロックモード 1 の制御回路



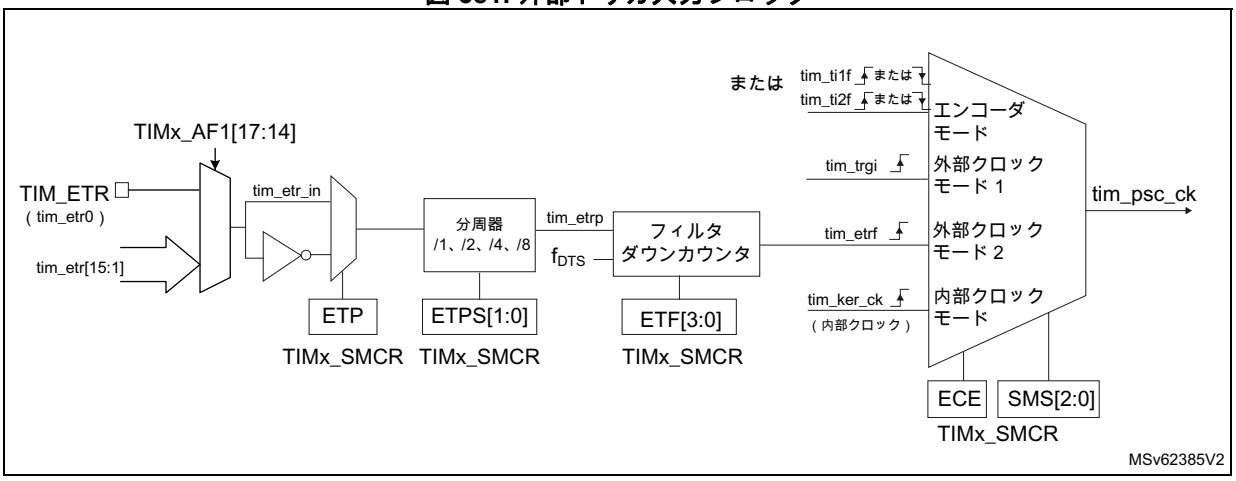
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 tim_etr_in の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 381 に、外部トリガ入力ブロックの概要を示します。

図 381. 外部トリガ入力ブロック



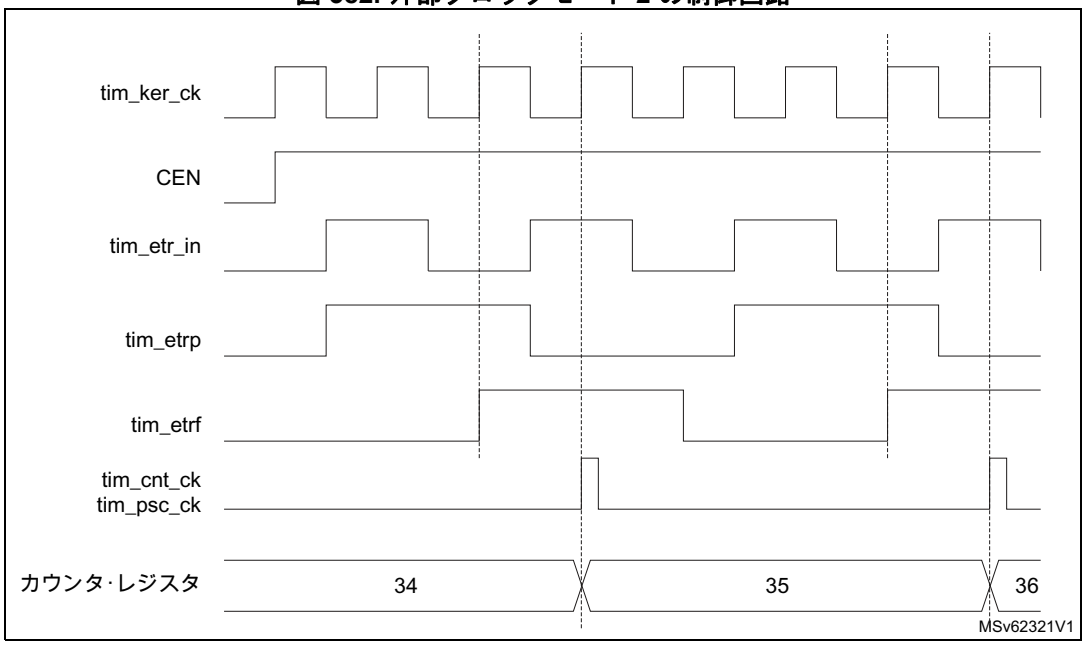
たとえば、tim_etr_in の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. TIMx_AF1 レジスタの ETRSEL[3:0] ビットで、適切な tim_etr_in ソース（内部または外部）を選択します。
2. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
3. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケーラを設定します。
4. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、tim_etr_in の立ち上がりエッジ検出を選択します。
5. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の tim_etr_in 立ち上がりエッジごとに 1 回カウントします。

tim_etr_in の立ち上がりエッジから実際のカウンタクロックまでの間に、tim_etrp 信号の再同期回路による遅延があります。結果として、カウンタで正しくキャプチャできる最大周波数は、最大で TIMxCLK 周波数の 1/4 です。ETRP 信号が高速の場合、ユーザは適切な ETPS プリスケーラー設定によって外部信号の分周比を適用する必要があります。

図 382. 外部クロックモード 2 の制御回路



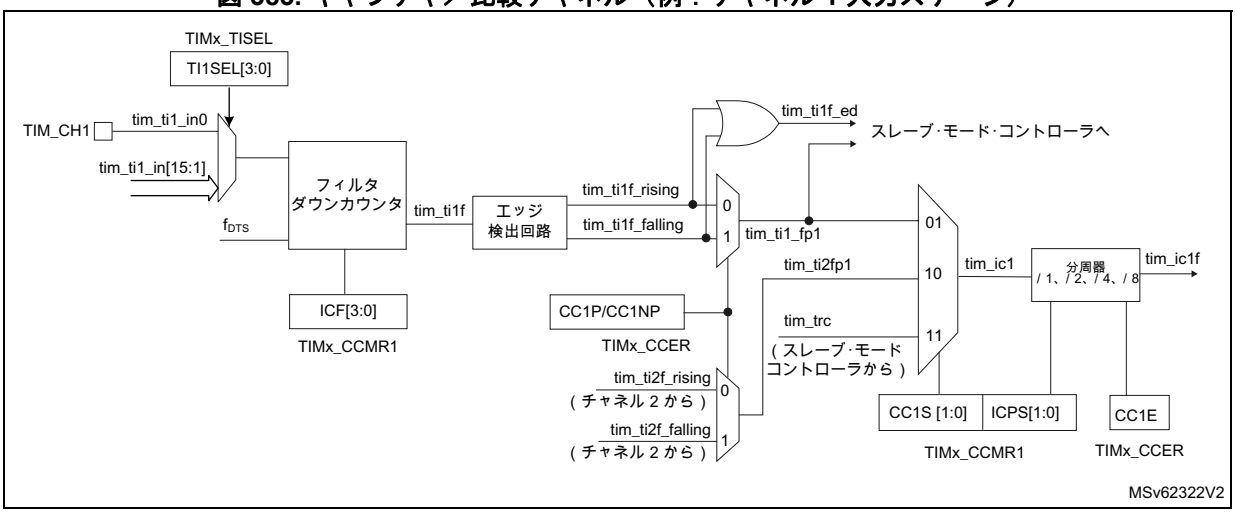
29.4.6 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する tim_tix 入力をサンプリングして、フィルタリングを行った信号 tim_tixf を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 383. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、その後、基準として使用される中間波形 tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図 384. キャプチャ／比較チャネル 1 メイン回路

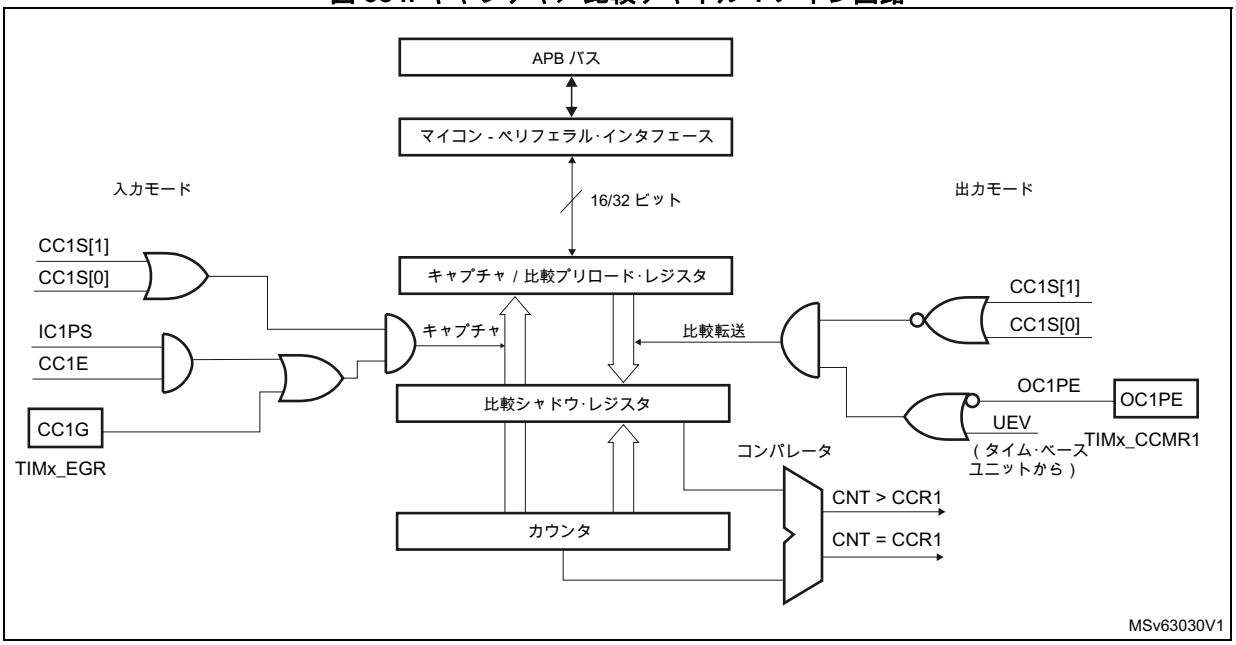
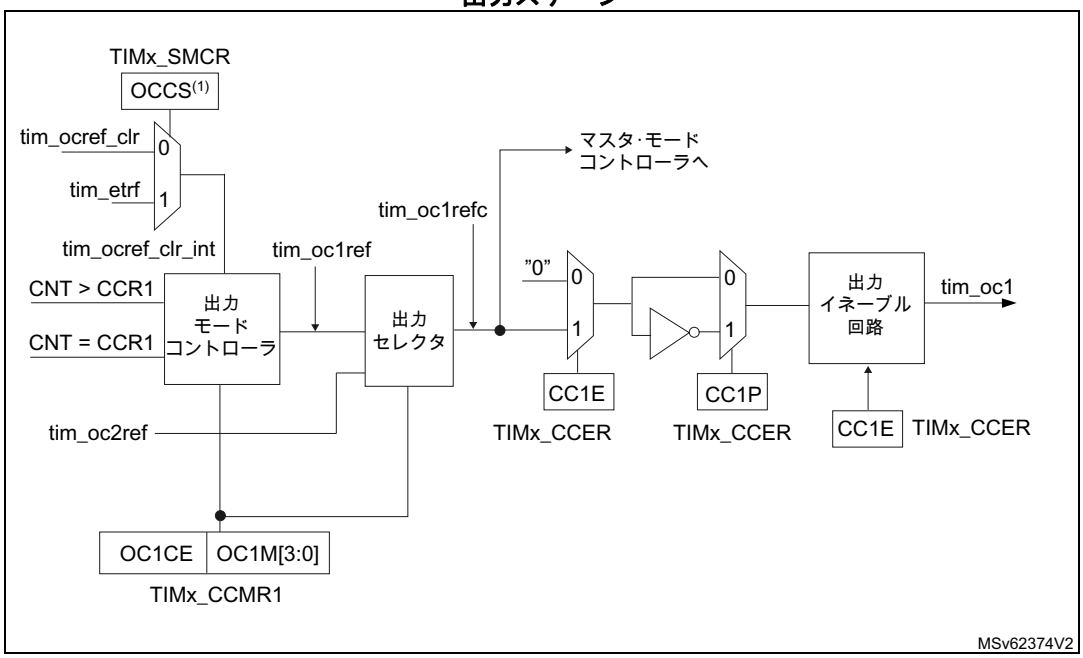


図 385. キャプチャ／比較チャネル（チャネル 1、同じくチャネル 2、3、および 4）の出力ステージ



1. いくつかのインスタンスのみで使用できます。使用できない場合は、tim_etrf は直接 tim_ocref_clr_int に接続されます。

キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

29.4.7 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、tim_ti1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な tim_tix_in[15:0] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は tim_ti1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続される信号に関して、必要な入力フィルタ時間をプログラムします（入力が tim_tix の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、tim_ti1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. tim_ti1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P、CC1NP、および CC1NP ビットに“000”を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに 00 を書き込みます）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

29.4.8 PWM 入力モード

このモードでは、単一の tim_tix 入力に接続された PWM 信号の周期とデューティサイクルの両方を以下のように測定できます。

- TIMx_CCR1 レジスタは周期値（2 つの連続した立ち上がりエッジ間の間隔）を保持します。
- TIM_CCR2 レジスタはパルス幅（2 つの連続した立ち上がりと立ち下がりエッジ間の間隔）を保持します。

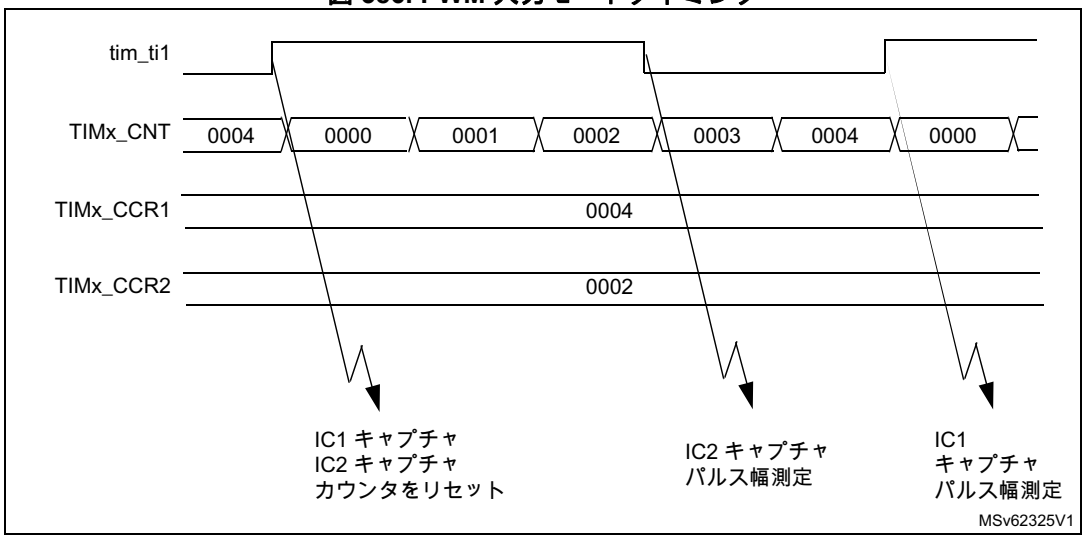
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2 つの ICx 信号が同じ tim_tix 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

tim_ti1 に適用される PWM 信号の周期とパルス幅は、以下の手順を使用して測定できます。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な tim_tix_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、tim_ti1fp1 のアクティブ極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用します）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2P ビットに“1”を、CC2NP ビットに“0”を書き込むことによって（立ち下がりエッジでアクティブ）、tim_ti1fp2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（tim_ti1fp1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに 1 を書き込むことによって、キャプチャを有効にします。

図 386. PWM 入力モードタイミング



1. tim_ti1fp1 と tim_ti2fp2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2信号でのみ使用できます。

29.4.9 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS ビット = 00）では、各出力比較信号（tim_ocxref、そして tim_ocx）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 101 を書き込む必要があります。これにより、tim_ocxref は強制的にハイレベルになり（tim_ocxref は常にアクティブハイ）、tim_ocx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（tim_ocx アクティブハイ）=> tim_ocx は強制的にハイレベルになります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、tim_ocxref 信号を強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、出力比較モードのセクションで説明します。

29.4.10 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。

- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

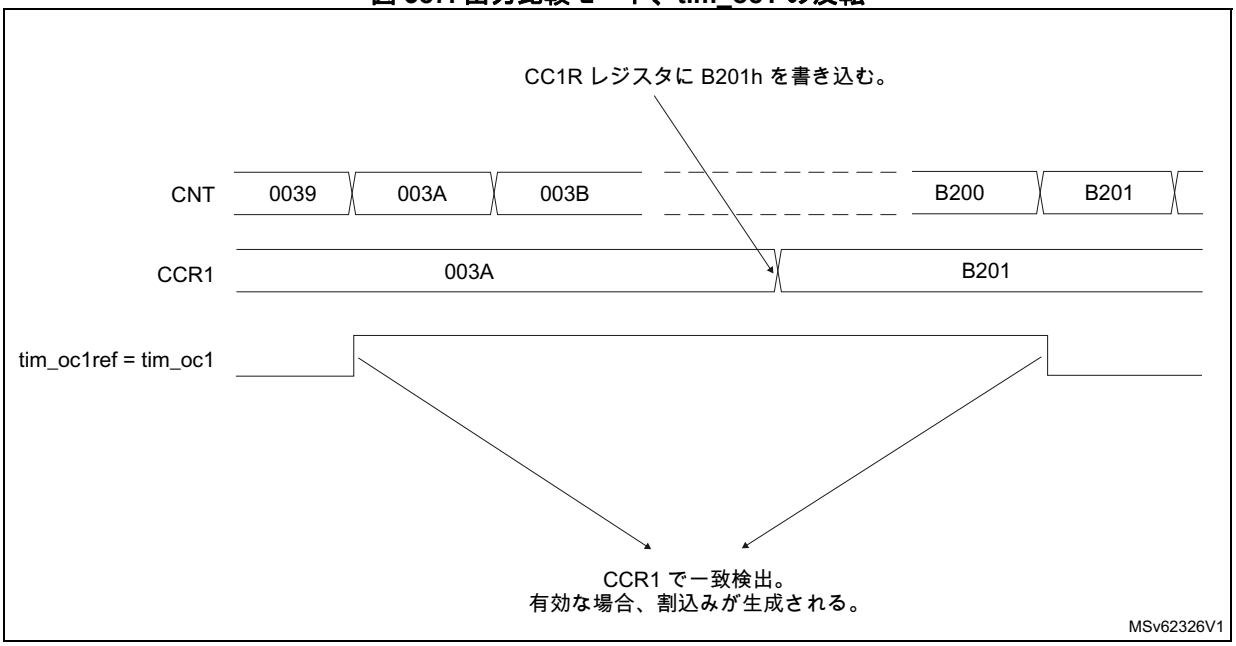
出力比較モードでは、更新イベント UEV は tim_ocxref および tim_ocx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエスト／DMA リクエストを生成する場合は、CCxIE ビット／CCxDE ビットをセットします。
4. 出力モードを選択します。例：
   a) CNT と CCRx が一致したときに tim_ocx 出力ピンを反転するには、OCxM ビットに 0011 を書き込みます。
   b) プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   c) アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   d) 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアで TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を図 387 に示します。

図 387. 出力比較モード、tim_oc1 の反転



29.4.11 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、個々のチャネル（tim_ocx 出力ごとに PWM 1 波形）で、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）や“111”（PWM モード 2）を書き込むことで、独自に選択できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

tim_ocx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。これは、アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。tim_ocref_clr は、tim_etr_in または tim_oceref_clr 信号を介した外部イベントによってクリアできます。この場合、tim_ocref_clr 信号は、以下の場合にのみ、アサートされます。

- 比較一致イベントの後。
- 出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）が停止構成（比較なし、OCxM=000）から PWM モードの 1 つ（OCxM=110 または 111）へ切り替えられたとき。タイマの動作中は、ソフトウェアで強制的に PWM になります。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

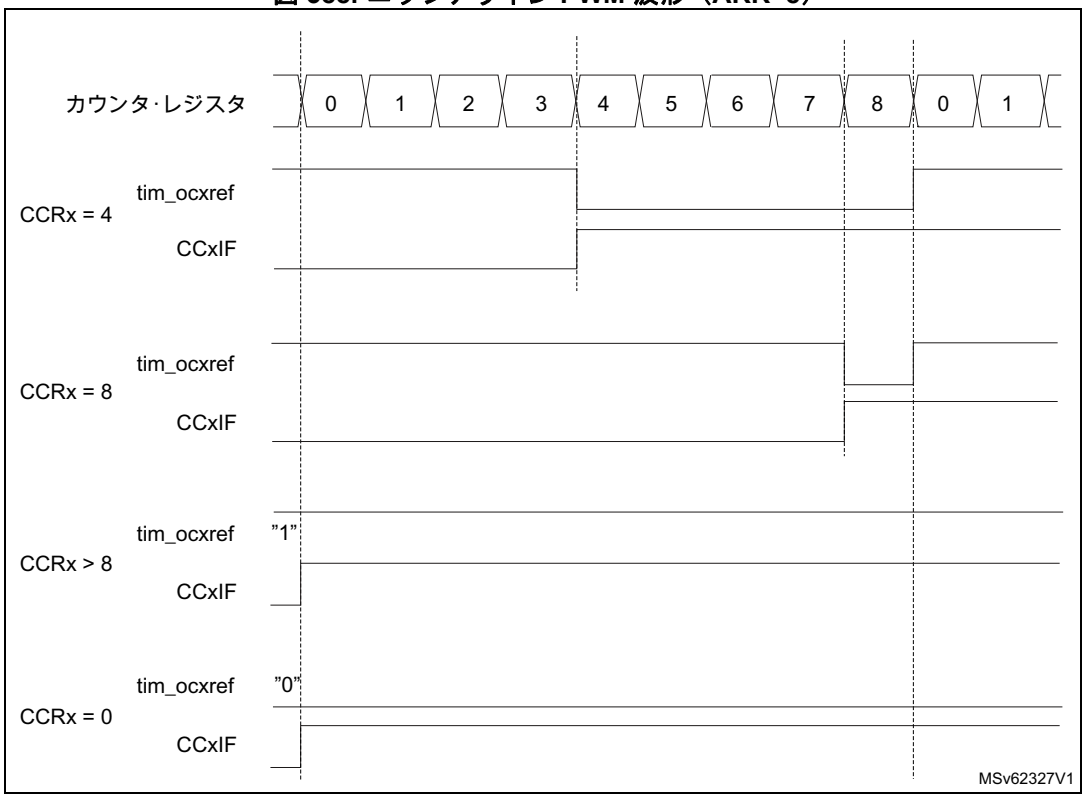
PWM エッジアラインモード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。1207 ページのアップカウントモードを参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 tim_ocxref は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、tim_ocxref は“1”に保持されます。比較値が 0 の場合、tim_ocxref は“0”に保持されます。図 388 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 388. エッジアライン PWM 波形（ARR=8）



ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。1210 ページのダウンカウントモードを参照してください。

PWM モード 1 では、基準信号 tim_ocxref は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、tim_ocxref は 100% に保持されます。このモードでは、PWM 信号を生成することはできません。

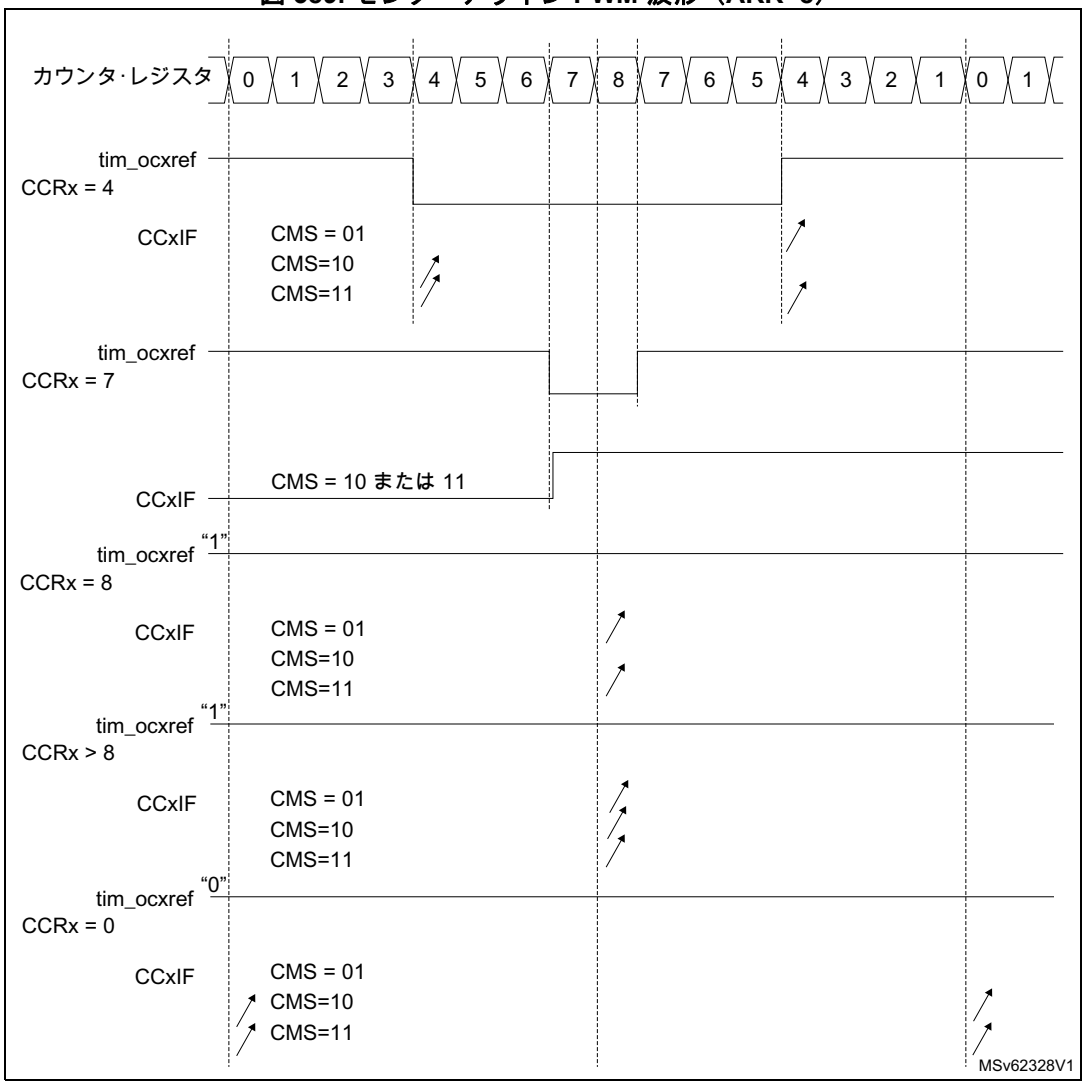
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての設定は、tim_ocxref/tim_ocx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット （DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。1213 ページのセンターアライン·モード（アップ／ダウン·カウント）を参照してください。

図 389 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 389. センターアライン PWM 波形（ARR=8）



センターアラインモードの使用に関するヒント：

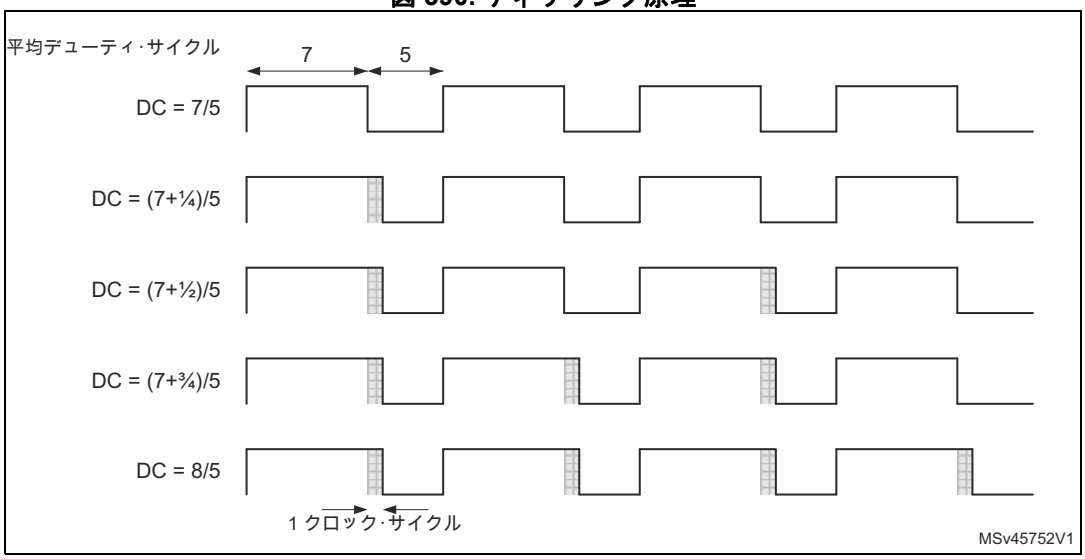
- センターアラインモードを開始するときには、現在のアップ／ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書込みは、予期しない結果を招くことがあるので推奨されません。特に、
  - 自動再ロード値より大きい値がカウンタに書き込まれた場合（TIMx_CNT > TIMx_ARR）、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
  - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して（TIMx_EGR レジスタの UG ビットをセットする）、動作中はカウンタへの書込みを行わないことです。

ディザリングモード

PWM モードの有効分解能は、TIMx_CR1 レジスタの DITHEN ビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）と ARR（PWM 周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16 個の連続した PWM 周期にわたって実際の CCR（または ARR）値をわずかに変更する（1 タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたは PWM 周期を考えると、16 倍の分解能の向上が可能になります。下記の図 390 は、4 つの連続する PWM サイクルに適用されたディザリング原理を示します。

図 390. ディザリング原理

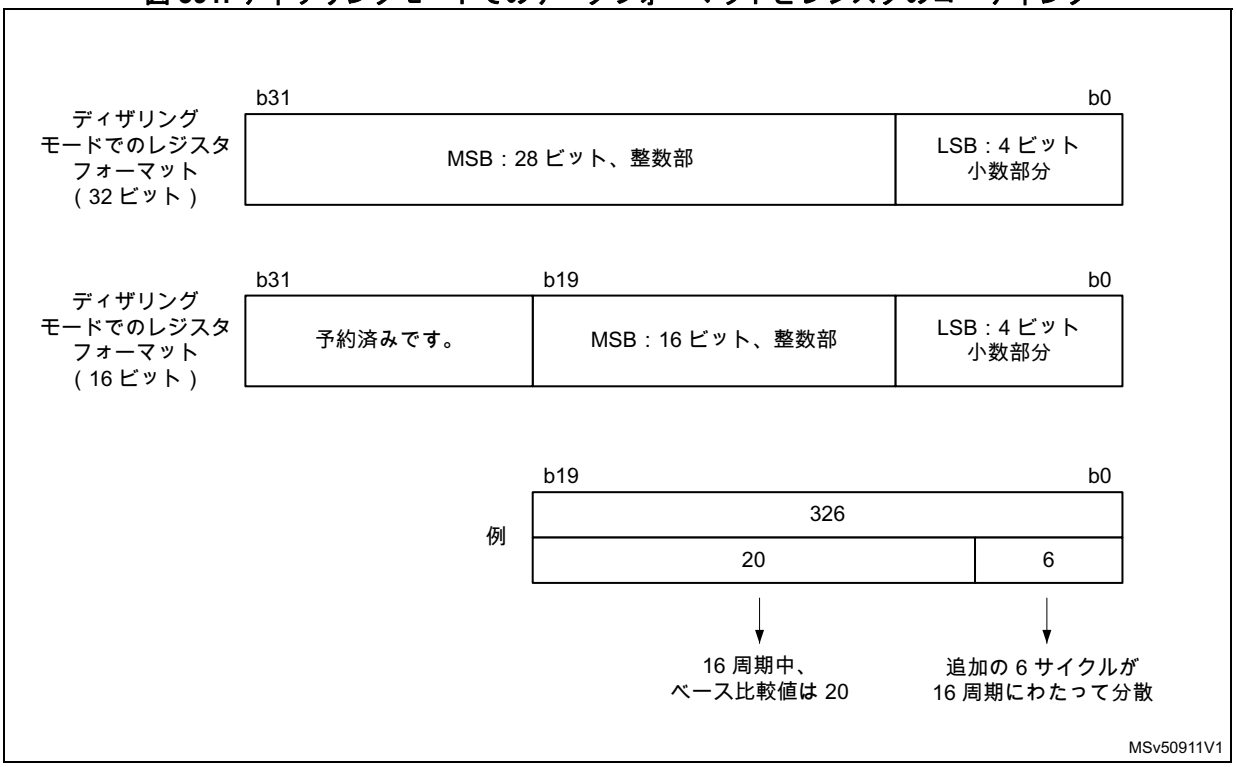


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図 391 を参照）。

- 下位 4 ビットは、拡張分解能部分（分数部分）用です。
- 上位ビットは、4 桁だけ左にシフトされ、ベース値用のコーディングになります。16 ビットモードでは、16 ビットのフォーマットが維持されます。

注： **DITHEN ビットがセット／リセットされると、ARR と CCR の値が自動的に更新されます（たとえば、DITHEN = 0 で ARR = 0x05 の場合、DITHEN = 1 で ARR = 0x50 に更新されます）。**
DITHEN ビットをリセットするときは、次の手順に従う必要があります。
1.CEN および ARPE ビットをリセットする必要があります。
2.ARR[3:0] ビットをリセットする必要があります。
3.DITHEN ビットをリセットする必要があります。
4.CCIF フラグをクリアする必要があります。
5.CEN ビットをセットすることができます（最終的に ARPE = 1 とともに）。

図 391. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

$$\text{ディザリングモード無効時：} F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモード (16 ビットタイマ) ：} F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

$$\text{ディザリングモード (32 ビットタイマ) ：} F_{pwmMin} = \frac{F_{Tim}}{268435454 + \frac{15}{16}}$$

注： **16 ビットタイマの場合、TIMx_ARR および TIMxCCRy の最大値は、ディザリングモードでは 0xFFFEF に制限されています（整数部では 65534、ディザリング部分では 15 に対応します）。**
32 ビットタイマの場合、TIMx_ARR および TIMxCCRy の最大値は、ディザリングモードでは 0xFFFFFFEF に制限されています（整数部では 264435454、ディザリング部分では 15 に対応します）。

下記の図 392 と図 393 に示すように、ディザリングモードによって PWM 分解能を増加することができます。

図 392. PWM 分解能 対 周波数（16 ビットモード）

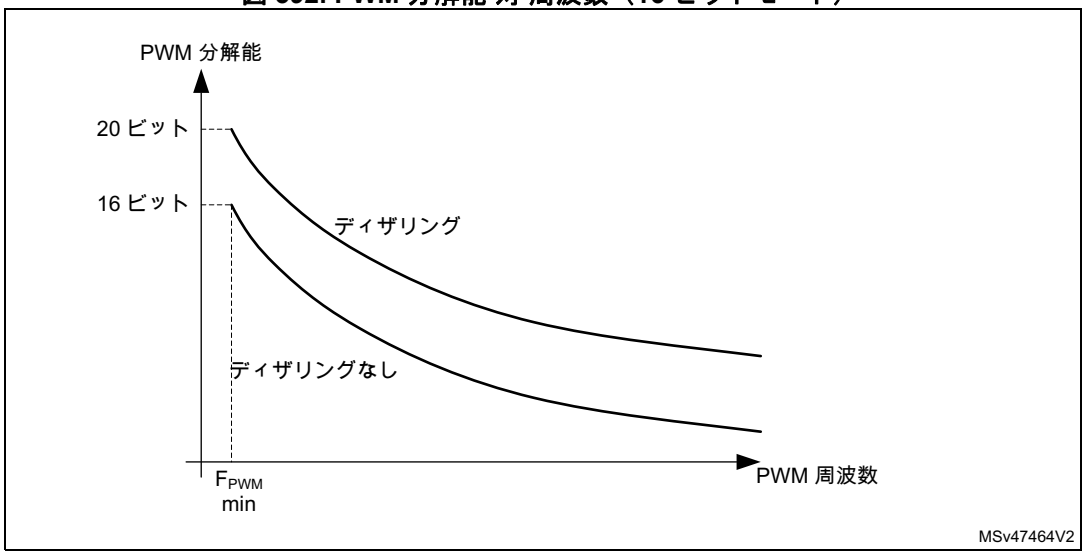
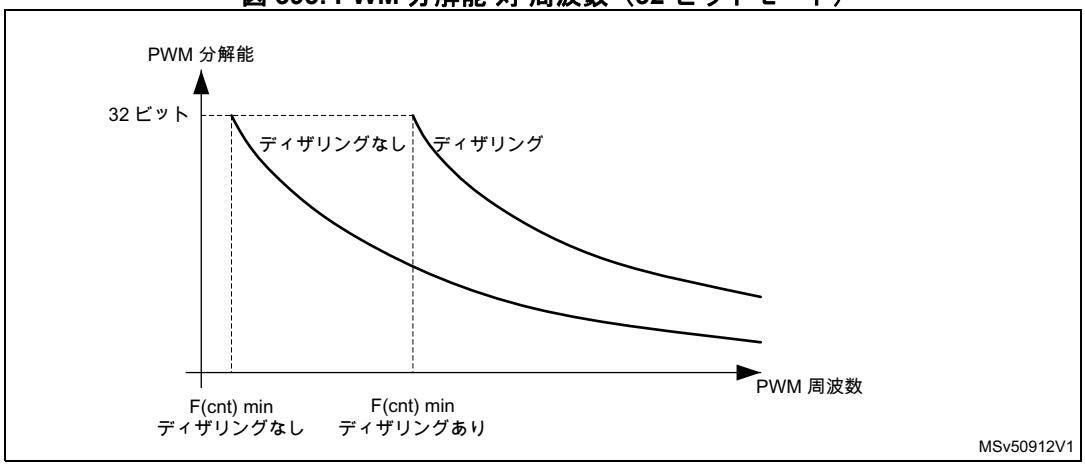
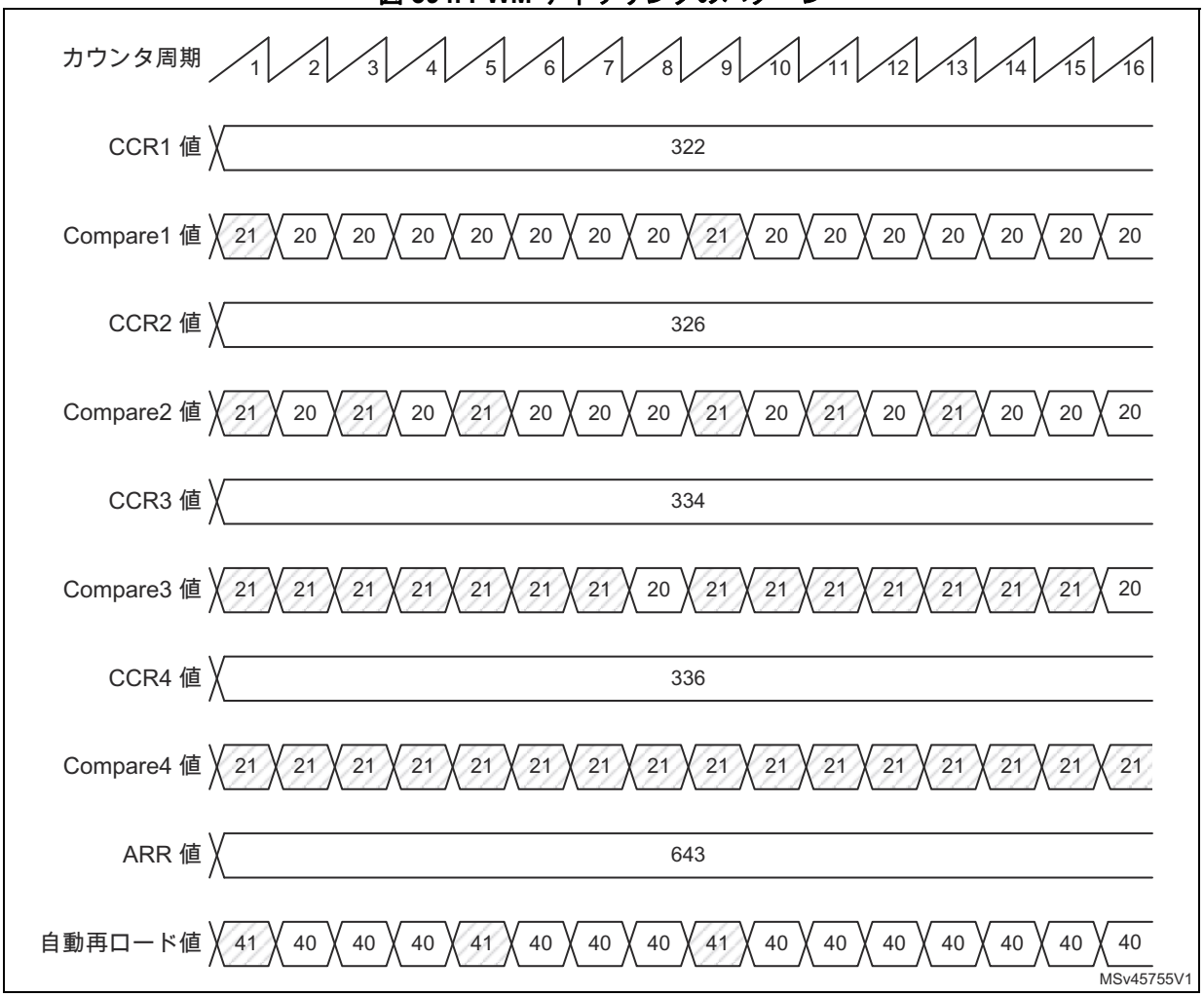


図 393. PWM 分解能 対 周波数（32 ビットモード）



下記の図 394 で説明するように、デューティサイクルや周期の変更は、連続した 16 周期にわたって広がります。

図 394. PWM ディザリングのパターン



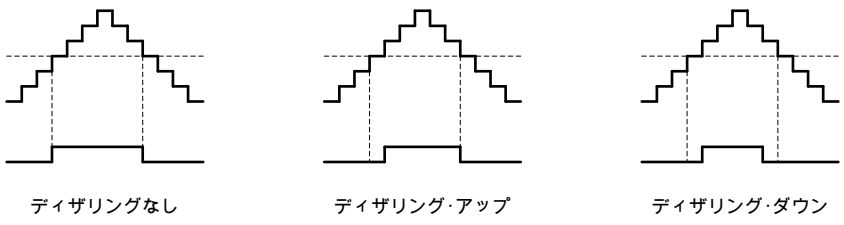
自動再ロード値と比較値の増分は、下記の表 278 に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表 278. CCR および ARR レジスタの変更のディザリングパターン

LSB 値	PWM 周期															
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

ディザリングモードはセンターアライン PWM モード（TIMx_CR1 レジスタの CMS ビットが“00”に等しくないとき）でも使用できます。この場合、ディザリングパターンは、以下の図 395 に示すように、アップおよびダウンカウントフェーズを考慮して、8 つの連続した PWM 周期にわたって適用されます。

図 395. センターアライン PWM モードでのデューティサイクルへのディザリングの影響



下記の表 279 は､センターアライン PWM モードでどのようにディザリングパターンが追加されるかを示しています。

表 279. センターアライン PWM モードでの CCR レジスタ変更のディザリングパターン

LSB 値	PWM 周期															
	1		2		3		4		5		6		7		8	
	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn	アップ	Dn
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

29.4.12 非対称 PWM モード

非対称モードでは、プログラム可能な位相シフトによって 2 つのセンターアライン PWM 信号の生成を可能にします。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや位相シフトは TIMx_CCRx レジスタペアで決定されます。1 つ目のレジスタがアップカウント時の PWM を制御し、2 つ目のレジスタがダウンカウント時の PWM を制御することで、PWM は PWM ハーフサイクルごとに調整されます。

- tim_oc1refc（または tim_oc2refc）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。
- tim_oc3refc（または tim_oc4refc）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

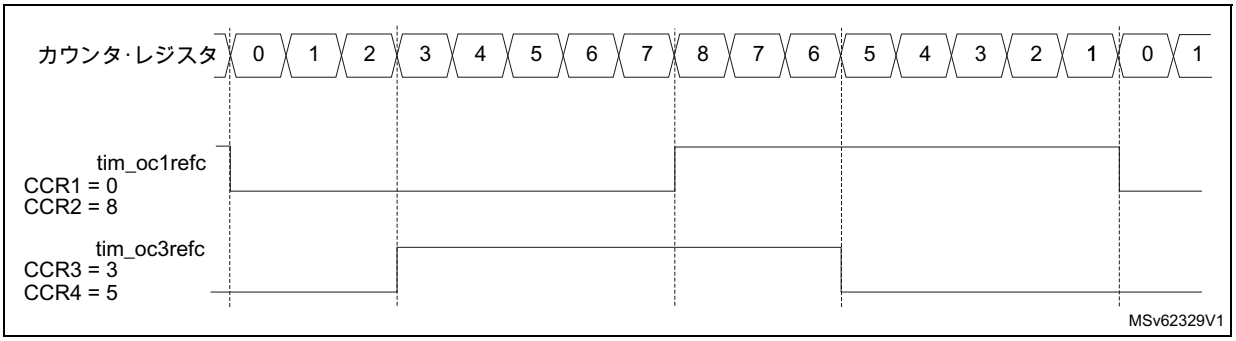
非対称 PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1110”（非対称 PWM モード 1）または“1111”（非対称 PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの tim_ocx 出力）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

特定のチャネルが非対称の PWM チャネルとして使用されると、その 2 次チャネルも使用できます。たとえば、tim_oc1refc 信号がチャネル 1（非対称 PWM モード 1）に生成されると、チャネル 2 の tim_oc2ref 信号、または非対称 PWM モード 2 の結果として得られる tim_oc2refc 信号を出力できます。

図 396 は、非対称 PWM モードを使用して生成される信号の例を表します（チャネル 1 から 4 は非対称 PWM モード 2 として設定されます）。

図 396. 50% デューティサイクルの 2 位相シフトされた PWM 信号の生成



29.4.13 組み合わせ PWM モード

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 tim_ocxrefc は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

– tim_oc1refc（または tim_oc2refc）は、TIMx_CCR1 および TIMx_CCR2 によって制御されます。

– tim_oc3refc（または tim_oc4refc）は、TIMx_CCR3 および TIMx_CCR4 によって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの tim_ocx 出力）。

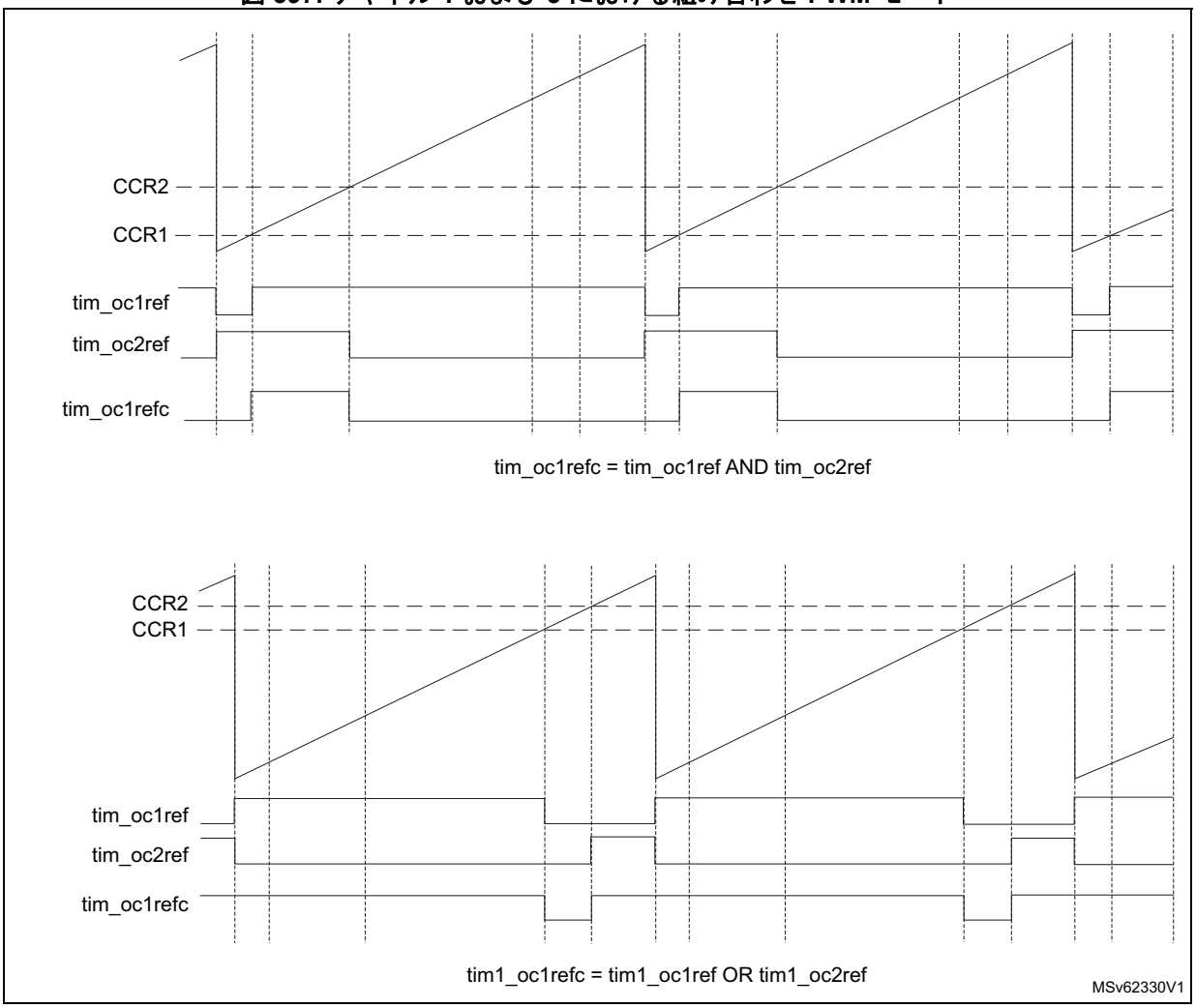
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、2 次チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 397 は、次の設定で取得可能な組み合わせ PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合
- チャネル 3 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 4 が PWM モード 1 で設定されている場合

図 397. チャネル 1 および 3 における組み合わせ PWM モード



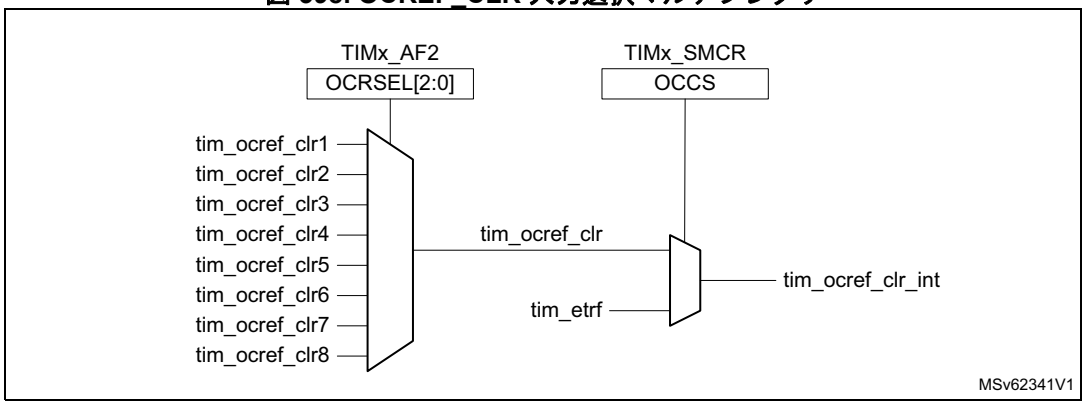
29.4.14 外部イベントによる tim_ocxref 信号のクリア

特定のチャネルの tim_ocxref 信号は tim_ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。tim_ocxref は、次の更新イベント（UEV）が発生するまでローのままになります。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

tim_ocref_clr_int のソースは、OCREF クリア選択機能の実装に依存します。セクション 29.3 : TIM2/TIM3/TIM4/TIM5 の実装を参照してください。

OCREF クリア選択機能が実装されている場合は、tim_ocref_clr_int は、TIMx_SMCR レジスタで OCCS ビットを設定することで、tim_ocref_clr 入力と tim_etrf 入力（フィルタ後の tim_etr_in）の間で選択できます。tim_ocref_clr 入力は、以下の図 398 に示すように、TIMx_AF2 レジスタの OCRSEL[2:0] ビットフィールドを使用して複数の tim_ocref_clr[7..0] 入力から選択できます。

図 398. OCREF_CLR 入力選択マルチプレクサ



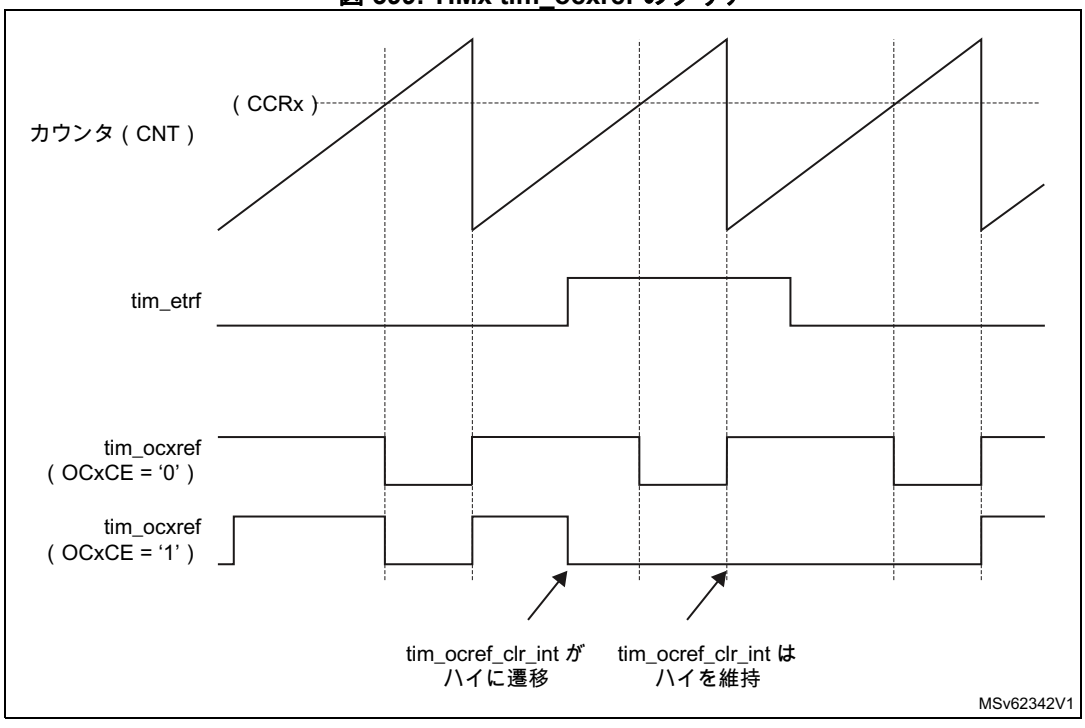
OCREF クリア選択機能が実装されていない場合は、tim_ocref_clr_int 入力は直接 tim_etrf 入力に接続されます。

たとえば、tim_ocref_clr_int 信号をコンパレータの出力に接続して、電流処理に使用することができます。この場合、tim_etr_in は次のように設定する必要があります。

1. 外部トリガプリスケーラをオフに保つ必要があります。すなわち、TIMx_SMCR レジスタのビット ETPS[1:0] が 00 にクリアされます。
2. 外部クロックモード 2 を無効にする必要があります。すなわち、TIM1_SMCR レジスタのビット ECE が 0 にクリアされます。
3. 外部トリガ極性（ETP）と外部トリガフィルタ（ETF）は、アプリケーションのニーズに応じて設定できます。

図 399 に、OCxCE イネーブルビットの両方の値について、tim_etrf 入力がハイになったときの tim_ocxref 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 399. TIMx tim_ocxref のクリア



注： 100% デューティサイクルの PWM の場合 （CCRx > ARR の場合）、次のカウンタオーバーフローで tim_ocxref が再度有効になります。

29.4.15 ワンパルスモード

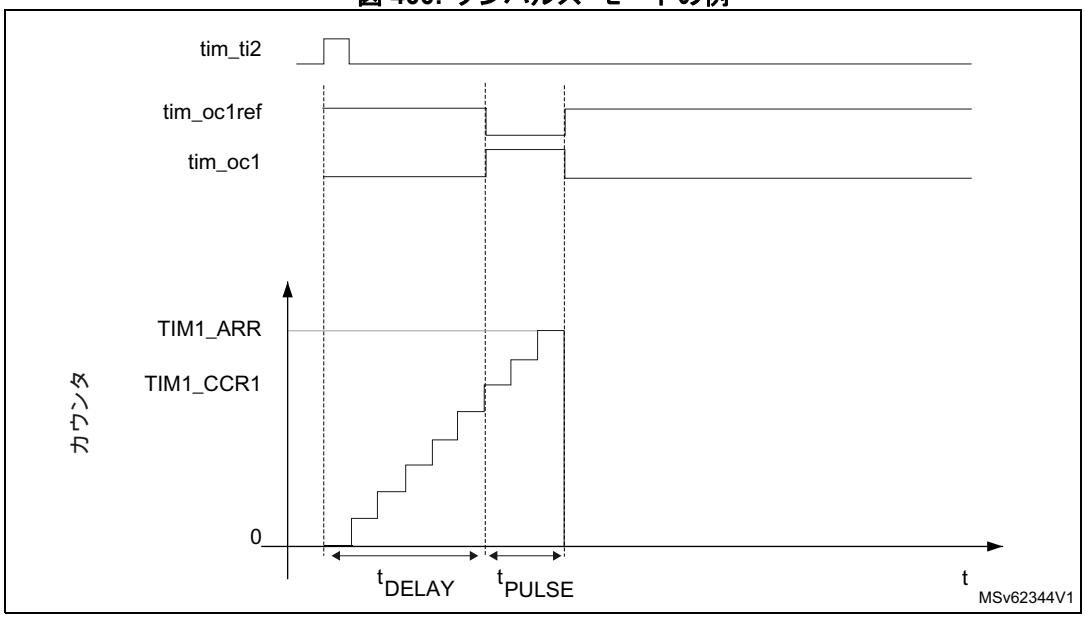
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 400. ワンパルス･モードの例



たとえば、tim_ti2 入力ピンで立ち上がりエッジが検出されたときに、tim_oc1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

tim_ti2fp2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な tim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタに CC2S = 01 を書き込むことによって、tim_ti2fp2 を tim_ti2 にマッピングします。
3. tim_ti2fp2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタの CC2P ビットと CC2NP ビットに“0” を書き込みます。
4. tim_ti2fp2 をスレーブモードコントローラのトリガ（tim_trgi）として構成するために、TIMx_SMCR レジスタの TS=“00110”を書き込みます。
5. tim_ti2fp2 を使用してカウンタを開始するために、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE=1 を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、tim_ti2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので（シングルモード）、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊ケース：tim_ocx 高速イネーブル：

ワンパルスモードでは、tim_tix 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、tim_ocxref（および tim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

29.4.16 再トリガ可能なワンパルス･モード

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 29.4.15 : で説明する再トリガ不可能なワンパルス･モードと、次のような違いがあります。

- トリガが発生するとパルスが直ちに始まります（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] = 「1000」（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が「1000」または「1001」にセットされています。

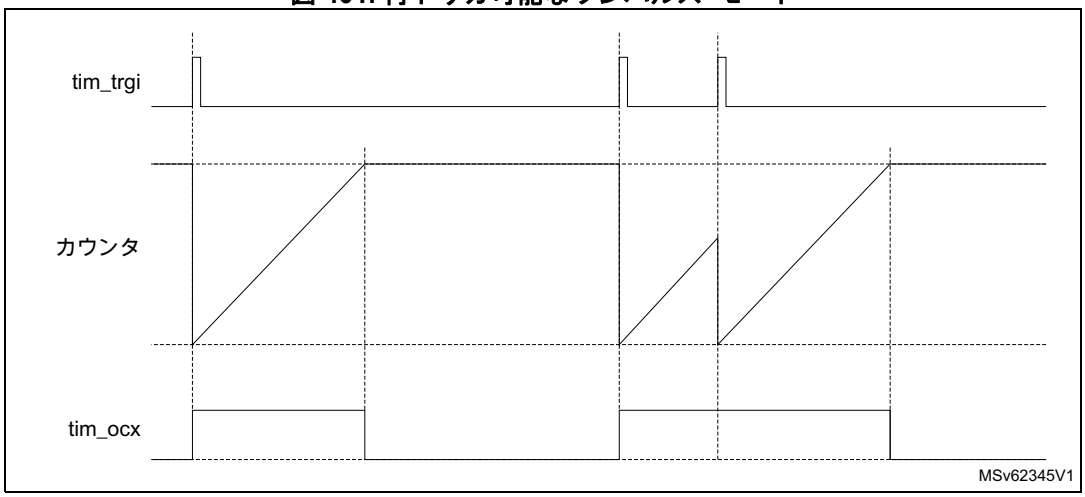
タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **再トリガ可能なワンパルス･モードでは、CCxIF フラグは意味を持ちません。**

OCxM[3:0] および SMS[3:0] ビットフィールドは互換性を確保するために 2 つの部分に分割され、最上位ビットと下位 3 ビットは連続していません。

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 401. 再トリガ可能なワンパルス･モード

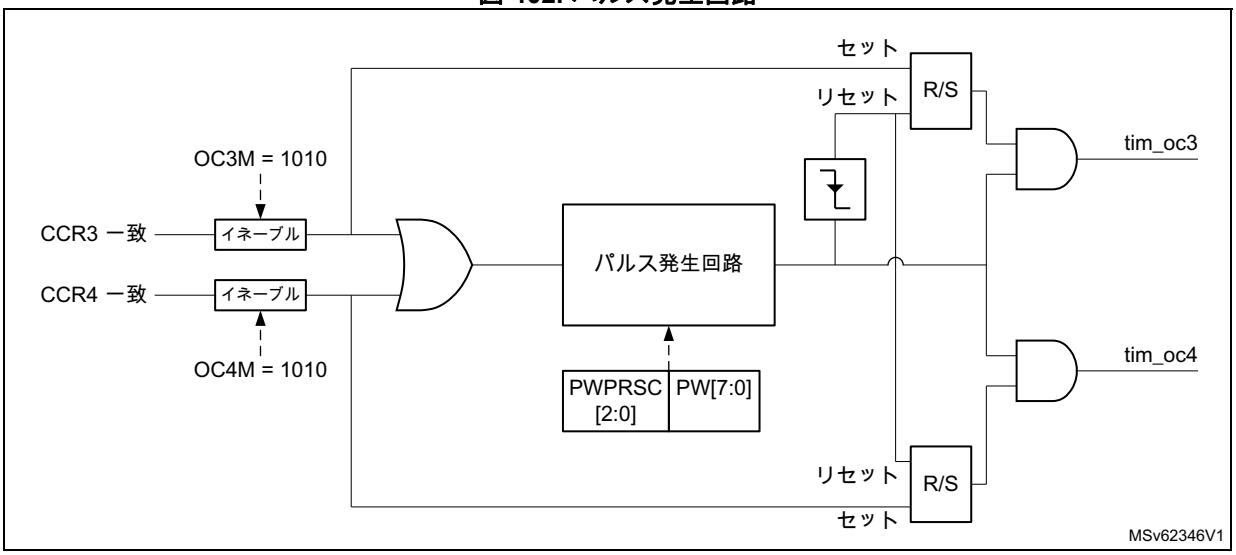


29.4.17 比較モードでのパルス

比較一致イベント時にパルスを生成できます。デバッグまたは同期のために、カウンタ値が特定の比較値と等しいときに、プログラム可能なパルス幅を持つ信号が生成されます。

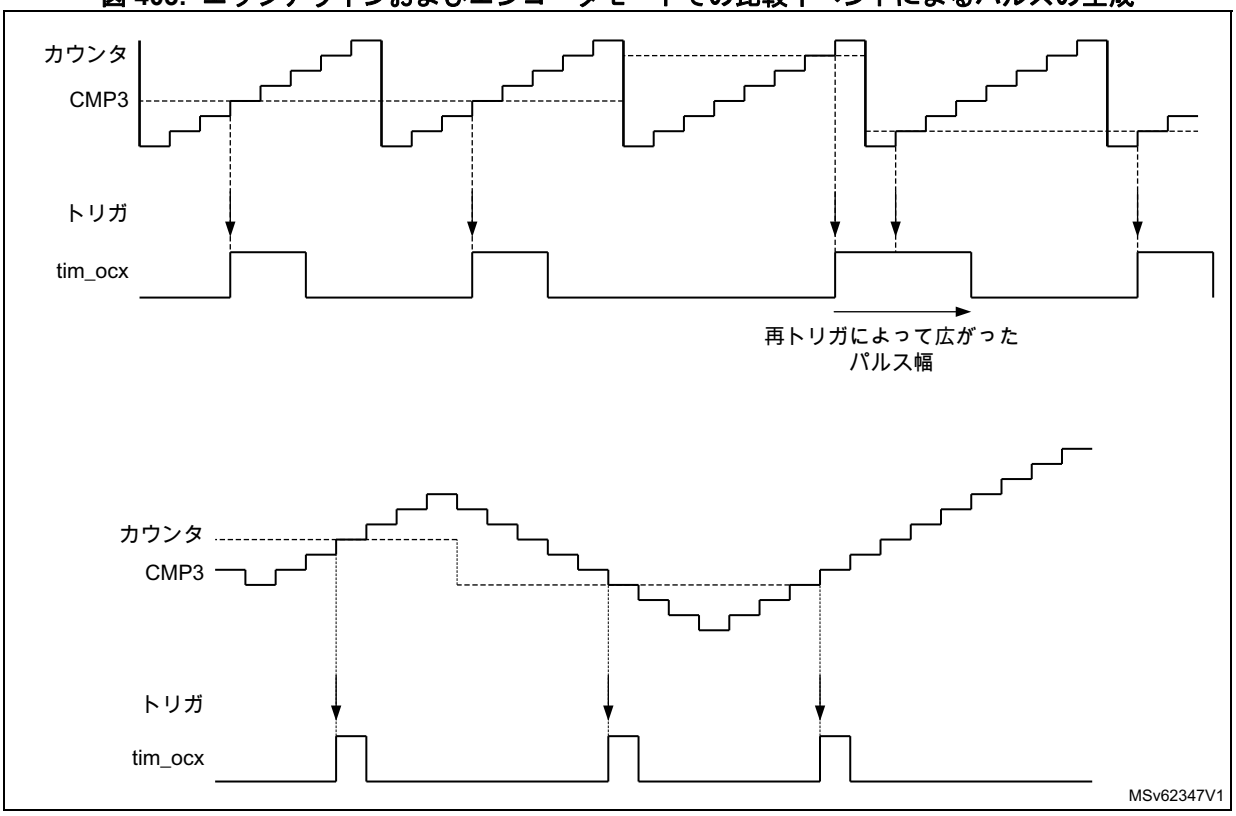
このモードは、エンコーダモードを含むすべてのスレーブモードの選択で、エッジアラインおよびセンタアラインのカウントモードで使用できます。これは、チャネル 3 とチャネル 4 のみで使用できます。以下の図 402 に示すように、パルス発生回路はただ 1 つであり、2 つのチャネルで共有されます。

図 402. パルス発生回路



下記の図 403 は、エッジアラインおよびエンコーダ動作モードでパルスがどのように生成されるかを示しています。

図 403. エッジアラインおよびエンコーダモードでの比較イベントによるパルスの生成



この出力比較モードは、TIMx_CCMR2 レジスタの OC3M[3:0] および OC4M[3:0] ビットフィールドを使用して選択されます。

パルス幅は、次のように、PWPRSC[2:0] ビットに従ってプリスケールされた特定のクロックを使用し、レジスタの PW[7:0] ビットフィールドを使用してプログラムします。

$t_{PW} = PW[7:0] \times t_{PWG}$

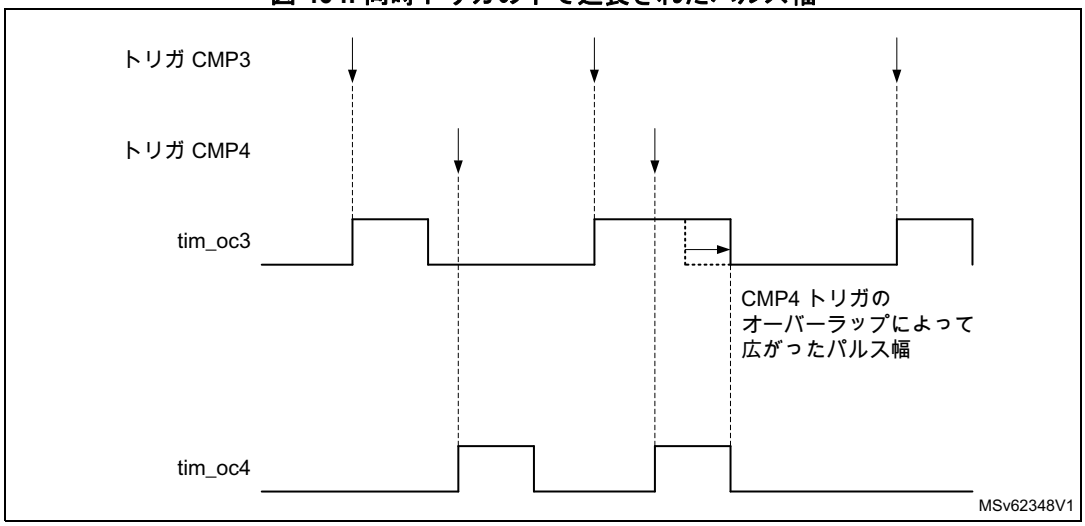
ここで、$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$。

上記は、プリスケーラの値に応じた分解能と最大値を示します。

パルスは再トリガ可能です。パルスが駆動されている間に新しいトリガが入ると、パルスは延長されます。

注： **2 つのチャネルが同時に有効になっている場合、いずれかのチャネルのトリガが、同時出力に生成されたパルスとオーバーラップしない限り、両チャネルのパルスは互いに独立して発行されます。反対に、2 つのトリガがオーバーラップしている場合、最初に到着したトリガに関連するパルス幅は（再トリガのために）拡張されますが、後に到着したトリガのパルス幅は正しくなります（以下の図 404 に示すとおり）。**

図 404. 同時トリガの下で延長されたパルス幅



29.4.18 エンコーダインタフェースモード

直交エンコーダ

エンコーダ·インタフェース·モードを選択するには、カウンタが tim_ti1 エッジのみをカウントしている場合は SMS = 0001、tim_ti2 エッジのみをカウントしている場合は SMS = 0010、tim_ti1 と tim_ti2 の両方のエッジをカウントしている場合は SMS = 0011 を TIMx_SMCR レジスタに書き込みます。

tim_ti1 と tim_ti2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。CC1NP と CC2NP はクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。

2 つの入力 tim_ti1 と tim_ti2 は、インクリメンタルエンコーダとのインタフェースに使用されます。表 280 を参照してください。カウンタのクロックは、tim_ti1fp1 または tim_ti2fp2（入力フィルタおよび極性選択後は tim_ti1 および tim_ti2。フィルタされず、反転されない場合は tim_ti1fp1=tim_ti1、フィルタされず、反転されない場合は tim_ti2fp2=tim_ti2）の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります（TIMx_CR1 レジスタの CEN ビットが“1”）。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが tim_ti1 のみ、tim_ti2 のみ、または tim_ti1 と tim_ti2 の両方をカウントしている場合でも、DIRビットは、いずれかの入力（tim_ti1 または tim_ti2）の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします（方向に応じて、0 から ARR まで、または ARR から 0 まで）。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケーラ、およびトリガ出力機能は、通常動作を続けます。エンコーダ·モードと外部クロック·モード 2 には相互の互換性がないので、同時に選択することはできません。

このモードでは、カウンタは直交エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します（tim_ti1 と tim_ti2 は同時に切り替わらないと想定しています）。

表 280. カウント方向とエンコーダ信号（CC1P = CC2P = 0）

アクティブエッジ	SMS[3:0]	他方の信号のレベル（tim_ti2 の場合は tim_ti1fp1、tim_ti1 の場合は tim_ti2fp2）	tim_ti1fp1 信号		tim_ti2fp2 信号	
			立ち上がり	立ち下がり	立ち上がり	立ち下がり
tim_ti1 のみカウント x1 モード	1110	ハイ	ダウン	アップ	カウントなし	カウントなし
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti2 のみカウント x1 モード	1111	ハイ	カウントなし	カウントなし	アップ	ダウン
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
tim_ti1 のみカウント x2 モード	0001	ハイ	ダウン	アップ	カウントなし	カウントなし
		ロー	アップ	ダウン	カウントなし	ダウン
tim_ti2 のみカウント x2 モード	0010	ハイ	カウントなし	カウントなし	アップ	ダウン
		ロー	カウントなし	カウントなし	ダウン	アップ
tim_ti1 と tim_ti2 の両方をカウント x4 モード	0011	ハイ	ダウン	アップ	アップ	ダウン
		ロー	アップ	ダウン	ダウン	アップ

直交エンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部トリガ入力に接続して、カウンタのリセットをトリガできます。

図 405 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S = 01（TIMx_CCMR1 レジスタ、tim_ti1fp1 は tim_ti1 に配置）
- CC2S = 01（TIMx_CCMR2 レジスタ、tim_ti2fp2 は tim_ti2 に配置）
- CC1P および CC1NP = 0（TIMx_CCER レジスタ、tim_ti1fp1 非反転、tim_ti1fp1=tim_ti1）
- CC2P および CC2NP = 0（TIMx_CCER レジスタ、tim_ti2fp2 非反転、tim_ti2fp2=tim_ti2）
- SMS=011（TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ）
- CEN=1（TIMx_CR1 レジスタ、カウンタ有効）

図 405. エンコーダインタフェースモードにおけるカウンタの動作例

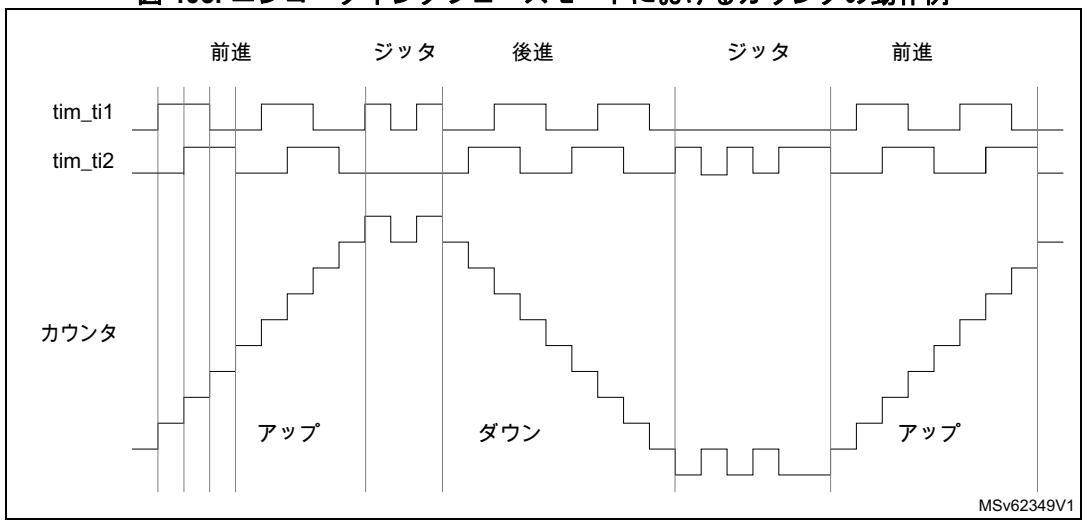
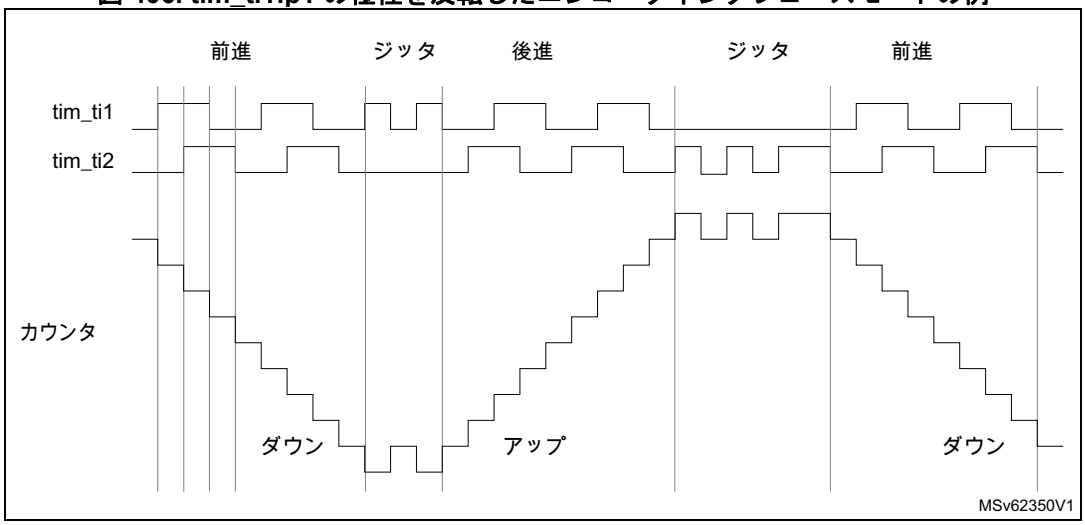


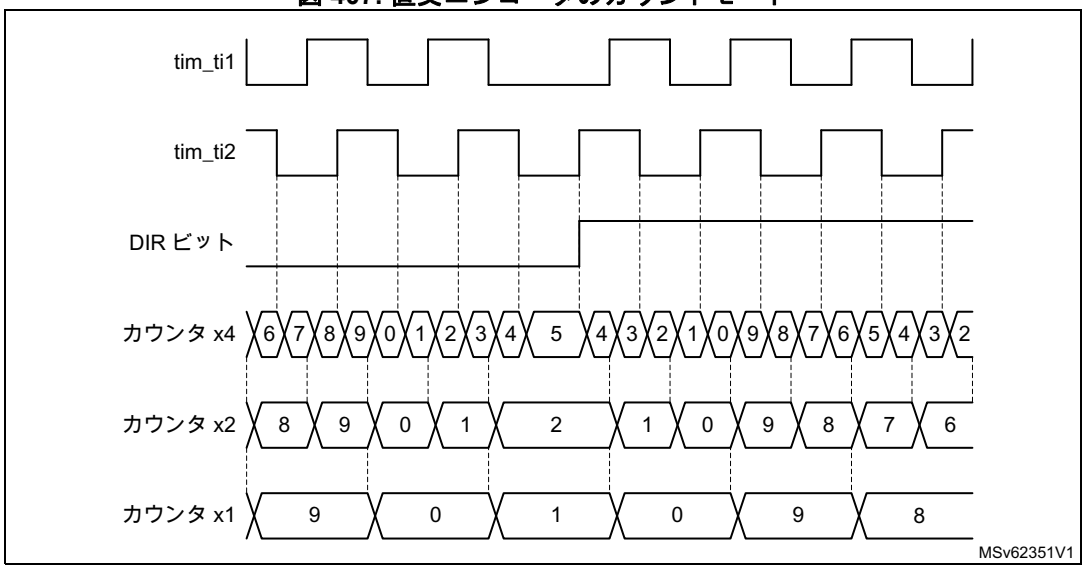
図 406 に、tim_ti1fp1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図 406. tim_ti1fp1 の極性を反転したエンコーダインタフェースモードの例



以下の図 407 は、さまざまなカウントモードでの速度反転中のタイマカウンタ値を示しています。

図 407. 直交エンコーダのカウントモード



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならず、別のタイマによって生成できます）。使用可能なときには、DMA リクエストを通じて値を読み出すことも可能です。

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

クロックプラス方向エンコーダモード

直交エンコーダモードのほか、タイマは他のタイプのエンコーダもサポートしています。

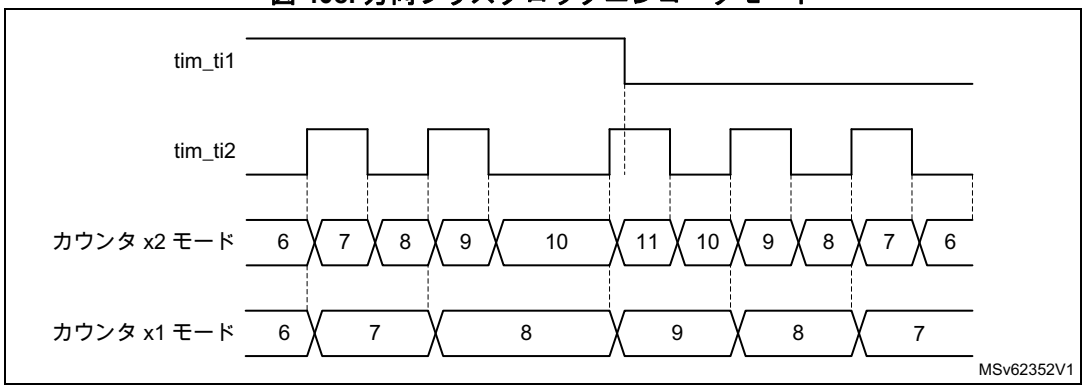
図 408 に示す「クロックプラス方向」モードでは、tim_ti2 の単一ラインでクロックが提供され、tim_ti1 入力を使用して方向が強制されます。

このモードは TIMx_SMCR レジスタの SMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1010：X2 モード。カウンタはクロックの立ち上がりと立ち下がりの両エッジで更新されます。
- 1011：X1 モード。カウンタは 次の CC2P ビット値に従って単一のクロックエッジで更新されます。CC2P = 0 は立ち上がりエッジ検出に対応し、CC2P = 1 は立ち下がりエッジ検出に対応します。

tim_ti1 の方向信号の極性は、次の CC1P ビットで設定されます。0 は正の極性（tim_ti1 がハイのときアップカウントで、tim_ti1 がローのときダウンカウント）に対応し、CC1P = 1 は負の極性（tim_ti1 がローのときアップカウント）に対応します。

図 408. 方向プラスクロックエンコーダモード



方向クロックエンコーダモード

図 409 の「方向クロック」モードでは、クロックは 2 本のラインで提供され、1 本のアップカウントクロックラインと 1 本のダウンカウントクロックラインを持つというように、方向に応じて一度に 1 つが提供されます。

このモードは TIMx_SMCR レジスタの SMS[3:0] ビットフィールドによって、以下のように有効にされます。

- 1100：x2 モード。カウンタは、2 つのクロックラインのいずれかの立ち上がりと立ち下がりの両エッジで更新されます。CC1P および CC2P ビットは、クロックアイドル状態用のコーディングです。CCxP = 0 はハイレベルアイドル状態に対応し（下記図 409 を参照）、CCxP = 1 はローレベルアイドル状態に対応します（下記図 410 を参照）。
- 1101：x1 モード。カウンタは、CC1P および CC2P ビット値に従って、単一のクロックエッジで更新されます。CCxP = 0 は立ち下がり検出とハイレベルアイドル状態に対応し（下記図 409 を参照）、CCxP = 1 は立ち上がり検出とローレベルアイドル状態に対応します（下記図 410 を参照）。

図 409. 方向クロックエンコーダモード（CC1P = CC2P = 0）

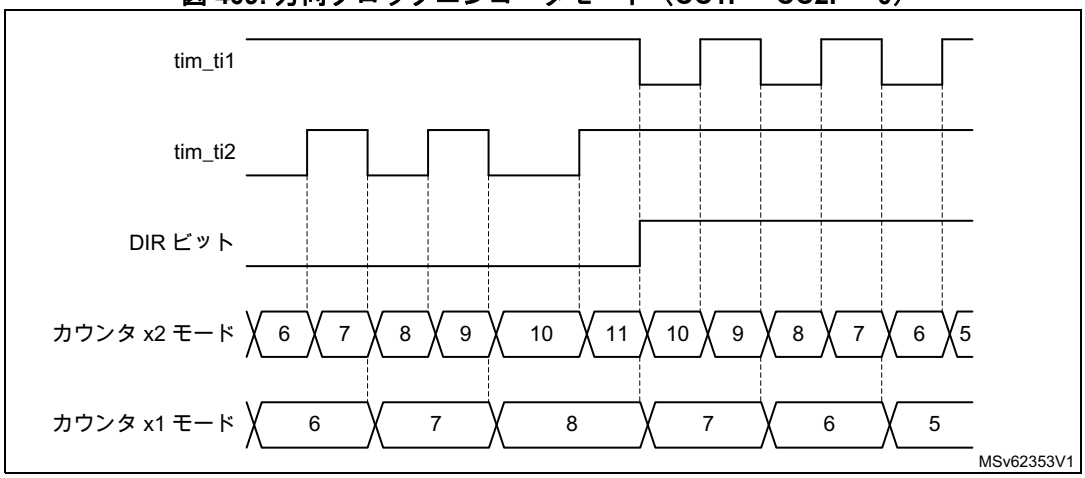
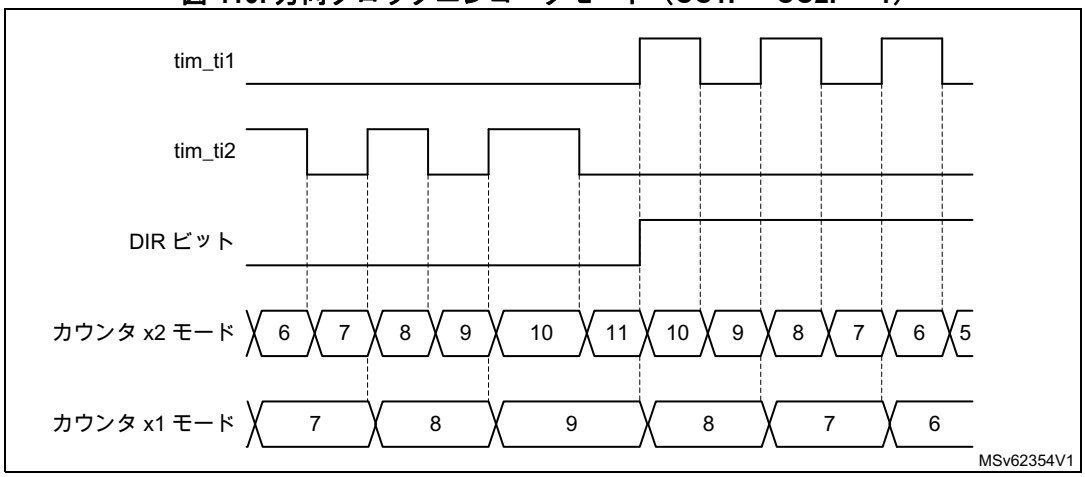


図 410. 方向クロックエンコーダモード（CC1P = CC2P = 1）



以下の表 281 に、任意の入力遷移に対して方向クロックモードがどのように動作するかを詳しく説明します。

表 281. カウント方向 対 エンコーダ信号および極性設定

方向クロックモード	SMS[3:0]	他方の信号のレベル（tim_ti2 の場合は tim_ti1fp1、tim_ti1 の場合は tim_ti2fp2）	tim_ti1fp1 信号		tim_ti2fp2 信号	
			立ち上がり	立ち下がり	立ち上がり	立ち下がり
x2 モード CCxP=0	1100	ハイ	ダウン	ダウン	アップ	アップ
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
x2 モード CCxP=1	1100	ハイ	カウントなし	カウントなし	カウントなし	カウントなし
		ロー	ダウン	ダウン	アップ	アップ
x1 モード CCxP=0	1101	ハイ	カウントなし	ダウン	カウントなし	アップ
		ロー	カウントなし	カウントなし	カウントなし	カウントなし
x1 モード CCxP=1	1101	ハイ	カウントなし	カウントなし	カウントなし	カウントなし
		ロー	ダウン	カウントなし	アップ	カウントなし

インデックス入力

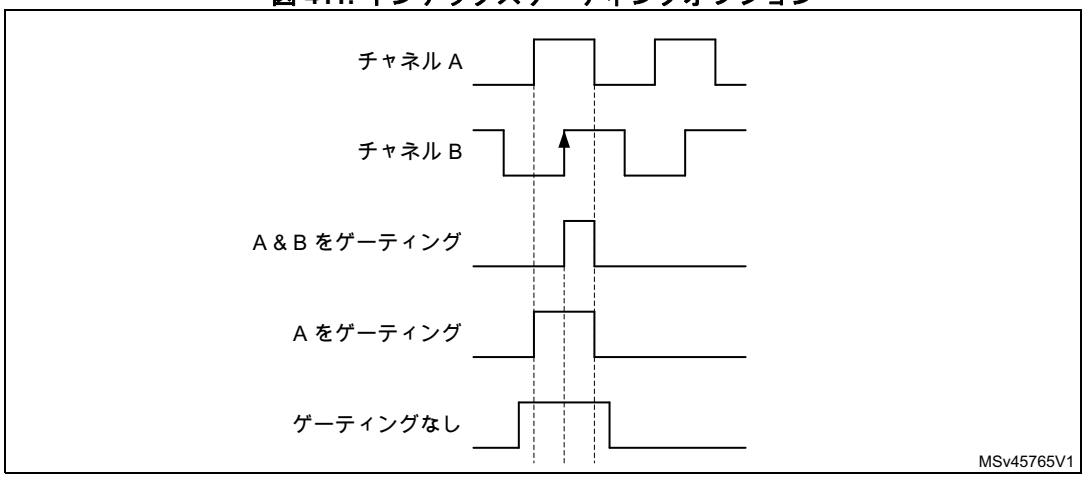
カウンタは、絶対基準位置を示すエンコーダからのインデックス信号によってリセットできます。インデックス信号は、tim_etr_in 入力に接続する必要があります。これは、デジタル入力フィルタを使用してフィルタリングできます。

インデックス機能は、TIMx_ECR レジスタの IE ビットで有効にされます。IE ビットは、SMS[3:0] ビットフィールドが次の値であるときに、エンコーダモードでのみセットする必要があります。0001、0010、011、1010、1011、1100、1101、1110、1111。

以下の図 411 に示すように、インデックスパルス調整のいくつかのオプションを備えた市販のエンコーダが提案されています。

- A および B でゲート処理：パルス幅は 1 つのチャネル周期の 1/4 であり、A および B の両方のエッジにそろえられます。
- A でゲート処理（または B でゲート処理）：パルス幅は 1 つのチャネル周期の 1/2 であり、チャネル A（または チャネル B）の 2 つのエッジにそろえられます。
- ゲート処理なし：パルス幅は最大 1 チャネル周期であり、どのエッジにもそろえられません。

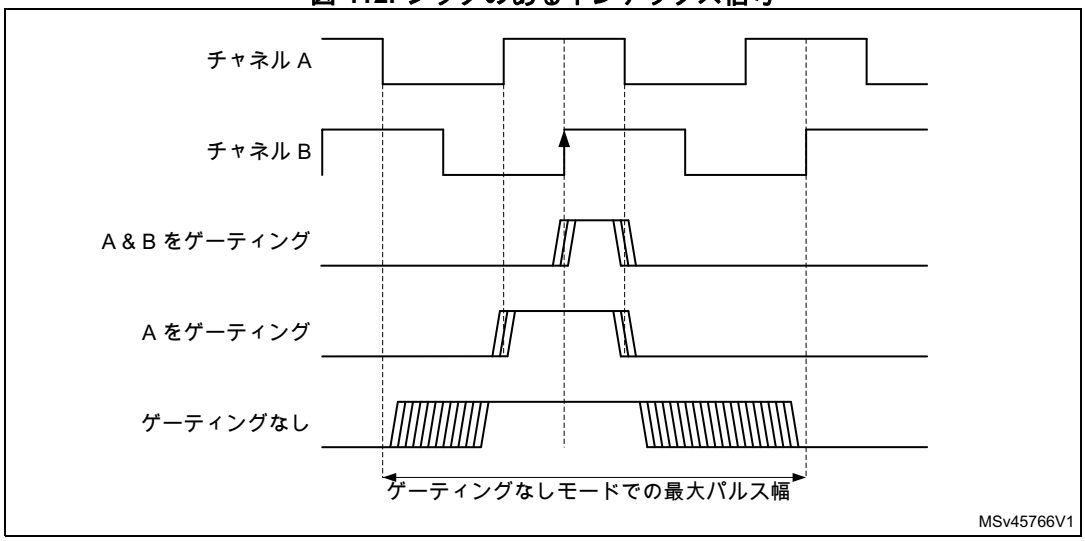
図 411. インデックスゲーティングオプション



以下の図 412 に示すように、ゲーティングモードに関係なく、回路はインデックス信号のジッタを許容します。

ゲート処理なしモードでは、信号はエンコーダの 2 周期より厳密に小さくなければなりません。パルス幅がエンコーダの 2 周期以上の場合、カウンタは複数回リセットされます。

図 412. ジッタのあるインデックス信号



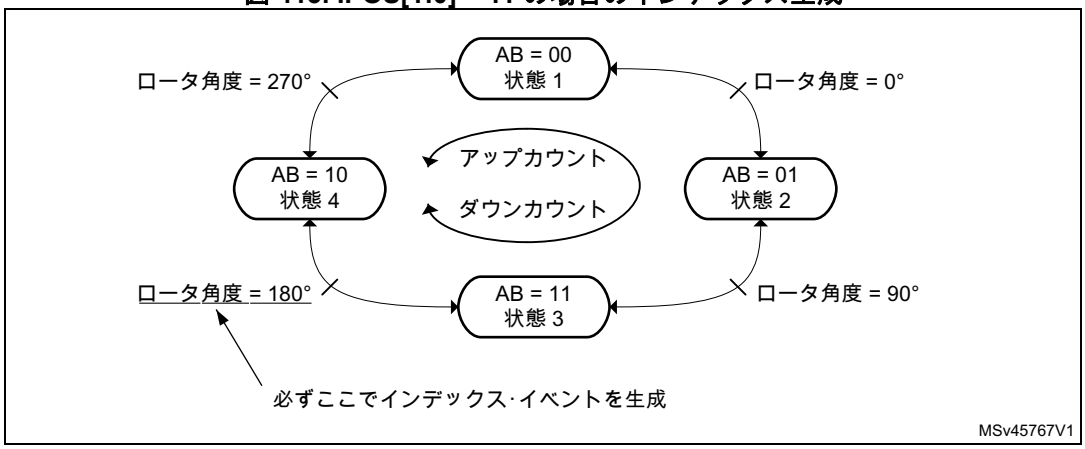
タイマは、特定のプログラミングを必要とせずに、3 つのゲーティングオプションを同様にサポートします。TIMx_ECR レジスタの IPOS[1:0] ビットフィールドを使用して、インデックスを同期する必要があるエンコーダ状態（チャネル A とチャネル B の状態の組み合わせ）を定義するだけです。

インデックス検出イベントは、速度の反転中に対称動作を確保するために、カウント方向に応じて異なる動作をします。

- カウンタはアップカウント中（DIR ビット = 0）に、リセットされます。
- カウンタはダウンカウント時に、TIMx_ARR に設定されます。

これにより、カウント方向に関係なく、まったく同じ機械的角度位置でインデックスを生成できます。以下の図 413 は、単純な例（機械的回転ごとに 4 つのエッジを提供するエンコーダ）について、生成されるインデックスの位置を示しています。

図 413. IPOS[1:0] = 11 の場合のインデックス生成



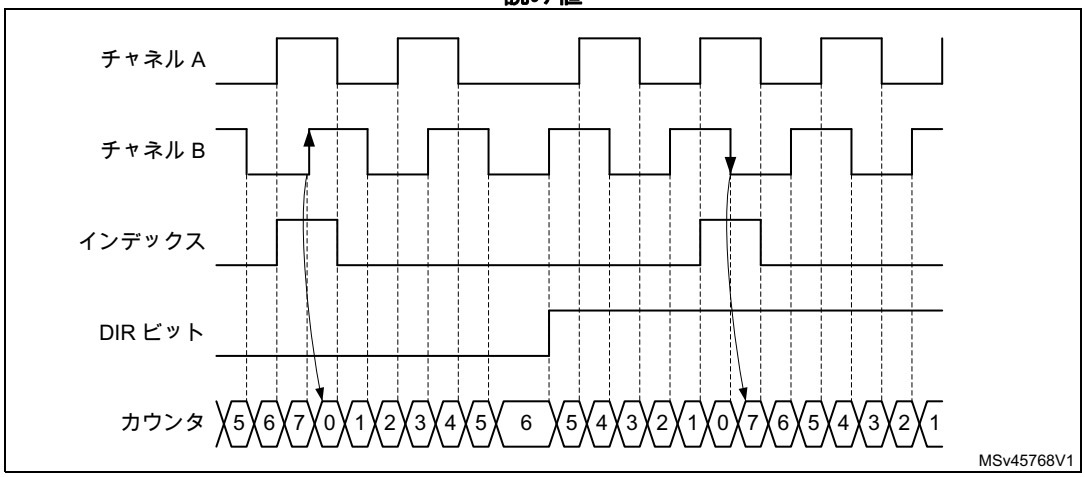
以下の図 414 に、IPOS[1:0] = 11 の場合の波形と対応する値を示します。カウンタの値が強制される瞬間は、カウント方向に応じて自動的に調整されることを示しています。

- アップカウント（DIR ビット = 0）の場合、エンコーダの状態が「11」（ChA = 1、ChB = 1）のとき、カウンタは 0 にセットされます。
- ダウンカウント（DIR ビット = 1）の場合、「11」状態を終了するとき、カウンタは TIMx_ARR にセットされます。

インデックス検出イベント時に割込みを発行できます。

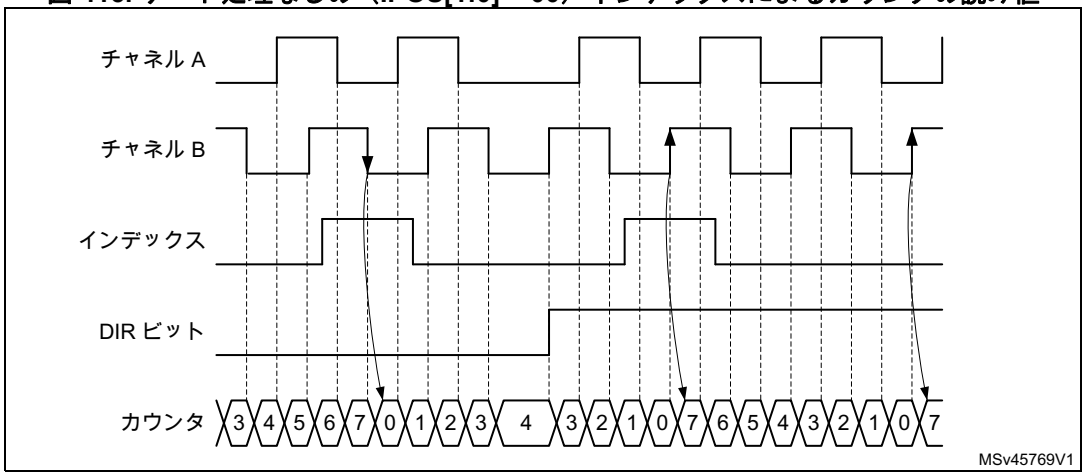
矢印は、どの遷移でインデックスイベント割込みが生成されるかを示しています。

図 414. チャネル A でゲート処理された（IPOS[1:0] = 11）インデックスによるカウンタの読み値



以下の図 415 に、ゲート処理なしモードの場合の波形と対応する値を示します。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図 415. ゲート処理なしの（IPOS[1:0] = 00）インデックスによるカウンタの読み値



以下の図 416 は、さまざまなパルスアライメントシナリオでの「A & B でゲート処理」モードの処理方法を示しています。矢印は、どの遷移でインデックスイベントが生成されるかを示しています。

図 416. チャネル A および B でゲート処理されたインデックスによるカウンタの読み値

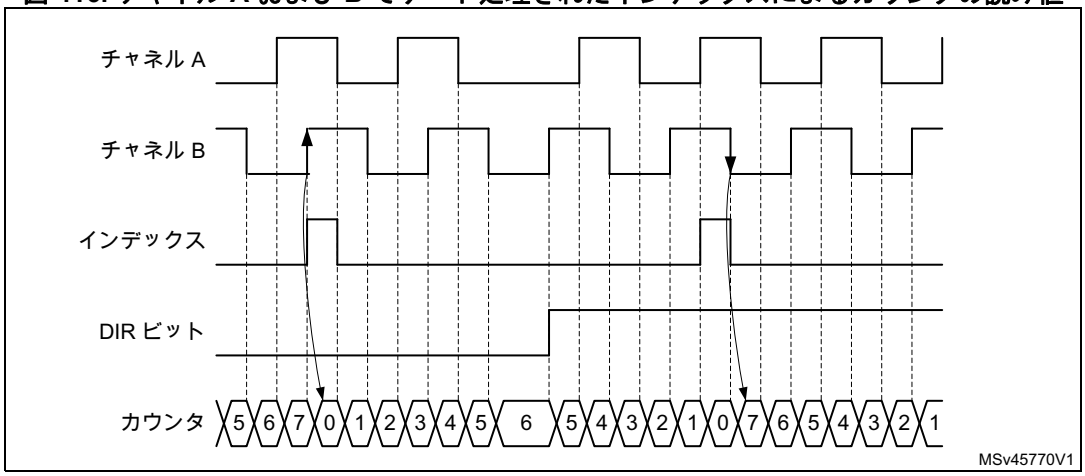


図 417 と 図 418 には、後続のインデックスパルスが、エンコーダクロック周期の 4 分の 1 よりも狭いことがあるケースが詳しく示されています。

図 417. 狭いインデックスパルスの場合のエンコーダモードの動作（IPOS[1:0] = 11）

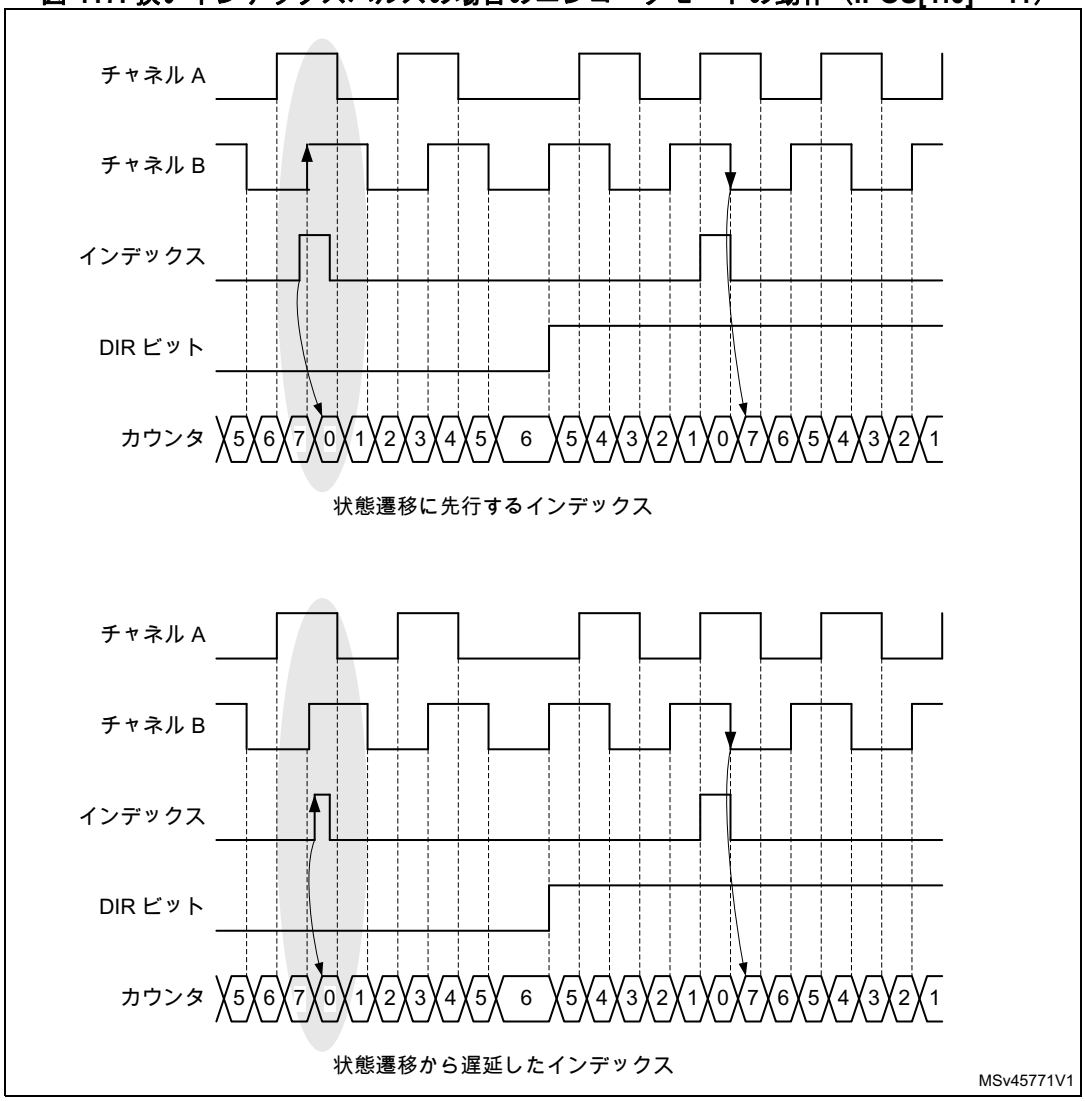
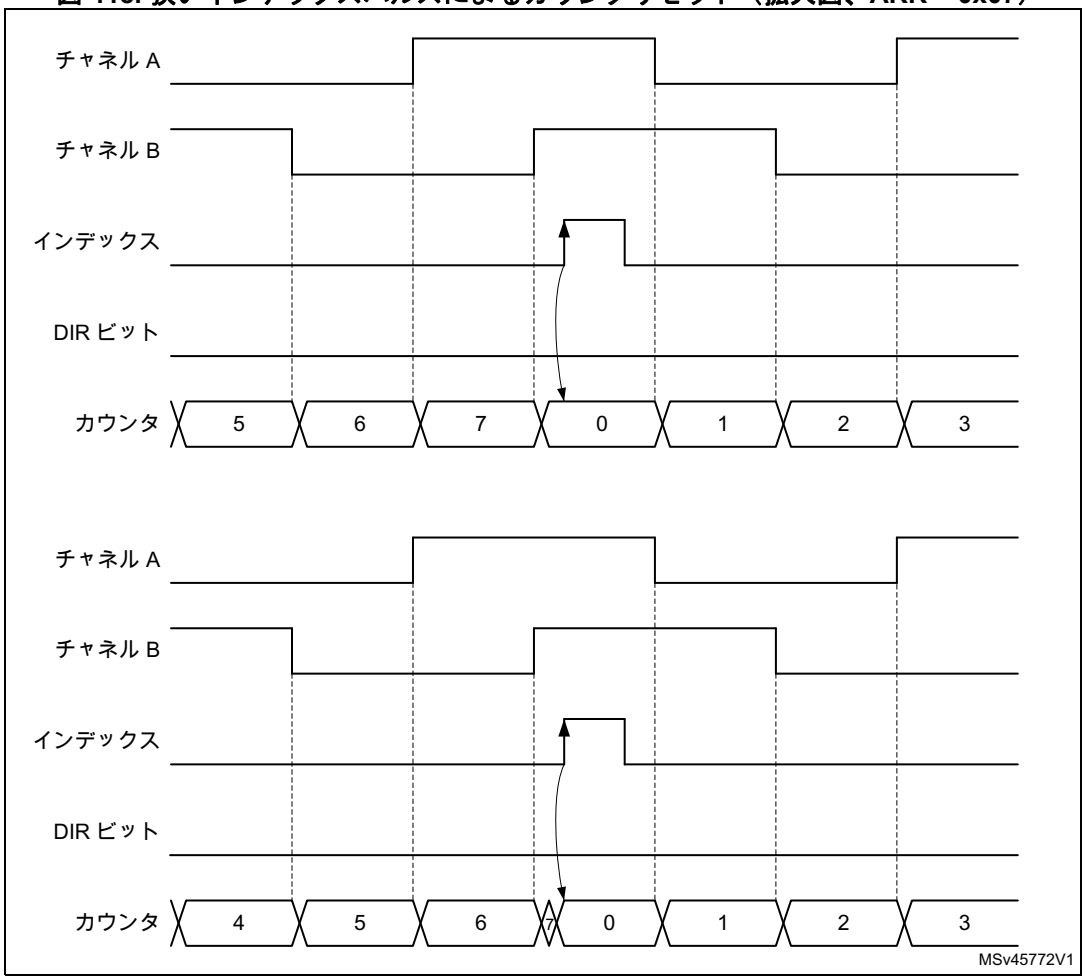
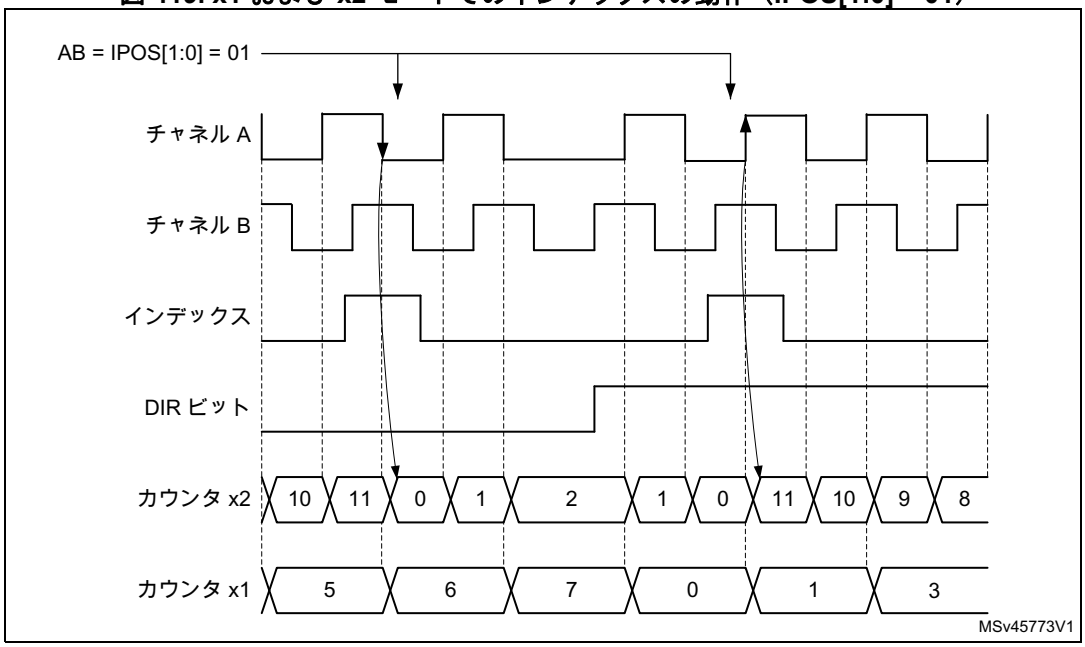


図 418. 狭いインデックスパルスによるカウンタリセット（拡大図、ARR = 0x07）



下記の図 419 は、x1 および x2 モードでインデックスがどのように管理されるかを示します。

図 419. x1 および x2 モードでのインデックスの動作（IPOS[1:0] = 01）



インデックスの方向感受性

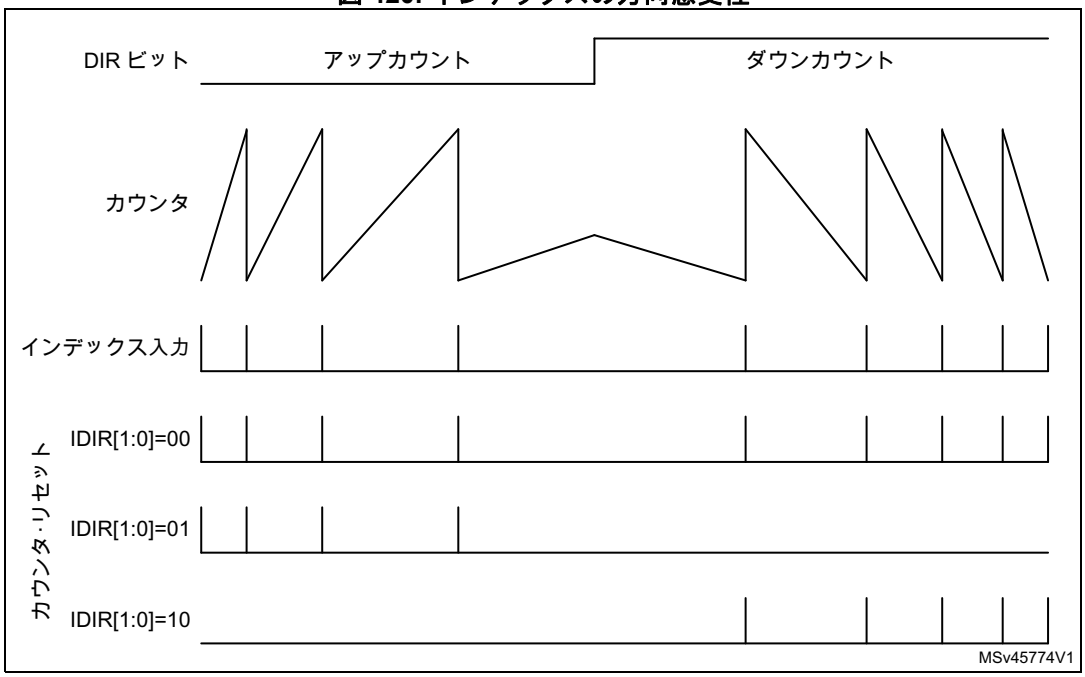
TIMx_ECR レジスタの IDIR[1:0] ビットフィールドにより、選択したカウント方向でのみインデックスをアクティブにできます。

以下の図 420 は、IDIR[1:0] 値に応じた、インデックスとカウンタのリセットイベントの関係を示しています。

注 : **IDR[1:0] ビットフィールドは、IE ビットがリセットされているとき（インデクスモード無効時）に書き込む必要があります。**

注 : **インデックスの方向感受性は、クロックプラス方向モードではサポートされていません。SMS[3:0] = 1010 または 1011 の場合、IDIR[1:0] は 00 にセットしなければなりません。**

図 420. インデックスの方向感受性

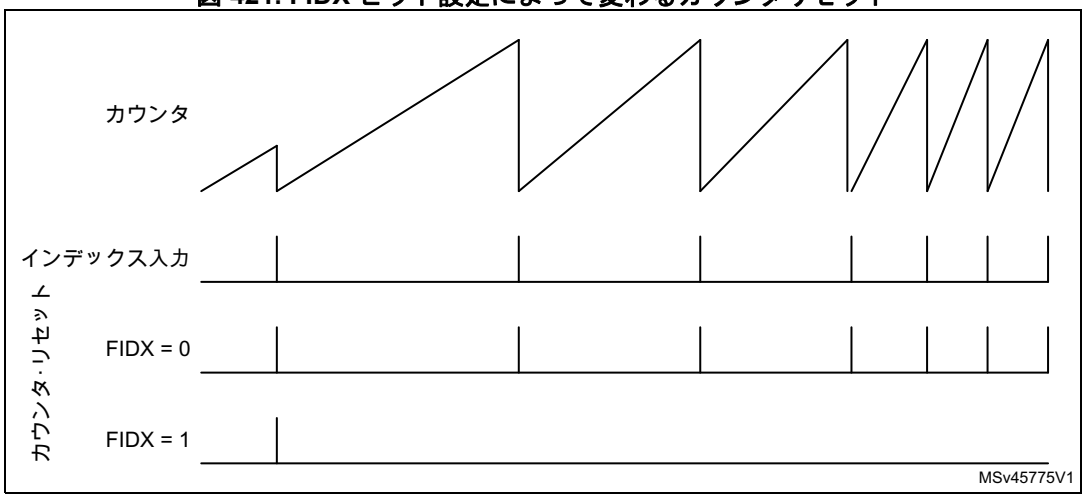


特殊な最初のインデックスイベント管理

TIMx_ECR レジスタの FIDX ビットによって、以下の図 421 に示すように、インデックスを 1 回だけ取得できます。最初のインデックスが到着すると、その後のインデックスは無視されます。必要に応じて、FIDX ビット に 0 を書き込み、再度 1 に設定することにより、回路を再びインデックスを取得できる状態にすることができます。

注： **FIDX = 1 のとき、位置 0（インデックスがアクティブ）で方向が変わると、インデックスを 2 回発行できます（IDXF フラグが設定されます）。**

図 421. FIDX ビット設定によって変わるカウンタリセット



非直交モードでのインデックス管理

以下の図 422 および図 423 では、SMS[3:0] ビットフィールドが 1010、1011、1100、1101 に等しい場合に、方向クロックモードおよびクロックプラス方向モードでインデックスがどのように管理されるかを詳しく説明します。

これらのモードの両方で、インデックスの検出は次のように IPOS[0] ビットで設定されます。

- IPOS[0] = 0：インデックスはクロックのローレベルで検出されます。
- IPOS[0] = 1：インデックスはクロックのハイレベルで検出されます。

IPOS[1] ビットは意味がありません。

図 422. クロックプラス方向モードでのインデックスの動作（IPOS[0] = 1）

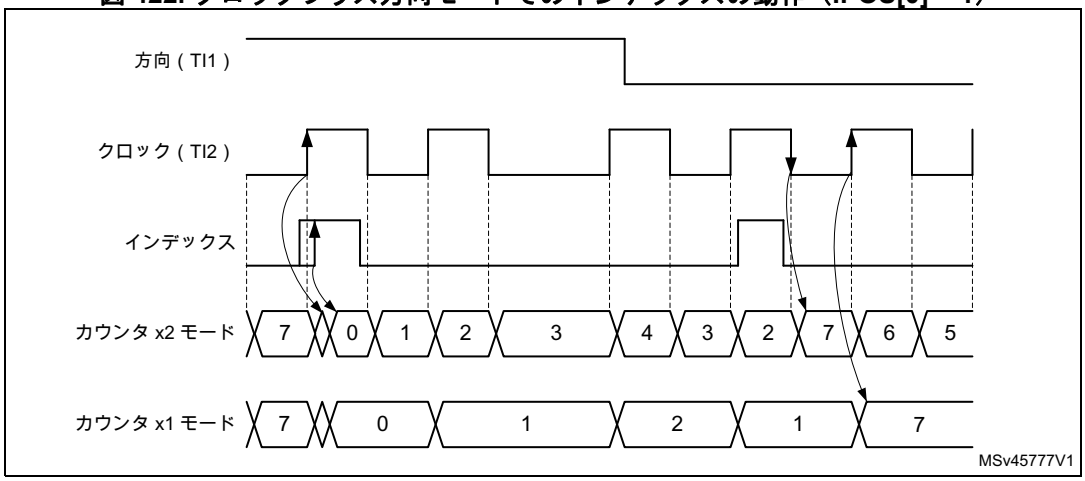
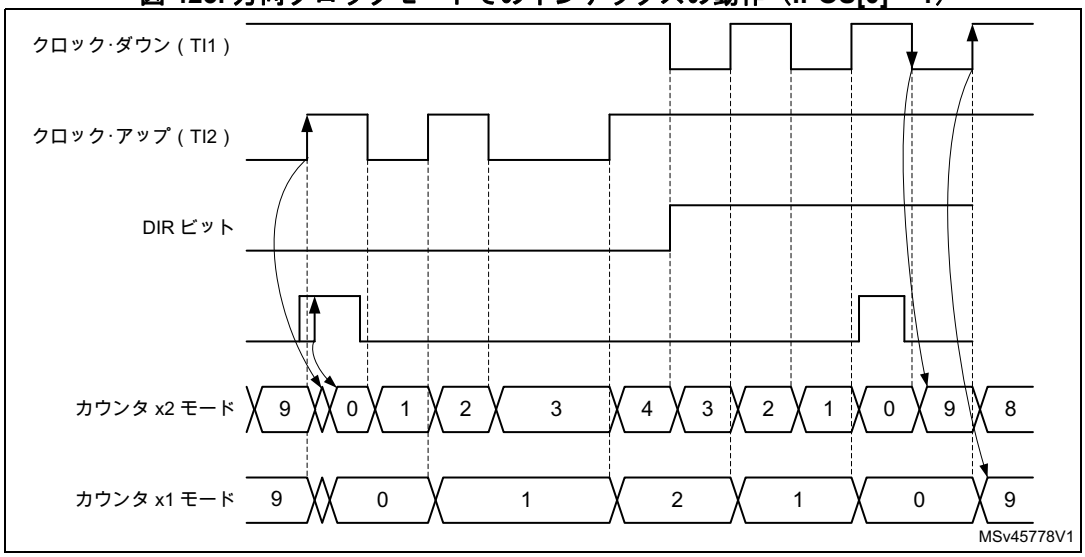


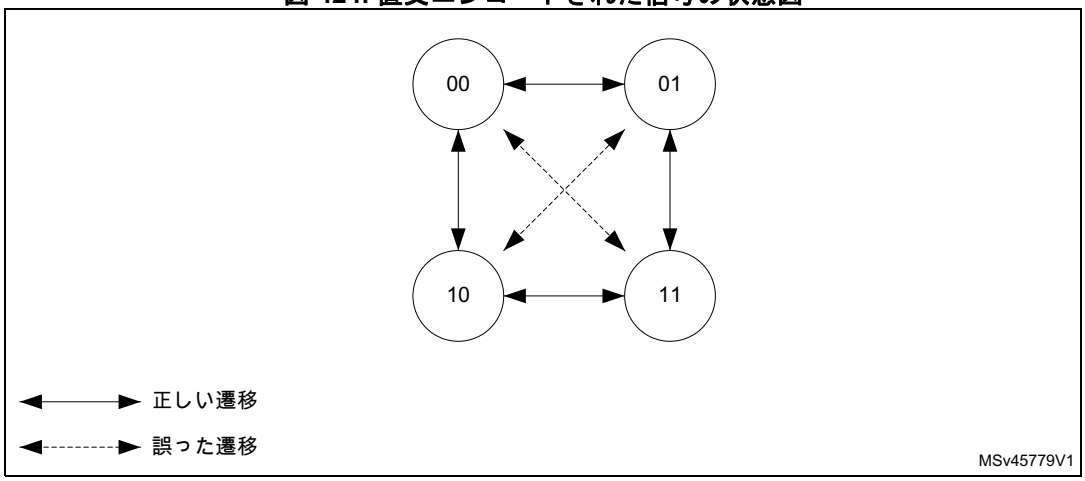
図 423. 方向クロックモードでのインデックスの動作（IPOS[0] = 1）



エンコーダのエラー管理

2 つの直交信号が使用できるエンコーダ構成の場合、遷移エラーを検出できます。2 つの入力の読み値は、以下の図 424 で、状態図として表すことができる 2 ビットのグレーコードに対応しています。一度に 1 つのビットだけが変化することが予測されています。間違った遷移があると、TIMx_SR ステータス·レジスタの TERRF 割込みフラグがセットされます。TIMx_DIER レジスタの TERRIE ビットがセットされている場合、遷移エラー割込みが生成されます。

図 424. 直交エンコードされた信号の状態図



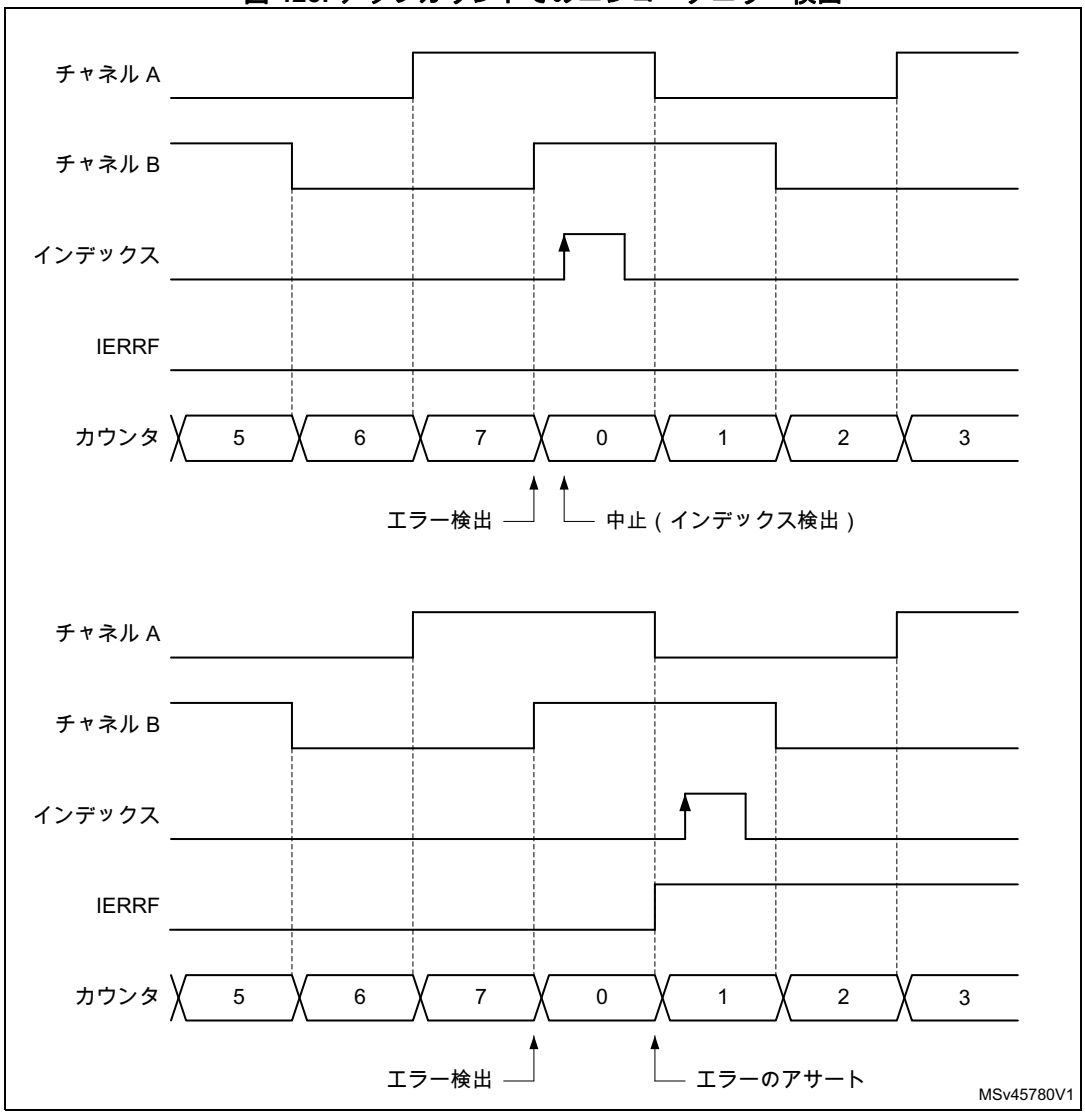
インデックス信号を持つエンコーダの場合、1 回転あたりのパルスが過剰になる異常動作を検出できます。1 回転あたり N 個のパルスを生成するエンコーダでは、1 回転あたり 4xN カウントが得られます。インデックス信号によって、4xN クロック周期ごとにカウンタがリセットされます。

カウンタ値がインデックス·イベントなしに TIMx_ARR から 0 にインクリメントされるか、0 から TIMxARR 値にデクリメントされると、インデックス位置エラーとして報告されます。

オーバーフロー閾値は、TIMx_ARR レジスタを使用してプログラムします。1,000 ラインのエンコーダでは、カウンタ値が 0 ~ 3999 になります（4x 読み取りモードの場合）。オーバーフロー検出閾値は、TIMx_ARR = 3999 + 1 = 4000 を設定してプログラムする必要があります。

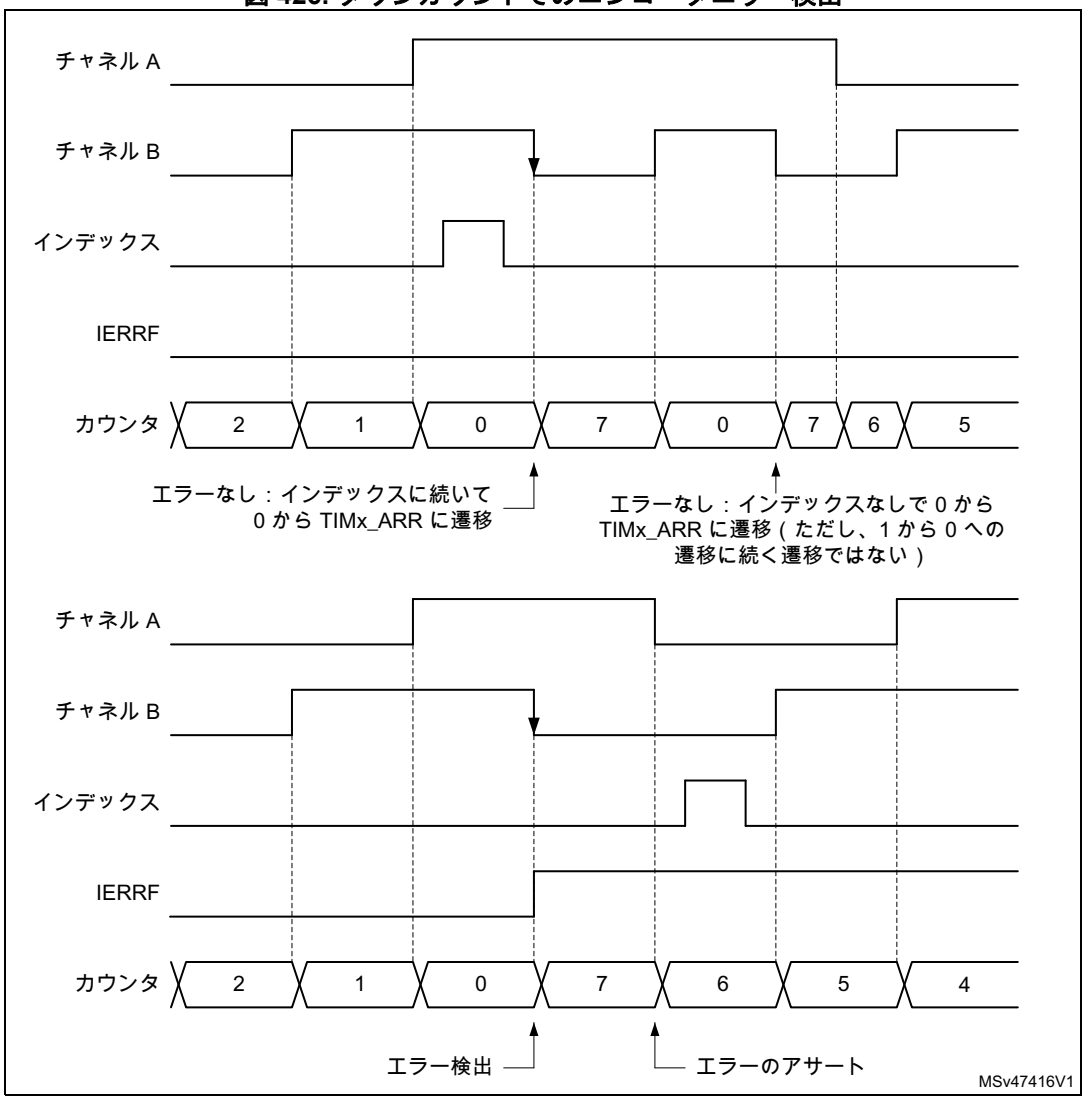
アップカウントでは、エラーアサーションは 0 から 1 への遷移まで遅延されます。これは、以下の図 425 に示すように、A および B ゲート処理モードでの狭いインデックスパルスへの対処です。

図 425. アップカウントでのエンコーダエラー検出



ダウンカウントモードでは、検出は 1 から 0 への事前遷移によって条件付けられます。これは、以下の図 426 に示すように、A および B ゲート処理モードでの狭いインデックスパルスに対処し、インデックス検出直後にエンコーダが TIMx_ARR と 0 の間でディザリングする場合の誤ったエラー検出を回避するためです。

図 426. ダウンカウントでのエンコーダエラー検出



インデックスエラーによって、TIMx_SR ステータスレジスタの IERRF 割込みフラグがセットされます。TIMx_DIER レジスタの IERRIE ビットがセットされている場合、インデックス･エラー割込みが生成されます。

エンコーダの機能的割込み

エンコーダモードでは、以下の割込みも使用できます。

- 方向変更：エンコーダ･モードでのカウント方向の変更によって、TIMx_CR1 レジスタの DIR ビットが反転します。方向変更によって、TIMx_SR ステータス･レジスタの DIRF 割込みフラグがセットされます。TIMx_DIER レジスタの DIRIE ビットがセットされている場合、方向変更割込みが生成されます。

- インデックス·イベント：インデックス·イベントによって、TIMx_SR ステータス·レジスタの IDXF 割込みフラグがセットされます。TIMx_DIER レジスタの IDXIE ビットがセットされている場合、インデックス割込みが生成されます。

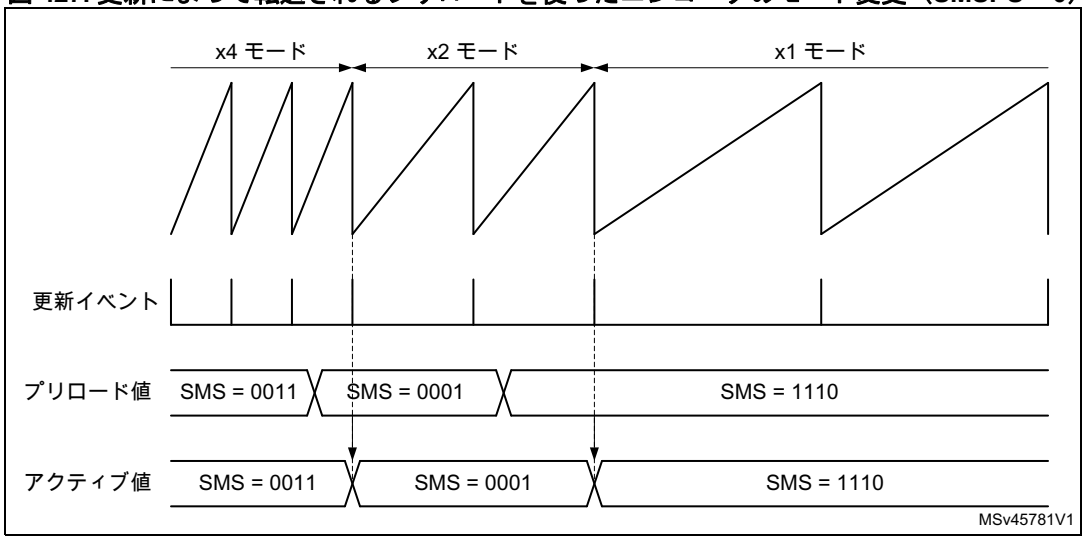
実行時エンコーダ·モード更新のためのスレーブ·モード選択プリロード

実行中に、あるエンコーダモードを別のモードに切り替える必要があることがあります。以下の図 427 に示すように、これは通常、高速時に更新割込みレートを低下させるために、x4 モードから x2 に、x1 に切り替えることにより、実行されます。

この目的のために、SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCR レジスタの SMSPE イネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、TIMx_SMCR レジスタの SMSPS ビットで選択できます。

- SMSPS = 0：転送は、アップカウント時にカウンタがオーバーフローするとき、およびダウンカウント時にアンダーフローするときに発生する更新イベント（UEV）によってトリガされます。このモードは、インデックスが無効な（ビット IE = 0 の）ときにのみ使用する必要があります。
- SMSPS = 1：転送は、インデックスイベントによってトリガされます。

図 427. 更新によって転送されるプリロードを使ったエンコーダのモード変更（SMSPS = 0）



エンコーダクロック出力

エンコーダ·モードの動作原理は、十分な数のクロック·エッジ取得と正確な測定を行うために比較的長い積分時間を必要とするため、低速における高分解能の速度測定に最適とはいえません。

低速でのより良い対策方法は、エッジ間のクロック周期測定です。これは、スレーブタイマを使用して達成できます。タイマによって、tim_trgo 出力にエンコーダクロック情報を出力できます。その後、スレーブタイマはエンコーダクロックエッジごとに周期測定を実行し、速度情報を提供できます。

このモードは TIMx_CR2 レジスタの MMS[3:0] ビットフィールドを 1000 に設定することによって有効になります。これは、以下の SMS[3:0] 値に対して有効です。0001、0010、0011、1010、1011、1100、1101、1110、1111。その他の SMS[3:0] コードは許されず、予想外の動作につながることがあります。

29.4.19 方向ビットの出力

tim_oc3 および tim_oc4 出力信号に、タイマから方向信号を出力できます（TIMx_CR1 レジスタの DIR ビットのコピー）。これを達成するには、TIMx_CCMR2 レジスタの OC3M[3:0] または OC4M[3:0] ビットフィールドを 1011 にセットします。

この機能を使用すると、エンコーダモードでカウント方向（または回転方向）を監視すること、またはセンタアライン PWM モードでアップ／ダウンフェーズを示す信号を得ることができます。

29.4.20 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）を連続コピーします。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。バックグラウンドタスク（カウンタの読出し）と割込み（更新割込み）との間で共有されている処理などによって生じる競合状態を避けることで、角速度の計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

32 ビットのタイマの実装で、IUFREMAP ビットがセットされている場合、カウンタのビット 31 は読出しアクセス時に UIFCPY フラグによって上書きされます（カウンタの最上位ビットには書込みモード時のみアクセス可能）。

29.4.21 タイマ入力 XOR 機能

TIM1xx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 3 つの入力ピンtim_ti1、tim_ti2、および tim_ti3 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

この機能をホールセンサのインタフェースに使用した例をセクション 28.3.29：1136 ページのホールセンサとのインタフェースに示します。

29.4.22 タイマと外部トリガの同期

TIMx タイマは、いくつかのモードで外部トリガを使用して同期できます。すなわち、リセットモード、ゲートモード、トリガモード、リセット + トリガ モード、およびゲート + リセットモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

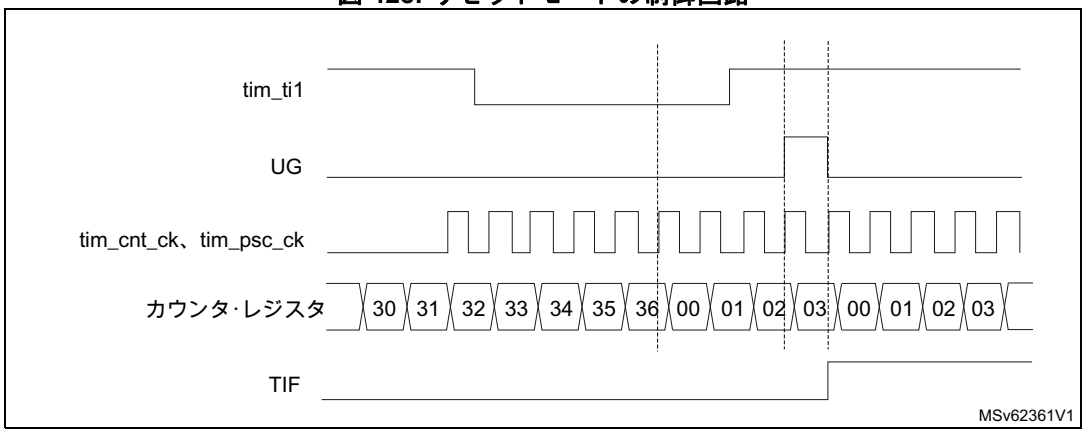
次の例では、tim_ti1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. tim_ti1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1 の立ち上がりエッジまで通常の動作を行います。tim_ti1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。tim_ti1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1 入力の同期回路によるものです。

図 428. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

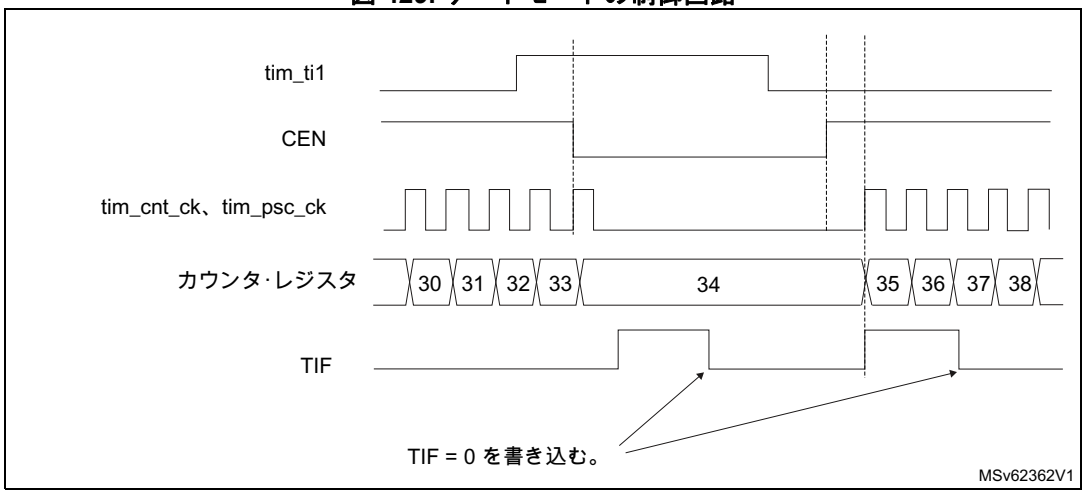
次の例では、アップカウンタは tim_ti1 入力がローのときだけカウントします。

1. tim_ti1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1 がローになると内部クロックでカウントを開始して、tim_ti1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

tim_ti1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1 入力の再同期回路によるものです。

図 429. ゲートモードの制御回路



注： **ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がりと立ち下がり両エッジの検出）はゲートモードでは意味がありません。**

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

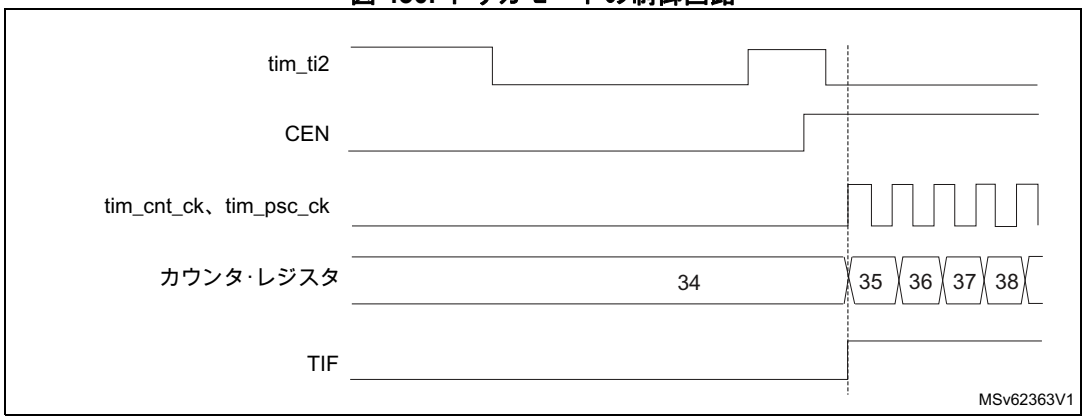
次の例では、アップカウンタは、tim_ti2 入力の立ち上がりエッジに応じて開始します。

1. tim_ti2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして tim_ti2 を選択します。

tim_ti2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2 入力の再同期回路によるものです。

図 430. トリガモードの制御回路



実行時エンコーダモード更新のためのスレーブモード選択プリロード

SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCR レジスタの SMSPE イネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、カウンタがオーバーフローするときに発生する更新イベント（UEV）です。

スレーブモード - リセットモードとトリガモードの組み合わせ

この場合、選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成し、カウンタを開始します。

このモードはワンパルスモードで使用します。

スレーブモード - ゲートモードとリセットモードの組み合わせ

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードによって、範囲外の PWM 信号（最大期待値を超えるデューティサイクル）を検出できます。

スレーブモード – 外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、tim_etr_in 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCR レジスタの TS ビットを通じて tim_trgi として tim_etr_in を選択しないようにしてください。

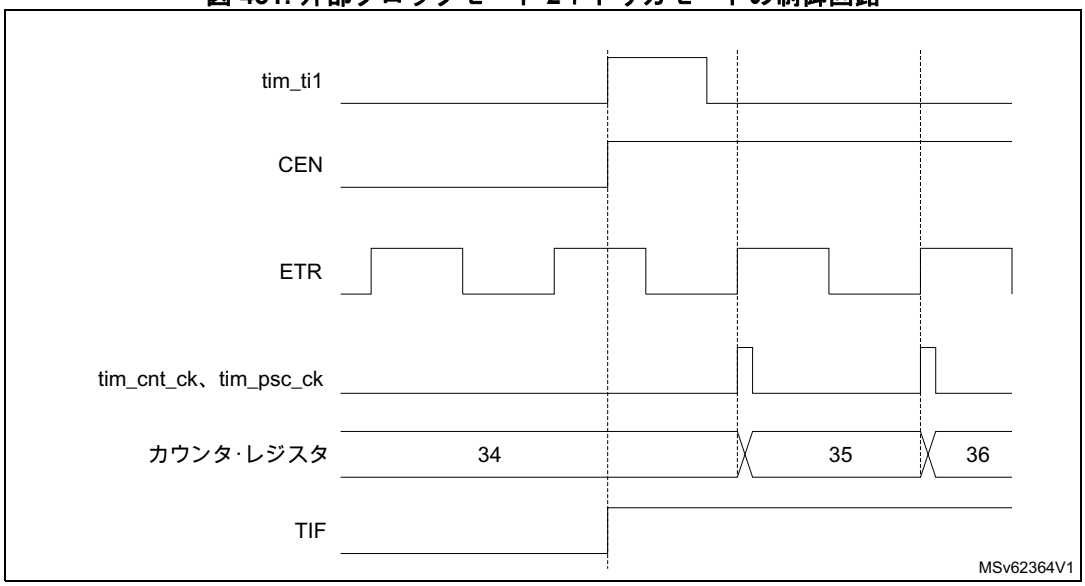
次の例では、アップカウンタは、tim_ti1 の立ち上がりエッジが発生すると、tim_etr_in 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
   – ETF = 0000：フィルタなし
   – ETPS=00：プリスケーラ無効
   – ETP=0：tim_etr_in の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャネル 1 を次のように構成します。
   – IC1F=0000：フィルタなし。
   – キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。
   – TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
   – TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。

tim_ti1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、その後 tim_etr_in の立ち上がりエッジでカウントします。

tim_etr_in 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_etrp 入力の再同期回路によるものです。

図 431. 外部クロックモード 2＋トリガモードの制御回路



29.4.23 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

図 432 と 図 433 にマスタ/スレーブ･タイマの接続例を示します。

図 432. マスタ/スレーブ･タイマの例

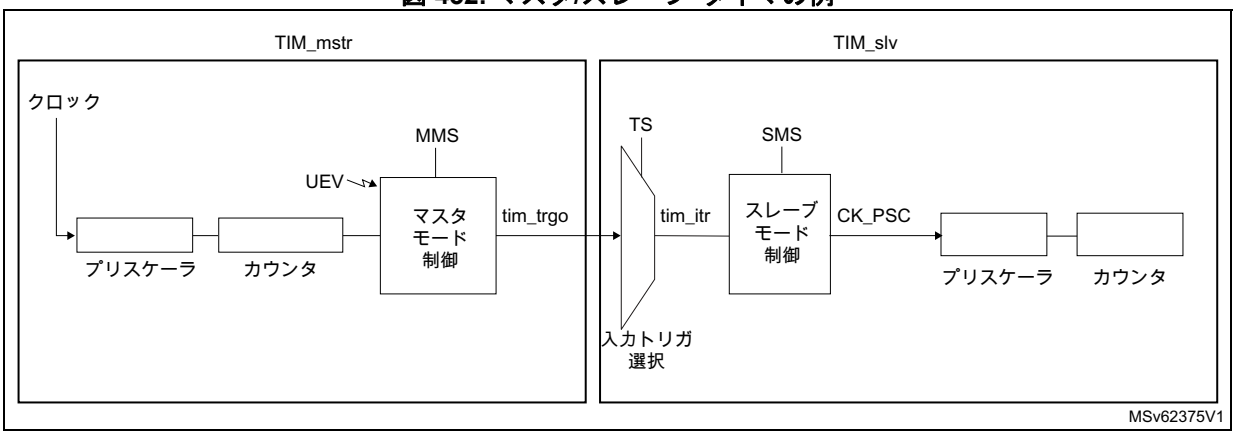
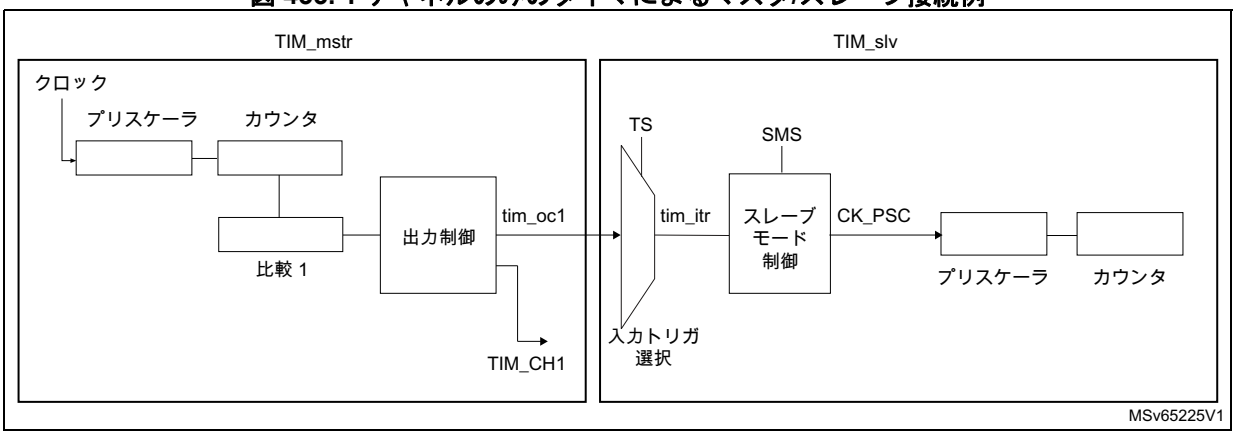


図 433. 1 チャネルのみのタイマによるマスタ/スレーブ接続例



注： **チャネルが 1 つのみのタイマ（図 433 参照）には、マスタ･モードがありません。しかし、tim_oc1 出力信号をスレーブ･タイマのトリガとして使用できます（セクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号の TIMx 内部トリガ接続表を参照してください）。**
tim_oc1 信号のパルス幅が送信先タイマの 2 クロック･サイクル以上になるようにプログラムし、スレーブ･タイマが確実にトリガを検出できるようにする必要があります。
たとえば、送信先タイマの tim_ker_ck クロックが、送信元タイマの 4 分の 1 の速度である場合、OC1 のパルス幅には 8 クロック･サイクルが必要です。

別のタイマのプリスケーラとしたタイマの使用

たとえば、TIM_mstr が TIM_slv のプリスケーラとして動作するように設定できます。図 432 を参照してください。このためには、次の操作を行います。

1. 更新イベント UEV ごとに定期的なトリガ信号を出力するように、TIM_mstr をマスタモードに設定します。TIM_mstr_CR2 レジスタの MMS=010 を書き込んだ場合、更新イベントが生成されるたびに、tim_trgo で立ち上がりエッジが出力されます。
2. TIM_mstr の tim_trgo 出力を TIM_slv に接続するためには、TIM_slv を、ITR2 を内部トリガとして 使用するスレーブモードに設定する必要があります。これは、TIM_slv_SMCR レジスタの TS ビットを通して選択されます（TS=00010 を書き込みます）。
3. 次に、スレーブモードコントローラを外部クロックモード 1 に設定しなければなりません（TIM_slv_SMCR レジスタで SMS=111 を書き込みます）。これにより TIM_slv は、TIM_mstr の定期的なトリガ信号の立ち上がりエッジ（TIM_mstr カウンタのオーバーフローに対応しています）をクロックとして動作します。
4. 最後に、それぞれの CEN ビット（TIMx_CR1 レジスタ）をセットすることによって、両方のタイマを有効にする必要があります。

注： **TIM_mstr のトリガ出力として tim_ocx が選択された場合（MMS=1xx）、その立ち上がりエッジが TIM_slv カウンタのクロックとして使用されます。**

タイマを使用して別のタイマを有効にする

この例では、TIM_slv の有効化を、TIM_mstr の出力比較 1 で制御します。接続については、図 432 を参照してください。TIM_slv は、分周された内部クロックでカウントします（TIM_mstr の tim_oc1ref がハイのときのみ）。両方のカウンタクロック周波数は、tim_ker_ck をプリスケーラで 3 分周したものです（$f_{tim_cnt_ck} = f_{tim_ker_ck}/3$）。

1. TIM_mstr をマスタモードに設定して、その出力比較 1 基準（tim_oc1ref）信号をトリガ出力として送信します（TIM_mstr_CR2 レジスタの MMS=100）。
2. TIM_mstr の tim_oc1ref 波形を設定します（TIM_mstr_CCMR1 レジスタ）。
3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slv をゲートモードに設定します（TIM_slv_SMCR レジスタの SMS=101）。
5. CEN ビット（TIM_slv_CR1 レジスタ）に“1”を書き込んで、TIM_slv を有効にします。
6. CEN ビット（TIM_mstr_CR1 レジスタ）に“1”を書き込んで、TIM_mstr を開始します。

注： **スレーブタイマのカウンタクロックはマスタタイマのカウンタクロックと同期しないので、このモードは TIM_slv カウンタのイネーブル信号にのみ影響を与えます。**

図 434. TIM_mstr の tim_oc1ref による TIM_slv のゲート操作

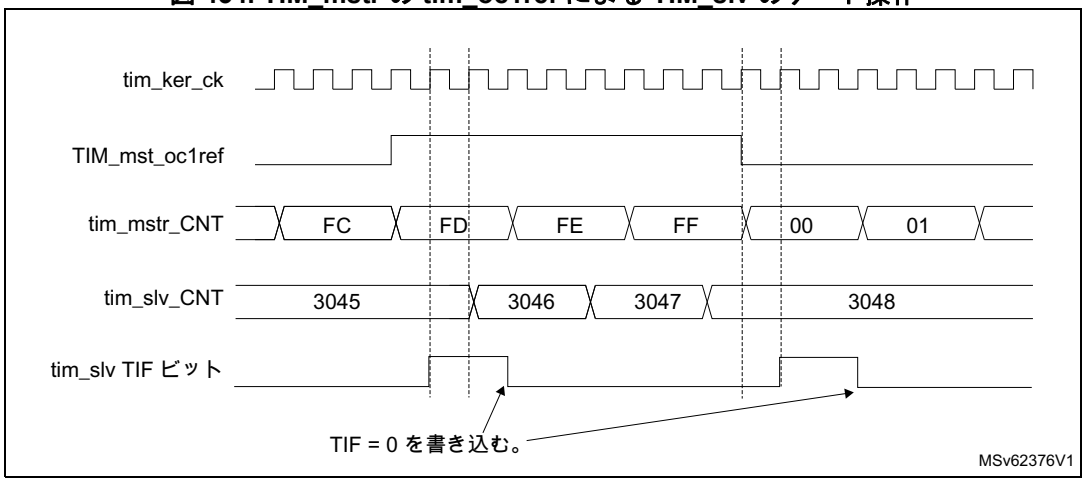
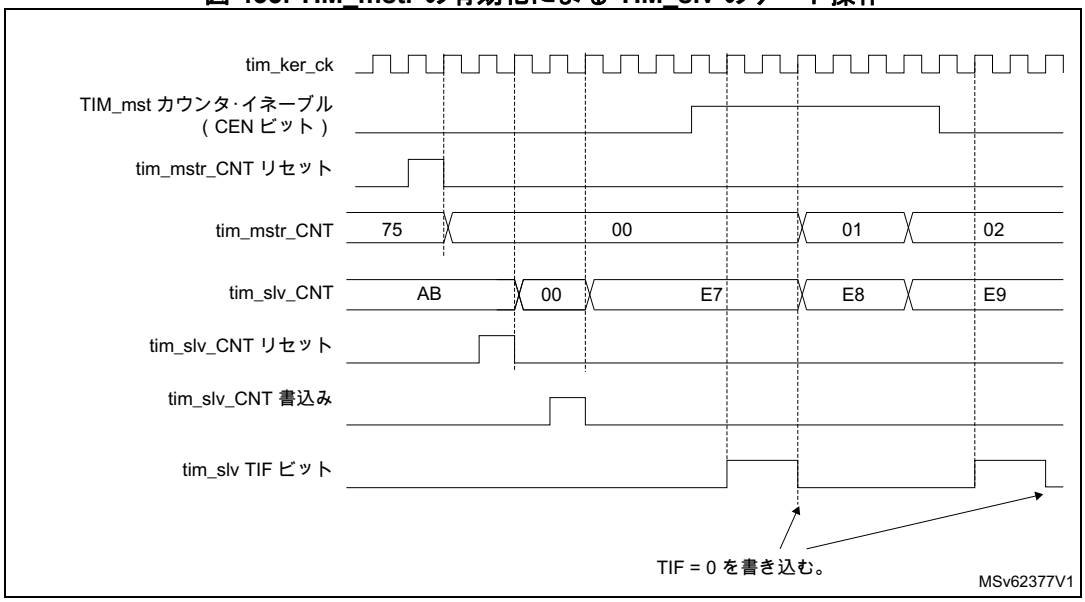


図 434 の例では、TIM_slv のカウンタとプリスケーラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM_mstr を開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。その後、任意の値をタイマカウンタに書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では（図 435 を参照）、TIM_mstr と TIM_slv を同期させます。TIM_mstr はマスタであり、0 からスタートします。TIM_slv はスレーブであり、0xE7 から開始します。プリスケーラの分周比は両方のタイマで同じです。TIM_mstr_CR1 レジスタの CEN ビットに 0 を書き込むことによって TIM_mstr を無効にすると、TIM_slv は停止します。

1. TIM_mstr をマスタモードに設定して、その出力比較 1 基準（tim_oc1ref）信号をトリガ出力として送信します（TIM_mstr_CR2 レジスタの MMS=100）。
2. TIM_mstr の tim_oc1ref 波形を設定します（TIM_mstr_CCMR1 レジスタ）。
3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slv をゲートモードに設定します（TIM_slv_SMCR レジスタの SMS=101）。
5. UG ビット（TIM_mstr_EGR レジスタ）に“1”を書き込むことによって、TIM_mstr をリセットします。
6. UG ビット（TIM_slv_EGR レジスタ）に“1”を書き込むことによって、TIM_slv をリセットします。
7. TIM_slv カウンタ（TIM_slv_CNT）に“0xE7”を書き込んで、TIM_slv を 0xE7 に初期化します。
8. CEN ビット（TIM_slv_CR1 レジスタ）に“1”を書き込んで、TIM_slv を有効にします。
9. CEN ビット（TIM_mstr_CR1 レジスタ）に“1”を書き込んで、TIM_mstr を開始します。
10. CEN ビット（TIM_mstr_CR1 レジスタ）に“0”を書き込んで、TIM_mstr を停止します。

図 435. TIM_mstr の有効化による TIM_slv のゲート操作

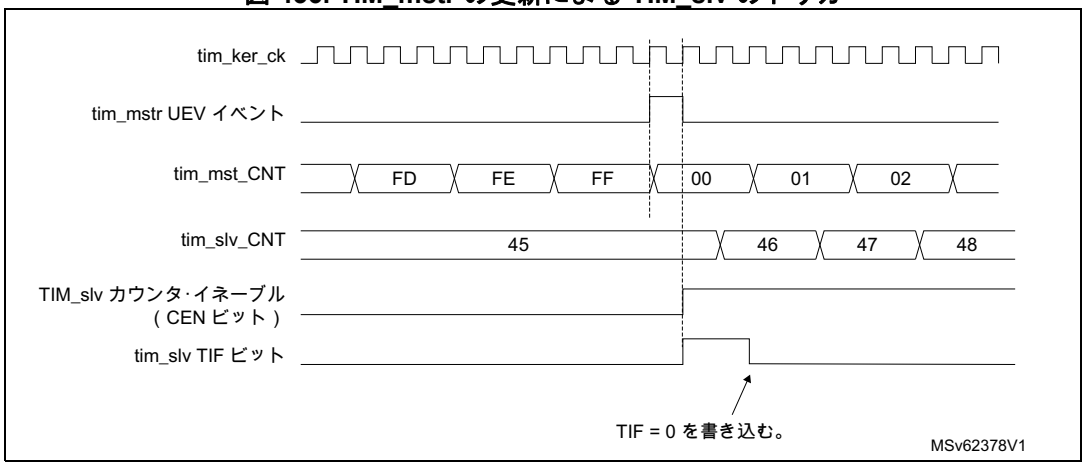


タイマを使用して別のタイマを開始する

この例では、TIM_mstr の更新イベントによって TIM_slv の有効化を設定します。接続については、図 432 を参照してください。TIM_mstr によって更新イベントが生成されると、TIM_slv は、分周された内部クロックで現在値（ゼロである必要はありません）からカウントを開始します。TIM_slv がトリガ信号を受信すると、その CEN ビットが自動的にセットされ、カウンタは TIM_slv_CR1 レジスタの CEN ビットに 0 が書き込まれるまでカウントします。両方のカウンタクロック周波数は、tim_ker_ck をプリスケーラで 3 分周したものです（$f_{tim_cnt_ck} = f_{tim_ker_ck}/3$）。

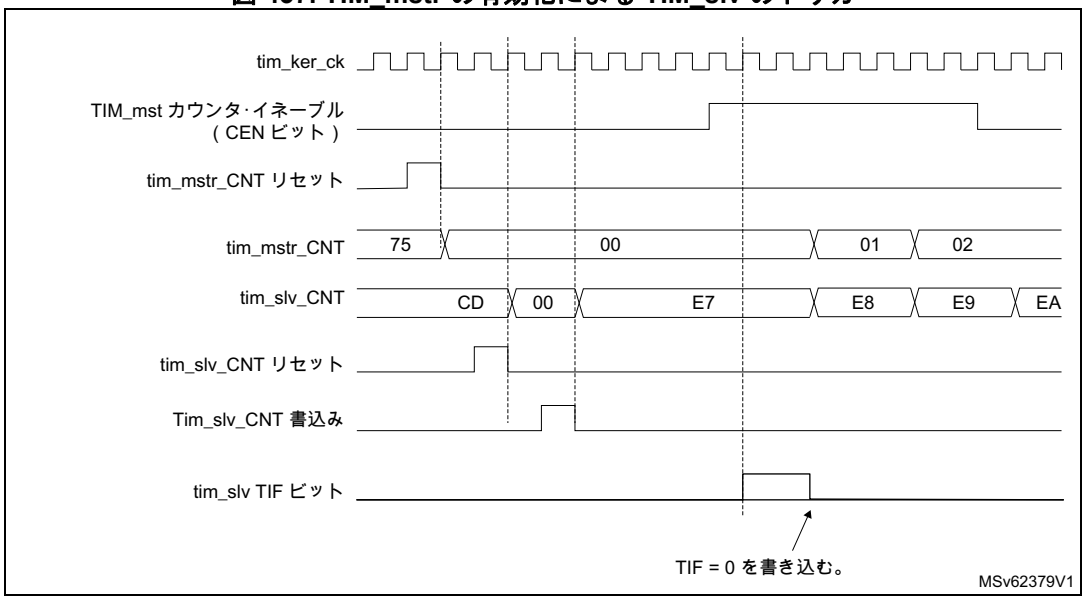
1. TIM_mstr をマスタモードに設定して、その更新イベント（UEV）をトリガ出力として送信します（TIM_mstr_CR2 レジスタの MMS=010）。
2. TIM_mstr の周期を設定します（TIM_mstr_ARR レジスタ）。
3. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00010）。
4. TIM_slv をトリガモードに設定します（TIM_slv_SMCR レジスタの SMS=110）。
5. CEN ビット（TIM_mstr_CR1 レジスタ）に“1”を書き込んで、TIM_mstr を開始します。

図 436. TIM_mstr の更新による TIM_slv のトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化できます。図 437 は 図 436 と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します（TIM_slv_SMCR レジスタの SMS=110）。

図 437. TIM_mstr の有効化による TIM_slv のトリガ



外部トリガに対応して 2 つのタイマを同期して開始します。

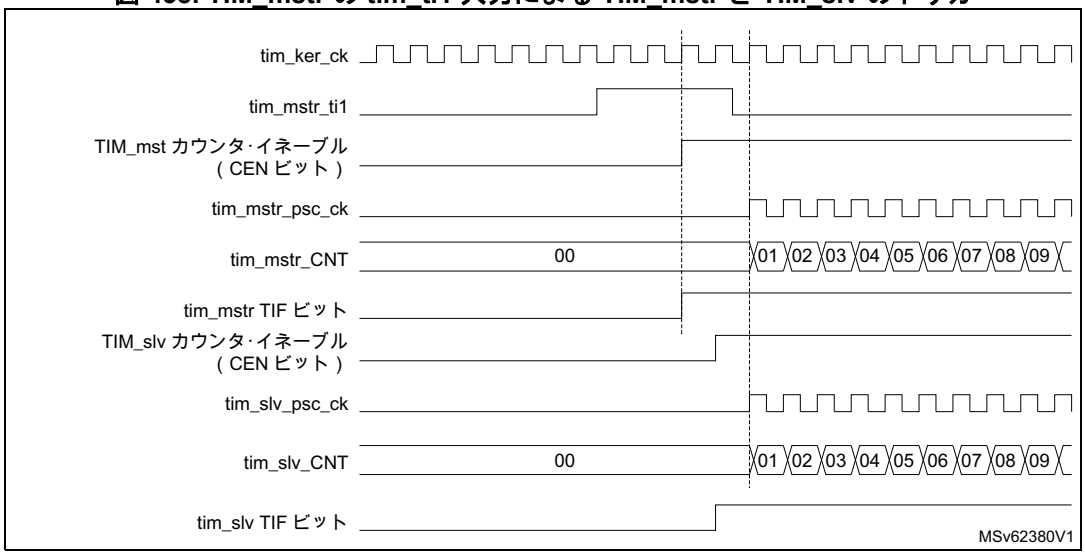
この例では、tim_ti1 入力の立ち上がりエッジで TIM_mstr を有効にし、TIM_mstr が有効になると、TIM_slv を有効にします。接続については、図 432 を参照してください。カウンタの同時性を確保するため、TIM_mstr はマスタ／スレーブモードに設定する必要があります（tim_ti1 に対してはスレーブ、TIM_slv に対してはマスタ）。

1. TIM_mstr をマスタモードに設定して、その有効化をトリガ出力として送信します（TIM_mstr_CR2 レジスタの MMS=001）。
2. TIM_mstr をスレーブモードに設定して、tim_ti1 から入力トリガを受け取るようにします（TIM_mstr_SMCR レジスタの TS=00100）。
3. TIM_mstr をトリガモードに設定します（TIM_mstr_SMCR レジスタの SMS=110）。
4. MSM=1（TIM_mstr_SMCR レジスタ）を書き込むことによって、TIM_mstr をマスタ/スレーブモードに設定します。
5. TIM_slv が TIM_mstr から入力トリガを受け取るように設定します（TIM_slv_SMCR レジスタの TS=00000）。
6. TIM_slv をトリガモードに設定します（TIM_slv_SMCR レジスタの SMS=110）。

tim_ti1（TIM_mstr）で立ち上がりエッジが発生すると、両方のカウンタが同期して内部クロックによるカウントを開始し、両方の TIF フラグがセットされます。

注： **この例では、両方のタイマが開始前に初期化されます（それぞれの UG ビットをセットすることによって）。両方のカウンタは 0 からスタートしますが、カウンタレジスタ（TIMx_CNT）のいずれかに書き込むことによって、両者間に容易にオフセットを挿入できます。マスタ／スレーブモードでは、TIM_mstr の CNT_EN と CK_PSC の間に遅延が挿入されます。**

図 438. TIM_mstr の tim_ti1 入力による TIM_mstr と TIM_slv のトリガ



注： **tim_trgo 信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

29.4.24 ADC トリガ

タイマでは、リセット·イベント、イネーブル·イベント、比較イベントなどのさまざまな内部信号で ADC トリガ·イベントを生成できます。

注： **tim_trgo 信号を受信するスレーブ·ペリフェラル（タイマ、ADC など）のクロックは、マスタ·タイマからイベントを受信する前に有効化する必要があり、マスタ·タイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

29.4.25 DMA バースト·モード

TIMx タイマには、1 つのイベントで多重 DMA リクエストを生成する機能があります。主な目的は、タイマの一部をソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   – DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   – DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   – 転送データ数 = 3 とします（下の注を参照）。
   – サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
   DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

29.4.26 TIM2/TIM3/TIM4/TIM5DMA リクエスト

表 282 に示すように、TIM2/TIM3/TIM4/TIM5 は DMA リクエストを生成できます。

表 282. DMA リクエスト

DMA リクエスト信号	DMA 略称	DMA リクエスト	イネーブル制御ビット
tim_upd_dma	TIM_UP	更新	UDE
tim_cc1_dma	TIM_CH1	キャプチャ／比較 1	CC1DE
tim_cc2_dma	TIM_CH2	キャプチャ／比較 2	CC2DE
tim_cc3_dma	TIM_CH3	キャプチャ／比較 3	CC3DE
tim_cc4_dma	TIM_CH4	キャプチャ／比較 4	CC4DE
tim_trg_dma	TIM_TRIG	トリガ	TDE

注： **タイマによっては、DMA リクエストが DMA コントローラに接続されないことがあります。詳細については、DMA のセクションを参照してください。**

29.4.27 デバッグ･モード

マイクロコントローラがデバッグモードになると（FPU 搭載 Cortex®-M4 コアは停止状態）、TIMx カウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

29.4.28 TIM2/TIM3/TIM4/TIM5 低電力モード

表 283. 低消費電力モードが TIM2/TIM3/TIM4/TIM5 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。 割込みによって、デバイスは SLEEP モードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

29.4.29 TIM2/TIM3/TIM4/TIM5 割込み

表 284 に示すように、TIM2/TIM3/TIM4/TIM5 は複数の割込みを生成できます。

表 284. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
TIM_UP	更新	UIF	UIE	UIF に 0 を書き込む	可能	不可
TIM_CC	キャプチャ／比較 1	CC1IF	CC1IE	CC1IF に 0 を書き込む	可能	不可
	キャプチャ／比較 2	CC2IF	CC2IE	CC2IF に 0 を書き込む	可能	不可
	キャプチャ／比較 3	CC3IF	CC3IE	CC3IF に 0 を書き込む	可能	不可
	キャプチャ／比較 4	CC4IF	CC4IE	CC4IF に 0 を書き込む	可能	不可
TIM_TRG	トリガ	TIF	TIE	TIF に 0 を書き込む	可能	不可
TIM_DIR_IDX	インデックス	IDXF	IDXIE	IDXF に 0 を書き込む	可能	不可
	方向	DIRF	DIRIE	DIRF に 0 を書き込む	可能	不可
TIM_IERR	インデックスエラー	IERRF	IERRIE	IERRF に 0 を書き込む	可能	不可
TIM_TER	遷移エラー	TERRF	TERRIE	TERRF に 0 を書き込む	可能	不可

29.5 TIM2/TIM3/TIM4/TIM5 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

ペリフェラル･レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

29.5.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 2～5）

アドレス･オフセット：0x000

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DITHEN**：ディザリング･イネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHEN ビットは、CEN ビットがリセットされているときのみ変更できます。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デジタルフィルタ（tim_etr_in、tim_tix）によって使用されるサンプリングクロックとの間の分周比を示します。

00：$t_{DTS} = t_{tim_ker_ck}$

01：$t_{DTS} = 2 \times t_{tim_ker_ck}$

10：$t_{DTS} = 4 \times t_{tim_ker_ck}$

11：予約済みです。

ビット 7 **ARPE**：自動再ロード･プリロード･イネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモードカウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。

01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントダウンしているときのみセットされます。

10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップしているときのみセットされます。

11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS=00）の出力比較割込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN=1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

注： このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読出し専用です。

ビット 3 **OPM**：ワンパルス･モード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエスト･ソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディスエーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタ･イネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

29.5.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 2～5）

アドレス･オフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
								rw	rw	rw	rw	rw			

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 24:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TI1S**：tim_ti1 選択

0：tim_ti1_in[15:0] マルチプレクサ出力が tim_ti1 入力に接続されます。
1：tim_ti1_in[15:0]、tim_ti2_in[15:0]、および tim_ti3_in[15:0] マルチプレクサ出力が XOR 演算されて tim_ti1 入力に接続されます。セクション 28.3.29：1136 ページのホールセンサとのインタフェース も参照してください。

ビット 25、6、5、4 **MMS[3:0]**：マスタ･モード選択

これらのビットにより､同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（tim_trgo）。組み合わせは、次のとおりです。

0000：**リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo 信号は実際のリセットより遅延します。

0001：**イネーブル** - カウンタ･イネーブル信号 CNT_EN がトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（AND）によって生成されます。
カウンタ･イネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブ･モードが選択されている場合を除き、tim_trgo には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照してください）。

0010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

0011：**パルス比較** - CC1IF フラグがすでにハイであっても、それがセットされるキャプチャまたは比較一致が発生すると、トリガ出力として正のパルスがただちに送信されます（tim_trgo）。

0100：**比較** - tim_oc1refc 信号がトリガ出力（tim_trgo）として使用されます。

0101：**比較** - tim_oc2refc 信号がトリガ出力（tim_trgo）として使用されます。

0110：**比較** - tim_oc3refc 信号がトリガ出力（tim_trgo）として使用されます。

0111：**比較** - tim_oc4refc 信号がトリガ出力（tim_trgo）として使用されます。

1000：**エンコーダ･クロック出力** - エンコーダ･クロック信号がトリガ出力（tim_trgo）として使用されます。このコードは次の SMS[3:0] 値に対して有効です。0001、0010、0011、1010、1011、1100、1101、1110、1111。その他の SMS[3:0] コードは許されず、予想外の動作につながることがあります。

その他：予約済みです。

注： スレーブ･タイマまたは ADC のクロックは、マスタ･タイマからイベントを受信する前に有効化する必要があり、マスタ･タイマからトリガを受信している間は動作中に変更しないでください。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。
1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

29.5.3 TIMx スレーブモード制御レジスタ（TIMx_SMCR）（x = 2～5）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS[4:3]		Res.	Res.	Res.	SMS[3]
						rw	rw			rw	rw				rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			OCCS	SMS[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **SMSPS**：SMS プリロード･ソース

このビットは、プリロードからアクティブへの SMS[3:0] ビットフィールドの転送をトリガするイベントのいずれかを選択します。

0：転送は、タイマの更新イベントによってトリガされます。

1：転送は、インデックス･イベントによってトリガされます。

ビット 24 **SMSPE**：SMS プリロード･イネーブル

このビットは、SMS[3:0] ビットフィールドをプリロードするかどうかを選択します。

0：SMS[3:0] ビットフィールドはプリロードされません。

1：SMS[3:0] プリロードは有効です。

ビット 23:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **ETP**：外部トリガ極性

このビットは、tim_etr_in と $\overline{\text{tim_etr_in}}$ のいずれがトリガ動作に使用されるかを選択します。

0：tim_etr_in は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。

1：tim_etr_in は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE**：外部クロック･イネーブル

このビットは、外部クロックモード 2 を有効にします。

0：外部クロックモード 2 は無効です。

1：外部クロックモード 2 は有効です。カウンタは、tim_etrf 信号のアクティブエッジによってクロック供給されます。

注： ECE ビットをセットすることは、tim_trgi が tim_etrf に接続された状態で外部クロック･モード 1 を選択することと同じ効果があります（SMS = 111、TS = 00111）。
外部クロック･モード 2 とスレーブ モードであるリセット･モード、ゲート･モード、またはトリガ･モードを同時に使用できます。ただし、この場合、tim_trgi を tim_etrf に接続することはできません（TS ビットが 00111 でないことが必要）。
外部クロック･モード 1 と外部クロック･モード 2 が同時に有効な場合、外部クロック入力は tim_etrf です。

ビット 13:12 **ETPS[1:0]**：外部トリガプリスケーラ

外部トリガ信号 tim_etrp の周波数は、tim_ker_ck 周波数の 1/4 までに制限されます。プリスケーラを有効にすると、tim_etrp 周波数を低減できます。これは、高速な外部クロックを tim_etr_in に入力するときに役立ちます。

00：プリスケーラオフ

01：tim_etrp 周波数は 2 分周されます。

10：tim_etrp 周波数は 4 分周されます。

11：tim_etrp 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]**：外部トリガフィルタ

このビットフィールドは、tim_etrp 信号をサンプルする周波数と、tim_etrp に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001： $f_{SAMPLING} = f_{tim_ker_ck}$、N = 2
0010： $f_{SAMPLING} = f_{tim_ker_ck}$、N = 4
0011： $f_{SAMPLING} = f_{tim_ker_ck}$、N = 8
0100： $f_{SAMPLING} = f_{DTS}/2$、N = 6
0101： $f_{SAMPLING} = f_{DTS}/2$、N = 8
0110： $f_{SAMPLING} = f_{DTS}/4$、N = 6
0111： $f_{SAMPLING} = f_{DTS}/4$、N = 8
1000： $f_{SAMPLING} = f_{DTS}/8$、N = 6
1001： $f_{SAMPLING} = f_{DTS}/8$、N = 8
1010： $f_{SAMPLING} = f_{DTS}/16$、N = 5
1011： $f_{SAMPLING} = f_{DTS}/16$、N = 6
1100： $f_{SAMPLING} = f_{DTS}/16$、N = 8
1101： $f_{SAMPLING} = f_{DTS}/32$、N = 5
1110： $f_{SAMPLING} = f_{DTS}/32$、N = 6
1111： $f_{SAMPLING} = f_{DTS}/32$、N = 8

ビット 7 **MSM**：マスタ／スレーブ･モード

0：影響なし。
1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgo を通じて）完全な同期状態が得られるまで遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（tim_itr0）
00001：内部トリガ 1（tim_itr1）
00010：内部トリガ 2（tim_itr2）
00011：内部トリガ 3（tim_itr3）
00100：tim_ti1 エッジ検出回路（tim_ti1f_ed）
00101：フィルタ･タイマ入力 1（tim_ti1fp1）
00110：フィルタ･タイマ入力 2（tim_ti2fp2）
00111：外部トリガ入力（tim_etrf）
01000：内部トリガ 4（tim_itr4）
01001：内部トリガ 5（tim_itr5）
01010：内部トリガ 6（tim_itr6）
01011：内部トリガ 7（tim_itr7）
01100：内部トリガ 8（tim_itr8）
01101：内部トリガ 9（tim_itr9）
01110：内部トリガ 10（tim_itr10）
01111：内部トリガ 11（tim_itr11）
10000：内部トリガ 12（tim_itr12）
10001：内部トリガ 13（tim_itr13）
10010：内部トリガ 14（tim_itr14）
10011：内部トリガ 15（tim_itr15）
その他：予約済みです。

製品固有の実装の詳細は セクション 29.4.2 : TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 **OCCS**：OCREF クリア選択

このビットは、OCREF クリアソースを選択するために使用されます。

0：tim_ocref_clr_int は tim_ocref_clr 入力に接続されます。

1：tim_ocref_clr_int は tim_etrf に接続されます。

注： OCREF クリア選択機能がサポートされていない場合、このビットは予約済みであり、ハードウェアによって強制的に "0" に設定されます。セクション 29.3：TIM2/TIM3/TIM4/TIM5 の実装を参照してください。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：エンコーダモード 1 - カウンタは、tim_ti2fp2 のレベルに応じて、tim_ti1fp1 のエッジでカウントアップ／ダウンします。

0010：エンコーダモード 2 - カウンタは、tim_ti1fp1 のレベルに応じて、tim_ti2fp2 のエッジでカウントアップ／ダウンします。

0011：エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、tim_ti1fp1 と tim_ti2fp2 の両方のエッジでカウントアップ/ダウンします。

0100：リセット･モード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲート･モード - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ･モード - カウンタは、トリガ tim_trgi の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロック･モード 1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセット･モードとトリガ･モードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲート･モードとリセット･モードの組み合わせ - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

1010：エンコーダ･モード：クロックプラス方向、x2 モード。

1011：エンコーダ･モード：クロックプラス方向、x1 モード。tim_ti2fp2 エッジ検出は CC2P によって設定されます。

1100：エンコーダ･モード：方向クロック、x2 モード。

1101：エンコーダ･モード：方向クロック、x1 モード。tim_ti1fp1 と tim_ti2fp2 エッジ検出は CC1P と CC2P によって設定されます。

1110：直交エンコーダ･モード：x1 モード、tim_ti1fp1 エッジのみでカウント、エッジ検出は CC1P によって設定されます。

1111：直交エンコーダ･モード：x1 モード、tim_ti2fp2 エッジのみでカウント、エッジ検出は CC2P によって設定されます。

注： トリガ入力として tim_ti1f_ed が選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_ed は tim_ti1f の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgo 信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

29.5.4 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 2～5）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.
								rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TERRIE**：遷移エラー割込みイネーブル
0：遷移エラー割込みは無効です。
1：遷移エラー割込みは有効です。

ビット 22 **IERRIE**：インデックス･エラー割込みイネーブル
0：インデックス･エラー割込みは無効です。
1：インデックス･エラー割込みは有効です。

ビット 21 **DIRIE**：方向変更割込みイネーブル
0：方向変更割込みは無効です。
1：方向変更割込みは有効です。

ビット 20 **IDXIE**：インデックス割込みイネーブル
0：インデックス割込みは無効です。
1：インデックス割込みは有効です。

ビット 19:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエスト･イネーブル
0：トリガ DMA リクエストは無効です。
1：トリガ DMA リクエストは有効です。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4DE**：キャプチャ／比較 4 DMA リクエストイネーブル
0：CC4 DMA リクエストは無効です。
1：CC4 DMA リクエストは有効です。

ビット 11 **CC3DE**：キャプチャ／比較 3 DMA リクエストイネーブル
0：CC3 DMA リクエストは無効です。
1：CC3 DMA リクエストは有効です。

ビット 10 **CC2DE**：キャプチャ／比較 2 DMA リクエストイネーブル
0：CC2 DMA リクエストは無効です。
1：CC2 DMA リクエストは有効です。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル
0：CC1 DMA リクエストは無効です。
1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエスト･イネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE**：トリガ割込みイネーブル

0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IE**：キャプチャ／比較 4 割込みイネーブル

0：CC4 割込みは無効です。
1：CC4 割込みは有効です。

ビット 3 **CC3IE**：キャプチャ／比較 3 割込みイネーブル

0：CC3 割込みは無効です。
1：CC3 割込みは有効です。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル

0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル

0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。
1：更新割込みは有効です。

29.5.5 TIMx ステータスレジスタ（TIMx_SR）（x = 2〜5）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	Res.	Res.
								rc_w0	rc_w0	rc_w0	rc_w0				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TERRF**：遷移エラー割込みフラグ

このフラグは、エンコーダモードで遷移エラーが検出されたときに、ハードウェアによってセットされます。
これは、ソフトウェアで“0”を書き込むことによってクリアされます。
0：エンコーダ遷移エラーは検出されていません。
1：エンコーダ遷移エラーが検出されています。

ビット 22 **IERRF**：インデックス･エラー割込みフラグ

このフラグは、インデックスエラーが検出されたときに、ハードウェアによってセットされます。ソフトウェアによって
“0”を書き込むことによってクリアされます。
0：インデックス･エラーは検出されていません。
1：インデックス･エラーが検出されています。

ビット 21 **DIRF**：方向変更割込みフラグ

このフラグは、エンコーダモードで方向が変わった（TIMx_CR の DIR ビット値が変わった）ときに、ハードウェアによってセットされます。

これは、ソフトウェアで“0”を書き込むことによってクリアされます。

0：方向の変更なし

1：方向の変更

ビット 20 **IDXF**：インデックス割込みフラグ

このフラグは、インデックスイベントが検出されたときに、ハードウェアによってセットされます。ソフトウェアによって

“0”を書き込むことによってクリアされます。

0：インデックス･イベントは発生していません。

1：インデックス･イベントが発生しています。

ビット 19:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **CC4OF**：キャプチャ／比較 4 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 11 **CC3OF**：キャプチャ／比較 3 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバーキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIF**：トリガ割込みフラグ

このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4IF**：キャプチャ／比較 4 割込みフラグ

CC1IF の説明を参照してください。

ビット 3 **CC3IF**：キャプチャ／比較 3 割込みフラグ

CC1IF の説明を参照してください。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタ･オーバーフロー時（アップカウント･モードおよびアップ／ダウンカウント･モードの場合）、またはアンダーフロー時（ダウンカウント･モードの場合）に CC1IF ビットはハイになります。センターアライン･モードでのフラグの設定には 3 つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

オーバーフローまたはアンダーフロー時と、TIMx_CR1 レジスタで UDIS=0 の場合。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

29.5.6 TIMx イベント生成レジスタ（TIMx_EGR）（x = 2〜5）

アドレスオフセット：0x014

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **CC4G**：キャプチャ／比較 4 生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。センターアラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとります。

29.5.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 2～5）

アドレス･オフセット：0x018

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。

10：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。

11：CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、tim_ti1 入力をサンプリングする周波数と、tim_ti1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E=0（TIMx_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

10：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。

11：CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

29.5.8 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 2～5）

アドレス･オフセット：0x018

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード
ビット 6:4 の OC1M 説明を参照

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択
このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC2 チャネルは出力として設定されます。
01：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。
10：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。
11：CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC2E=0）のときにのみ書込み可能です。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル
0：tim_oc1ref は tim_ocref_clr_int 入力の影響を受けません。
1：tim_oc1ref は tim_ocref_clr_int 入力のハイレベルが検出されるとクリアされます。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、tim_oc1 が導き出される出力基準信号 tim_oc1ref の動作を定義します。tim_oc1ref はアクティブハイですが、tim_oc1 のアクティブレベルは CC1P ビットに依存します。

0000 : 停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません（このモードはタイミングベースを生成するために使用されます）。

0001：比較が一致した場合、チャネル 1 をアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にハイになります。

0010 : 比較が一致した場合、チャネル 1 をインアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、tim_oc1ref は反転します。

0100：強制非アクティブレベル - tim_oc1ref は強制的にローになります。

0101：強制アクティブレベル - tim_oc1ref は強制的にハイになります。

0110：PWM モード 1 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はインアクティブ（tim_oc1ref=0）に、そうでない場合はアクティブ（tim_oc1ref=1）になります。

0111：PWM モード 2 - カウントアップ時、チャネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャネル 1 は、TIMx_CNT> TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済み。

1011：予約済み。

1100：組み合わせ PWM モード 1 - tim_oc1ref は、PWM モード 1 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理和（OR）です。

1101：組み合わせ PWM モード 2 - tim_oc1ref は、PWM モード 2 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理積（AND）です。

1110 : 非対称 PWM モード 1 - tim_oc1ref は、PWM モード 1 と同様に動作します。tim_oc1refc は、カウンタがカウント･アップするときに tim_oc1ref を出力し、カウント･ダウンするときに tim_oc2ref を出力します。

1111 : 非対称 PWM モード 2 - tim_oc1ref は、PWM モード 2 と同様に動作します。tim_oc1refc は、カウンタがカウント･アップするときに tim_oc1ref を出力し、カウント･ダウンするときに tim_oc2ref を出力します。

注： PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、tim_ocref_clr のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。
1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。
0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。
1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC1 チャネルは出力として設定されます。
01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。
10：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。
11：CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

29.5.9 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 2〜5）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4F [3:0]				IC4PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3PSC [1:0]		CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC4F[3:0]**：入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC[1:0]**：入力キャプチャ 4 プリスケーラ

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC4 チャネルは出力として設定されます。

01：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti4 に配置されます。

10：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti3 に配置されます。

11：CC4 チャネルは入力として設定され、tim_ic4 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7:4 **IC3F[3:0]**：入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC[1:0]**：入力キャプチャ 3 プリスケーラ

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC3 チャネルは出力として設定されます。

01：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti3 に配置されます。

10：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti4 に配置されます。

11：CC3 チャネルは入力として設定され、tim_ic3 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

29.5.10 TIMx キャプチャ／比較モードレジスタ 2 [オルタネート]（TIMx_CCMR2）（x = 2〜5）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC4CE**：出力比較 4 クリアイネーブル

ビット 24、14:12 **OC4M[3:0]**：出力比較 4 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照

ビット 11 **OC4PE**：出力比較 4 プリロードイネーブル

ビット 10 **OC4FE**：出力比較 4 高速イネーブル

ビット 9:8 **CC4S[1:0]**：キャプチャ／比較 4 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャネルは出力として設定されます。
01：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti4 に配置されます。
10：CC4 チャネルは入力として設定され、tim_ic4 は tim_ti3 に配置されます。
11：CC4 チャネルは入力として設定され、tim_ic4 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC4S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書込み可能です。

ビット 7 **OC3CE**：出力比較 3 クリアイネーブル

ビット 16、6:4 **OC3M[3:0]**：出力比較 3 モード

OC1M の説明（TIMx_CCMR1 レジスタのビット 6:4）を参照してください。

ビット 3 **OC3PE**：出力比較 3 プリロードイネーブル

ビット 2 **OC3FE**：出力比較 3 高速イネーブル

ビット 1:0 **CC3S[1:0]**：キャプチャ／比較 3 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャネルは出力として設定されます。
01：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti3 に配置されます。
10：CC3 チャネルは入力として設定され、tim_ic3 は tim_ti4 に配置されます。
11：CC3 チャネルは入力として設定され、tim_ic3 は tim_trc にマッピングされます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC3S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書込み可能です。

29.5.11 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 2〜5）

アドレスオフセット：0x020

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

ビット 15 **CC4NP**：キャプチャ／比較 4 出力極性

CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **CC4P**：キャプチャ／比較 4 出力極性

CC1P の説明を参照してください。

ビット 12 **CC4E**：キャプチャ／比較 4 出力イネーブル。

CC1E の説明を参照してください。

ビット 11 **CC3NP**：キャプチャ／比較 3 出力極性

CC1NP の説明を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC3P**：キャプチャ／比較 3 出力極性

CC1P の説明を参照してください。

ビット 8 **CC3E**：キャプチャ／比較 3 出力イネーブル。
CC1E の説明を参照してください。

ビット 7 **CC2NP**：キャプチャ／比較 2 出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル。
CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 出力極性
CC1 チャネルが出力として設定されている場合：この場合、CC1NP はクリアされたままでなければなりません。
CC1 チャネルが入力として設定されている場合：このビットは、tim_ti1fp1/tim_ti2fp1 の極性を定義するために CC1P と組み合わせて使用されます（CC1P の説明を参照してください）。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性
0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）
1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）
CC1 チャネルを入力として設定している場合は、CC1NP/CC1P の両ビットで、トリガまたはキャプチャ動作における TI1FP1 と TI2FP1 のアクティブ極性を選択します。
CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。
CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。
CC1NP=1、CC1P=1： 非反転/両エッジ。この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。
CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル。
0：キャプチャモードは無効／OC1 は非アクティブ。
1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

表 285. 標準 tim_ocx チャネルの出力制御ビット

CCxE ビット	tim_ocx 出力状態
0	出力無効（タイマによって駆動されない：ハイインピーダンス）
1	出力は有効（tim_ocx = tim_ocxref + 極性）

注： CCxE=0 のとき、標準 tim_ocx チャネルに接続されている外部 IO ピンの状態は、GPIO レジスタのみに依存します。

29.5.12 TIMx カウンタ（TIMx_CNT）（x = 3、4）

アドレスオフセット：0x024

リセット値：0x0000 0000

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

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

ビット 31 **UIFCPY**：値は TIMx_CR1 の IUFREMAP に依存します。

UIFREMAP = 0 の場合：

予約済みです。

UIFREMAP = 1 の場合：

UIFCPY：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分を保持します。分数部分は得られません。

29.5.13 TIMx カウンタ（TIMx_CNT）（x = 2、5）

アドレスオフセット：0x024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY_ CNT [31]	CNT[30:16]														
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

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

ビット 31 **UIFCPY_CNT[31]**：値は TIMx_CR1 の IUFREMAP に依存します。

UIFREMAP = 0 の場合：

CNT[31]：カウンタ値の最上位ビット

UIFREMAP = 1 の場合：

UIFCPY：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー

ビット 30:0 **CNT[30:0]**：カウンタ値の最下位部分

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[30:0] に非ディザリング部分を保持します。分数部分は得られません。

29.5.14 TIMx プリスケーラ（TIMx_PSC）（x = 2〜5）

アドレスオフセット：0x028

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 ftim_cnt_ck は $f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

29.5.15 TIMx 自動再ロード･レジスタ（TIMx_ARR）（x = 3、4）

アドレスオフセット：0x02C

リセット値：0x0000 FFFF

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **ARR[19:0]**：自動再ロード値下位ビット

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 29.4.3：1205 ページのタイムベース･ユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、自動再ロード値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

29.5.16 TIMx 自動再ロード･レジスタ（TIMx_ARR）（x = 2、5）

アドレスオフセット：0x02C

リセット値：0xFFFF FFFF

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

ビット 31:0 **ARR[31:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 29.4.3 : 1205 ページのタイムベース･ユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、自動再ロード値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[31:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

29.5.17 TIMx キャプチャ / 比較レジスタ 1（TIMx_CCR1）（x = 3、4）

アドレス･オフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR1[19:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc1 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1 チャネルが入力として設定されている場合：

CCR1 は､最後の入力キャプチャ 1 イベント（tim_ic1）によって転送されたカウンタ値です｡TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

CCR1[15:0] ビットはキャプチャ値を保持します。CCR1[19:16] ビットは予約済みです。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:0] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

29.5.18 TIMx キャプチャ / 比較レジスタ 1（TIMx_CCR1）（x = 2、5）

アドレス･オフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR1 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CCR1[31:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc1 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[31:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1 チャネルが入力として設定されている場合：

CCR1 は､最後の入力キャプチャ 1 イベント（tim_ic1）によって転送されたカウンタ値です｡TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[31:0] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

29.5.19 TIMx キャプチャ / 比較レジスタ 2（TIMx_CCR2）（x = 3、4）

アドレス･オフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR2[19:0]**：キャプチャ／比較 1 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc2 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR2[15:0] に比較値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2 チャネルが入力として設定されている場合：

CCR2 は､最後の入力キャプチャ 2 イベント（tim_ic2）によって転送されたカウンタ値です｡TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

CCR2[15:0] ビットはキャプチャ値を保持します。CCR2[19:16] ビットは予約済みです。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:0] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

29.5.20 TIMx キャプチャ / 比較レジスタ 2（TIMx_CCR2）（x = 2、5）

アドレス･オフセット：0x038

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR2 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CCR2[31:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc2 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[31:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2 チャネルが入力として設定されている場合：

CCR2 は､最後の入力キャプチャ 2 イベント（tim_ic2）によって転送されたカウンタ値です｡TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[31:0] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

29.5.21 TIMx キャプチャ / 比較レジスタ 3（TIMx_CCR3）（x = 3、4）

アドレス･オフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR3[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR3[19:0]**：キャプチャ／比較 3 値

CC3 チャネルが出力として設定されている場合：

CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。TIMx_CCMR3 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc3 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR3[15:0] に比較値を保持します。CCR3[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:4] に整数部を保持します。CCR3[3:0] ビットフィールドにはディザリング部分が含まれます。

CC3 チャネルが入力として設定されている場合：

CCR3 は､最後の入力キャプチャ 3 イベント（tim_ic3）によって転送されたカウンタ値です｡TIMx_CCR3 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

CCR3[15:0] ビットはキャプチャ値を保持します。CCR3[19:16] ビットは予約済みです。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[19:0] にキャプチャを保持します。CCR3[3:0] ビットはリセットされています。

29.5.22 TIMx キャプチャ / 比較レジスタ 3（TIMx_CCR3）（x = 2、5）

アドレス･オフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR3 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CCR3[31:0]**：キャプチャ／比較 3 値

CC3 チャネルが出力として設定されている場合：

CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。TIMx_CCMR3 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc3 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[31:4] に整数部を保持します。CCR3[3:0] ビットフィールドにはディザリング部分が含まれます。

CC3 チャネルが入力として設定されている場合：

CCR3 は､最後の入力キャプチャ 3 イベント（tim_ic3）によって転送されたカウンタ値です｡TIMx_CCR3 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR3[31:0] にキャプチャを保持します。CCR3[3:0] ビットはリセットされています。

29.5.23 TIMx キャプチャ / 比較レジスタ 4（TIMx_CCR4）（x = 3、4）

アドレス･オフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR4[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR4[19:0]**：キャプチャ／比較 4 値

CC4 チャネルが出力として設定されている場合：

CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。TIMx_CCMR4 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc4 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR4[15:0] に比較値を保持します。CCR4[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:4] に整数部を保持します。CCR4[3:0] ビットフィールドにはディザリング部分が含まれます。

CC4 チャネルが入力として設定されている場合：

CCR4 は､最後の入力キャプチャ 4 イベント（tim_ic4）によって転送されたカウンタ値です｡TIMx_CCR4 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

CCR4[15:0] ビットはキャプチャ値を保持します。CCR4[19:16] ビットは予約済みです。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[19:0] にキャプチャを保持します。CCR4[3:0] ビットはリセットされています。

29.5.24 TIMx キャプチャ / 比較レジスタ 4（TIMx_CCR4）（x = 2、5）

アドレス･オフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CCR4 [31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **CCR4[31:0]**：キャプチャ／比較 4 値

CC4 チャネルが出力として設定されている場合：

CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。TIMx_CCMR4 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc4 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、比較値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[31:4] に整数部を保持します。CCR4[3:0] ビットフィールドにはディザリング部分が含まれます。

CC4 チャネルが入力として設定されている場合：

CCR4 は､最後の入力キャプチャ 4 イベント（tim_ic4）によって転送されたカウンタ値です｡TIMx_CCR4 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、キャプチャ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR4[31:0] にキャプチャを保持します。CCR4[3:0] ビットはリセットされています。

29.5.25 TIMx タイマエンコーダ制御レジスタ（TIMx_ECR）（x = 2〜5）

アドレスオフセット：0x058

リセット値：0x0000 0000

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

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **PWPRSC[2:0]**：パルス幅プリスケーラ

このビットフィールドは、パルス発生回路用のクロックプリスケーラを次のように設定します。

$t_{PWG} = (2^{(PWPRSC[2:0])}) \times t_{tim_ker_ck}$

ビット 23:16 **PW[7:0]**：パルス幅

このビットフィールドは、パルスの長さを次のように定義します。

$t_{PW} = PW[7:0] \times t_{PWG}$

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **IPOS[1:0]**：インデックス位置決め

直交エンコーダモード（SMS[3:0] = 0001、0010、0011、1110、1111）で、このビットは、インデックスイベントがカウンタをリセットする AB 入力の設定を示します。

00：AB = 00 のときにインデックスがカウンタをリセットします。
01：AB = 01 のときにインデックスがカウンタをリセットします。
10：AB = 10 のときにインデックスがカウンタをリセットします。
11：AB = 11 のときにインデックスがカウンタをリセットします。

方向クロックモードまたはクロックプラス方向モード（SMS[3:0] = 1010、1011、1100、1101）で、これらのビットは、インデックスイベントがカウンタをリセットするレベルを示します。双方向クロックモードでは、これは両方のクロック入力に適用されます。

x0：クロックが 0 のときにインデックスがカウンタをリセットします。
x1：クロックが 1 のときにインデックスがカウンタをリセットします。

注： IPOS[1] ビットは意味がありません。

ビット 5 **FIDX**：最初のインデックス

このビットは、最初のインデックスのみが考慮されるかどうかを示します。

0：インデックスは常にアクティブです。
1：最初のインデックスのみがカウンタをリセットします。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:1 **IDIR[1:0]**：インデックスの方向

このビットは、どちらの方向のときにインデックスイベントがカウンタをリセットするかを示します。

00：方向に関係なく、インデックスがカウンタをリセットします。
01：アップカウントのときのみ、インデックスがカウンタをリセットします。
10：ダウンカウントのときのみ、インデックスがカウンタをリセットします。
11：予約済みです。

注： IDR[1:0] ビットフィールドは、IE ビットがリセット状態のとき（インデクス無効時）に書き込む必要があります。

ビット 0 IEインデックス･イネーブル

このビットは、インデックスイベントがカウンタをリセットするかどうかを示します。

0：インデックスは無効です。

1：インデックスは有効です。

29.5.26 TIMx タイマ入力選択レジスタ（TIMx_TISEL）（x = 2〜5）

アドレスオフセット：0x05C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **TI4SEL[3:0]**：tim_ti4[15:0] 入力を選択します。

0000：tim_ti4_in0：TIMx_CH4

0001：tim_ti4_in1

.....

1111：tim_ti4_in15

製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TI3SEL[3:0]**：tim_ti3[15:0] 入力を選択します。

0000：tim_ti3_in0：TIMx_CH3

0001：tim_ti3_in1

.....

1111：tim_ti3_in15

製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：tim_ti2[15:0] 入力を選択します。

0000：tim_ti2_in0：TIMx_CH2

0001：tim_ti2_in1

.....

1111：tim_ti2_in15

製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：tim_ti1[15:0] 入力を選択します。
0000：tim_ti1_in0：TIMx_CH1
0001：tim_ti1_in1
.....
1111：tim_ti1_in15
製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

29.5.27 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 2〜5）

アドレス･オフセット：0x060

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL[3:2]	
														rw	rw

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

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:14 **ETRSEL[3:0]**：etr_in ソース選択
これらのビットは、etr_in 入力ソースを選択します。
0000：tim_etr0：TIMx_ETR 入力
0001：tim_etr1
.....
1111：tim_etr15
製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

ビット 13:0 予約済みであり、リセット値に保持する必要があります。

29.5.28 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 2～5）

アドレス･オフセット：0x064

リセット値：0x0000 0000

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

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **OCRSEL[2:0]**：ocref_clr ソース選択

これらのビットは、ocref_clr 入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

.....

111：tim_ocref_clr7

製品固有の実装についてはセクション 29.4.2：TIM2/TIM3/TIM4/TIM5 ピンおよび内部信号を参照してください。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

29.5.29 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 2～5）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

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

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、転送数など DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送
00010：3 回転送
.....
11010：26 回転送

例：次の転送を考えます：DBL = 7 バイトかつ DBA = TIM2_CR1。

– DBL = 7 バイトおよび DBA = TIM2_CR1 が転送するバイトのアドレスを表す場合、転送のアドレスは次の式で与えられます。

（TIMx_CR1 アドレス）+ DBA +（DMA インデックス）、ここで DMA インデックス = DBL

この例では、（TIMx_CR1 アドレス）+ DBA に 7 バイトが加算され、データのコピー元とコピー先のアドレスが得られます。この場合、転送は、以下のアドレスを先頭とした 7 つのレジスタに対して行われます。（TIMx_CR1 アドレス）+ DBA

DMA データサイズの設定に応じて、いくつかのケースが想定されます。

– DMA データ･サイズをハーフワードで設定した場合、7 つのレジスタにそれぞれ 16 ビットのデータが転送されます。

– DMA データ･サイズをバイトで設定した場合も、データは 7 つのレジスタに転送されます（最初のレジスタには最初の MSB バイトが含まれ、2 番目のレジスタには最初の LSB バイトが含まれるなど、以降同様）。したがって、タイマへの転送では、DMA で転送するデータサイズも指定する必要があります。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し/書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

29.5.30 完全転送の TIMx DMA アドレス（TIMx_DMAR）（x = 2〜5）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

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

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

29.5.31 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマップされます。

表 286. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD [1:0]		ARPE	CMS [1:0]		DIR	OPM	URS	UDIS	CEN
	リセット値																				0	0		0	0	0	0	0	0	0	0	0	0
0x004	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	MMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1S	MMS[2:0]			CCDS	Res.	Res.	Res.
	リセット値							0																		0	0	0	0	0			
0x008	**TIMx_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	SMSPS	SMSPE	Res.	Res.	TS [4:3]		Res.	Res.	Res.	SMS[3]	ETP	ECE	ETPS[1:0]		ETF[3:0]				MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値							0	0			0	0				0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0
0x00C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRIE	IERRIE	DIRIE	IDXIE	Res.	Res.	Res.	Res.	Res.	Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	リセット値								0	0	0	0							0		0	0	0	0	0		0		0	0	0	0	0
0x010	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TERRF	IERRF	DIRF	IDXF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4OF	CC3OF	CC2OF	CC1OF	Res.	Res.	TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
	リセット値								0	0	0	0									0	0	0	0			0		0	0	0	0	0
0x014	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
	リセット値																										0		0	0	0	0	0
0x018	**TIMx_CCMR1** 入力キャプチャ・モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2 PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	**TIMx_CCMR2** 入力キャプチャ・モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC4F [3:0]				IC4 PSC [1:0]		CC4S [1:0]		IC3F [3:0]				IC3 PSC [1:0]		CC3S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIMx_CCMR2 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC4M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC3M[3]	O24CE	OC4M [2:0]			OC4PE	OC4FE	CC4S [1:0]		OC3CE	OC3M [2:0]			OC3PE	OC3FE	CC3S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
	リセット値																	0		0	0	0		0	0	0		0	0	0		0	0
0x024	**TIMx_CNT**	UIFCPY_CNT[31]	CNT[30:16] （32 ビット・タイマでのみ CNT[31:16]）															CNT[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 286. TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x02C	**TIMx_ARR**（x = 3、4）	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x02C	**TIMx_ARR**（x = 2、5）	ARR[31:0]																															
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x030	予約済みです。																																
0x034	**TIMx_CCR1**	CCR1 [31:20]（32 ビット・タイマのみ）												CCR1 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	**TIMx_CCR2**	CCR2 [31:20]（32 ビット・タイマのみ）												CCR2 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	**TIMx_CCR3**	CCR3 [31:20]（32 ビット・タイマのみ）												CCR3 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	**TIMx_CCR4**	CCR4 [31:20]（32 ビット・タイマのみ）												CCR4 [19:0]																			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044.. 0x054	予約済みです。	Res.																															
0x058	**TIMx_ECR**	Res.	Res.	Res.	Res.	Res.	PWPRSC [2:0]			PW[7:0]								Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IPOS [1:0]		FIDX	Res.	Res.	IDIR [1:0]		IE
	リセット値						0	0	0	0	0	0	0	0	0	0	0									0	0	0			0	0	0
0x05C	**TIMx_TISEL**	Res.	Res.	Res.	Res.	TI4SEL [3:0]				Res.	Res.	Res.	Res.	TI3SEL [3:0]				Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値					0	0	0	0					0	0	0	0					0	0	0	0					0	0	0	0
0x060	**TIMx_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ETRSEL [3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値															0	0	0	0														
0x064	**TIMx_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x068.. 0x3D8	予約済みです。	Res.																															
0x3DC	**TIMx_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIMx_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

30 汎用タイマ（TIM15/TIM16/TIM17）

30.1 TIM15/TIM16/TIM17 概要

TIM15/TIM16/TIM17 タイマは、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM、デッドタイムを挿入した相補 PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケーラと RCC クロックコントローラプリスケーラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

TIM15/TIM16/TIM17 タイマは完全に独立していて、いかなるリソースも共有しません。TIM15 は、セクション 30.4.26：タイマ同期（TIM15 のみ）に示すように同期させることができます。

30.2 TIM15 の主な機能

TIM15 には以下の機能があります。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- 次の機能を持つ、最大 2 つの独立チャネル：
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力（チャネル 1 のみ）
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー、カウンタの初期化（ソフトウェアまたは内部／外部トリガによる）
  - トリガイベント（カウンタの開始、停止、初期化、または内部／外部トリガによるカウント）
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力（割込みリクエスト）

30.3 TIM16/TIM17 の主な特徴

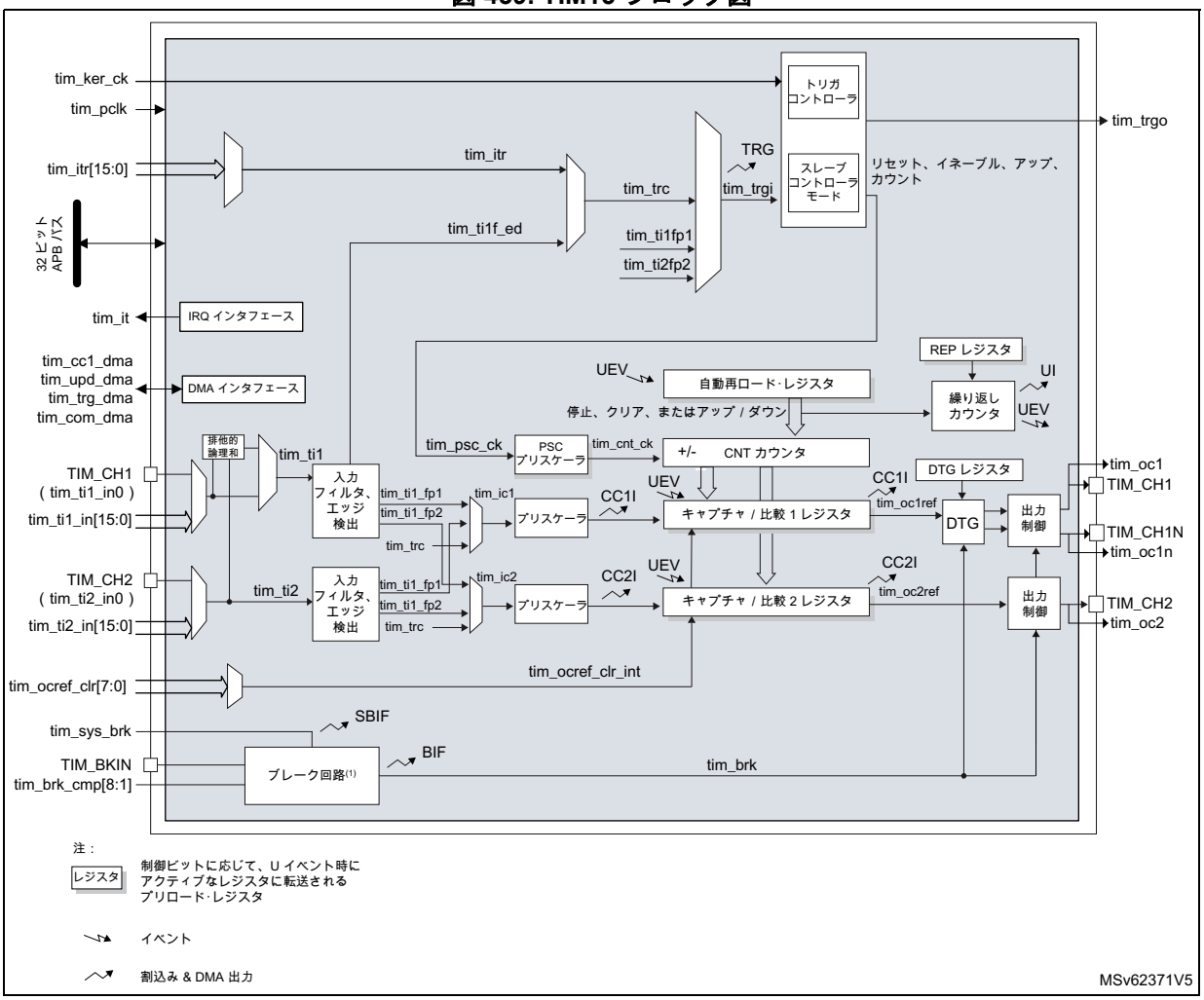
TIM16/TIM17 タイマには以下の機能があります。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- 次の機能を持つ、1 チャネルタイマ。
  - 入力キャプチャ
  - 出力比較
  - PWM 生成（エッジアラインモード）
  - ワンパルスモード出力
- プログラム可能なデッドタイムを持つ相補出力
- カウンタの特定のサイクル数後にのみタイマレジスタを更新する繰り返しカウンタ。
- タイマの出力信号をリセット状態または既知の状態にするブレーク入力。
- 以下のイベント時の割込み／DMA 生成：
  - 更新：カウンタオーバーフロー
  - 入力キャプチャ
  - 出力比較
  - ブレーク入力

30.4 TIM15/TIM16/TIM17機能詳細

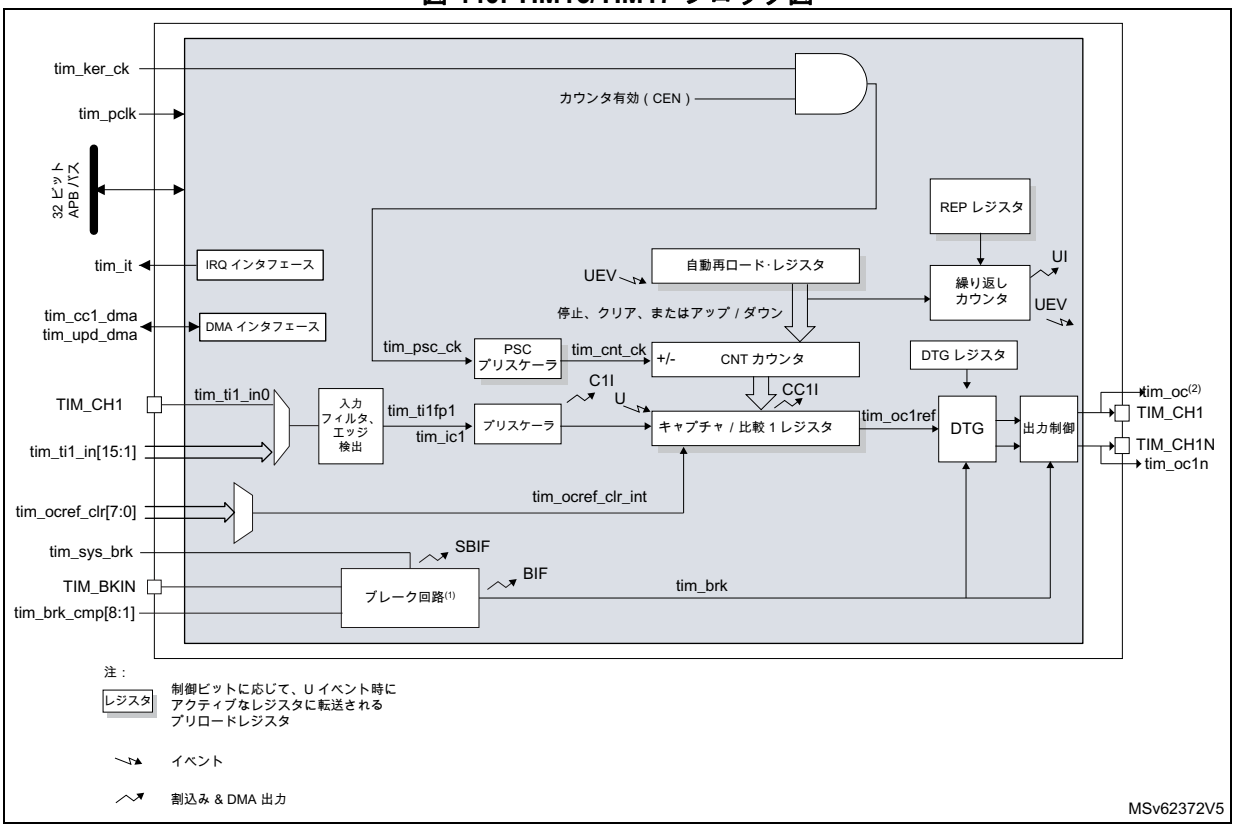
30.4.1 ブロック図

図 439. TIM15 ブロック図



1. 詳細については、セクション 30.4.15：ブレーク機能の使用を参照してください。

図 440. TIM16/TIM17 ブロック図



1. 詳細については、セクション 30.4.15：ブレーク機能の使用を参照してください。
2. この信号は、いくつかのスレーブ·タイマのトリガとして使用できます（次のセクションの内部トリガ接続表を参照してください）。詳細については、セクション 30.4.27：タイマ出力を他のタイマのトリガとして使用する（TIM16/TIM17 のみ）を参照してください。

30.4.2 TIM15/TIM16/TIM17 のピンと内部信号

このセクションの表 287 および表 288 は、TIM の入力および出力の一覧です。

表 287. TIM の入出力ピン

ピン名	信号タイプ	説明
TIM_CH1 TIM_CH2(1)	入力／出力	タイマ多目的チャネル。 各チャネルは、キャプチャ、比較、または PWM で使用できます。 TIM_CH1 と TIM_CH2 は、外部クロック（周波数は tim_ker_ck クロックの 1/4 未満）、外部トリガ入力としても使用できます。
TIM_CH1N	出力	デッドタイムを挿入できる、TIM_CH1 出力から得られるタイマ相補出力。
TIM_BKIN	入力／出力	ブレーク入力。この入力は双方向モードにも設定することができます。

1. TIM15 でのみ使用できます。

表 288. TIM 内部入力／出力信号

内部信号名	信号タイプ	説明
tim_ti1_in [15:0] tim_ti2_in [15:0](1)	入力	内部タイマ入力バス。これらの入力は、キャプチャまたは外部クロック（周波数は tim_ker_ck クロックの 1/4 未満）として使用できます。
tim_itr[15:0] (1)	入力	内部トリガ入力バス。これらの入力は、スレーブモードコントローラ用に、または入力クロック（tim_ker_ck クロックの 1/4 未満）として使用できます。
tim_trgo (1)	出力	内部トリガ出力。このトリガは他のオンチップペリフェラルをトリガできます。
tim_ocref_clr[7:0]	入力	タイマの tim_ocref_clr 入力バス。これらの入力は、通常ハードウェアのサイクルごとのパルス幅制御のために、tim_ocxref 信号をクリアするために使用できます。
tim_brk_cmp[8:1]	入力	内部信号のブレーク入力
tim_sys_brk[n:0]	入力	システムブレーク入力この入力は MCU のシステムレベルのエラーを集めます。
tim_pclk	入力	タイマ APB クロック
tim_ker_ck	入力	タイマカーネルクロック。このクロックは、tim_pclk（同じソースから導出）と同期する必要があります。クロック比 tim_ker_ck/tim_pclk は整数 1、2、3、・・・16（最大値）でなければなりません。
tim_it	出力	キャプチャ／比較、更新、ブレークトリガ、および転流リクエストを取りまとめるグローバルタイマ割込み。
tim_cc1_dma	出力	タイマ･キャプチャ／比較 1 DMA リクエスト
tim_upd_dma	出力	タイマ更新 DMA リクエスト
tim_trg_dma	出力	タイマトリガ DMA リクエスト
tim_com_dma	出力	タイマ転流 DMA リクエスト

1. TIM15 でのみ使用できます。

表 289 および表 290 に、tim_ti[2:1] 入力マルチプレクサに接続するソースを示します。

表 289. tim_ti1 入力マルチプレクサへの相互接続

tim_ti1 入力	ソース		
	TIM15	TIM16	TIM17
tim_ti1_in0	TIM15_CH1	TIM16_CH1	TIM17_CH1
tim_ti1_in1	LSE	comp6_out	comp5_out
tim_ti1_in2	comp1_out	MCO	MCO
tim_ti1_in3	comp2_out	HSE/32(1)	HSE/32 (1)
tim_ti1_in4	comp5_out	RTC クロック	RTC クロック
tim_ti1_in5	comp7_out	LSE	LSE
tim_ti1_in6	予約済みです。	LSI	LSI
tim_ti1_in [15:7]	予約済みです。		

1. この信号は、TIMx_OR1 レジスタの HSE32EN ビットがセットされている場合のみ、使用できます。詳細については、セクション 30.8.19 : TIMx オプション･レジスタ 1（TIMx_OR1）（x = 16～17）を参照してください。

表 290. tim_ti2 入力マルチプレクサへの相互接続

tim_ti2 入力	ソース
	TIM15
tim_ti2_in0	TIM15_CH2
tim_ti2_in1	comp2_out
tim_ti2_in2	comp3_out
tim_ti2_in3	comp6_out
tim_ti2_in4	comp7_out
tim_ti2_in [15:5]	予約済みです。

表 291 に、tim_itr 入力マルチプレクサに接続される内部ソースを列挙します。

表 291. TIMx 内部トリガ接続

tim_itrx 入力	TIM15
tim_itr0	tim1_trgo
tim_itr1	tim2_trgo
tim_itr2	tim3_trgo
tim_itr3	tim4_trgo
tim_itr4	tim5_trgo
tim_itr5	tim8_trgo
tim_itr6	予約済みです。
tim_itr7	tim16_oc1
tim_itr8	tim17_oc1
tim_itr9	tim20_trgo
tim_itr10	hrtim_out_sync2
tim_itr[15:11]	予約済みです。

表 292 および 表 293 に、tim_brk 入力に接続するソースを示します。

表 292. タイマブレーク相互接続

tim_brk 入力	TIM15	TIM16	TIM17
TIM_BKIN	TIM15_BKIN ピン	TIM16_BKIN ピン	TIM17_BKIN ピン
tim_brk_cmp1	comp1_out	comp1_out	comp1_out
tim_brk_cmp2	comp2_out	comp2_out	comp2_out
tim_brk_cmp3	comp3_out	comp3_out	comp3_out
tim_brk_cmp4	comp4_out	comp4_out	comp4_out
tim_brk_cmp5	comp5_out	comp5_out	comp5_out
tim_brk_cmp6	comp6_out	comp6_out	comp6_out

表 292. タイマブレーク相互接続 （続き）

tim_brk 入力	TIM15	TIM16	TIM17
tim_brk_cmp7	comp7_out	comp7_out	comp7_out
tim_brk_cmp8	予約済みです。		

表 293. システム･ブレーク相互接続

tim_sys_brk 入力	TIM15 / TIM16 / TIM17	SYSCFG_CFGR2 レジスタの イネーブルビット
tim_sys_brk0	FPU 搭載 Cortex®-M4 ロックアップ	CLL
tim_sys_brk1	プログラム可能な電圧検出器（PVD）	PVDL
tim_sys_brk2	SRAM パリティ･エラー	SPL
tim_sys_brk3	Flash ダブル ECC エラー	ECCL
tim_sys_brk4	クロック･セキュリティ･システム（CSS）	なし（常に有効）

表 294 に、tim_ocref_clr 入力マルチプレクサに接続される内部ソースを列挙します。

表 294. ocref_clr 入力マルチプレクサへの相互接続

タイマ OCREF クリア信号	タイマ OCREF クリア信号割り当て		
	TIM15	TIM16	TIM17
tim_ocref_clr0	comp1_out	comp1_out	comp1_out
tim_ocref_clr1	comp2_out	comp2_out	comp2_out
tim_ocref_clr2	comp3_out	comp3_out	comp3_out
tim_ocref_clr3	comp4_out	comp4_out	comp4_out
tim_ocref_clr4	comp5_out	comp5_out	comp5_out
tim_ocref_clr5	comp6_out	comp6_out	comp6_out
tim_ocref_clr6	comp7_out	comp7_out	comp7_out
tim_ocref_clr7	予約済みです。		

30.4.3 タイムベース･ユニット

プログラマブル高機能制御タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）
- 繰り返しカウンタレジスタ（TIMx_RCR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプ

リロードイネーブルビット（ARPE）に応じて、常時または更新イベント（UEV）ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときにのみ有効な、プリスケーラ出力 tim_cnt_ck から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

TIM x _CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 441 と 図 442 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 441. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

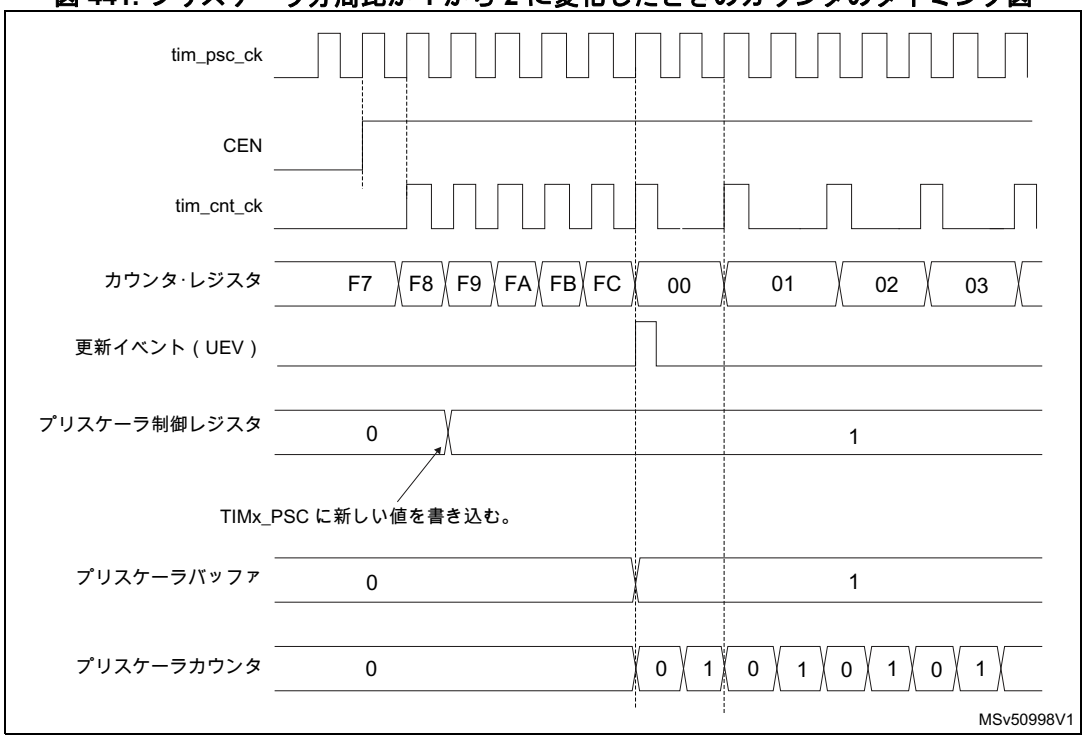
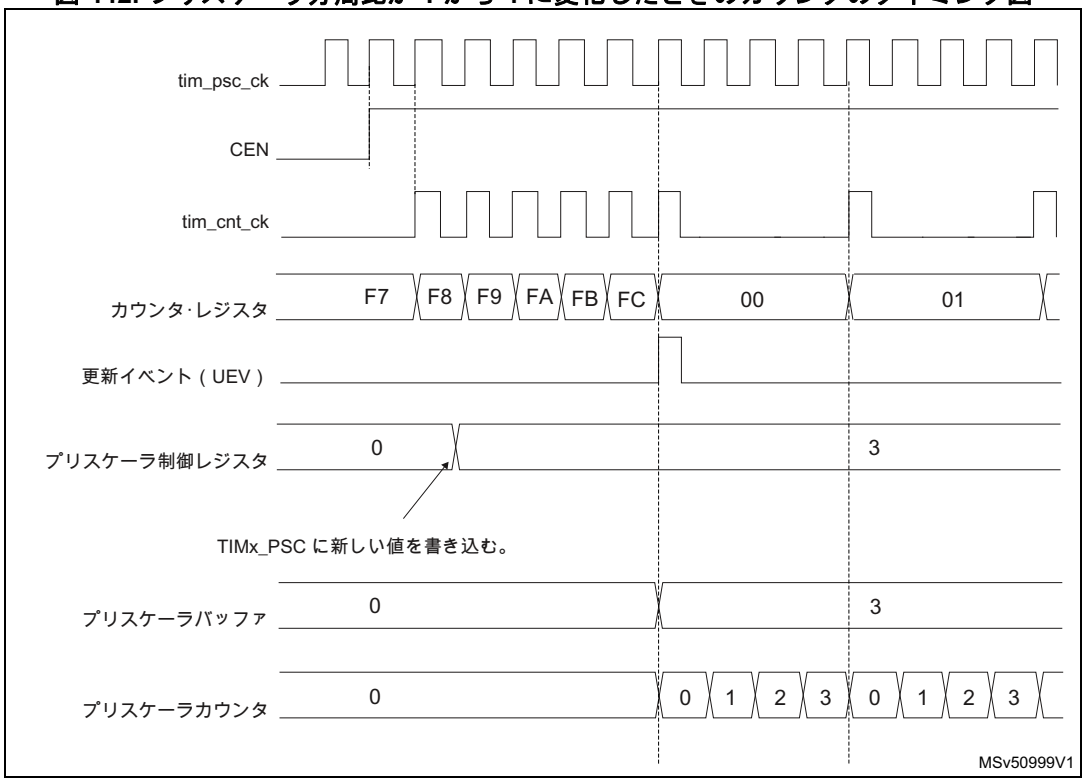


図 442. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



30.4.4 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

繰り返しカウンタが使用されている場合には、繰り返しカウンタレジスタにプログラムされている回数（TIMx_RCR）までアップカウント動作が繰り返され、その後に更新イベント（UEV）が生成されます。繰り返しカウンタが使用されていないときには、カウンタのオーバーフローごとに更新イベントが生成されます。

（ソフトウェアによって、またはスレーブモードコントローラを使用して）TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケーラのカウンタと同じく、カウンタは 0 からカウントをリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS ビット（更新リクエスト選択）がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（したがって、割込みや DMA リクエストは送信されません）。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- 繰り返しカウンタには TIMx_RCR レジスタの内容が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。
- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再びロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 443. 内部クロック分周比が 1 の場合のカウンタのタイミング図

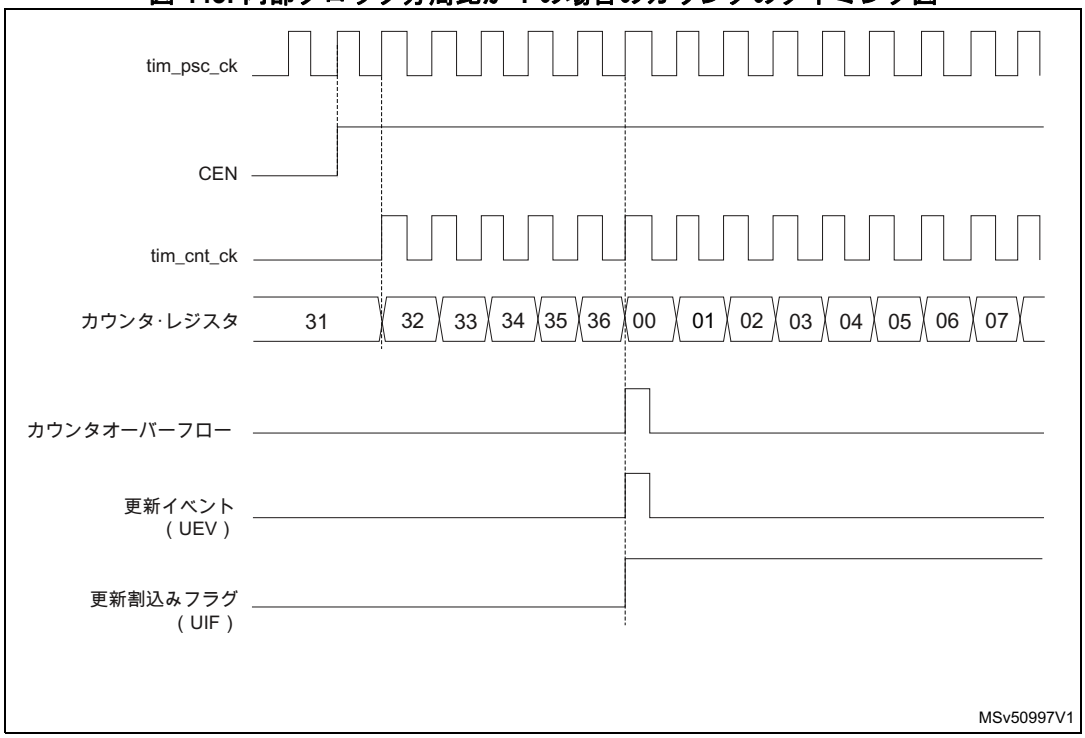


図 444. 内部クロック分周比が 2 の場合のカウンタのタイミング図

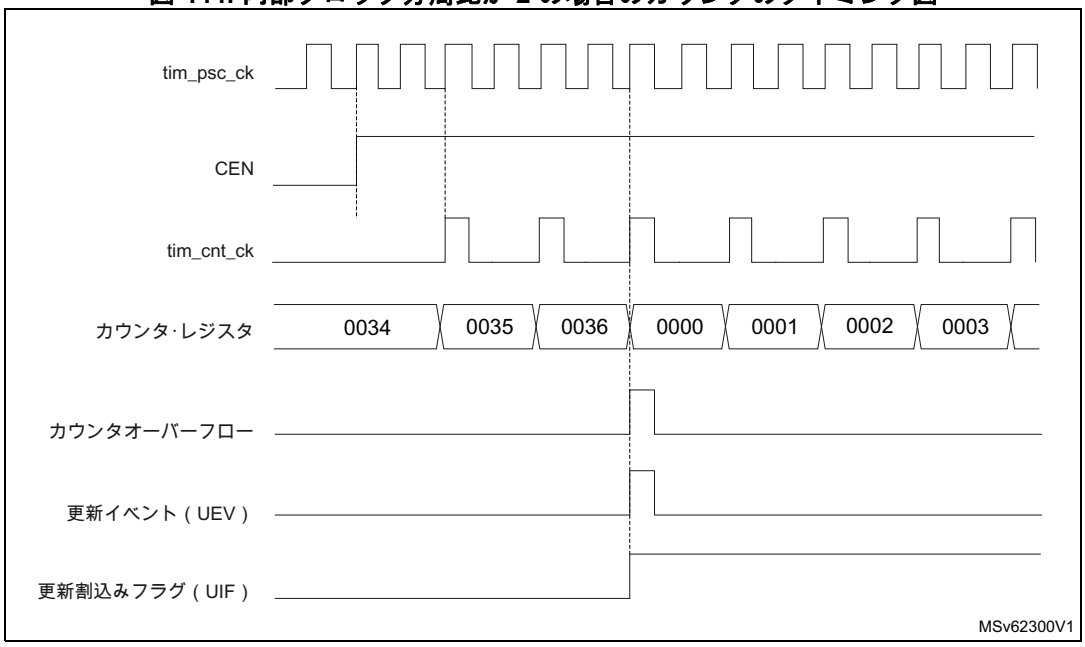


図 445. 内部クロック分周比が 4 の場合のカウンタのタイミング図

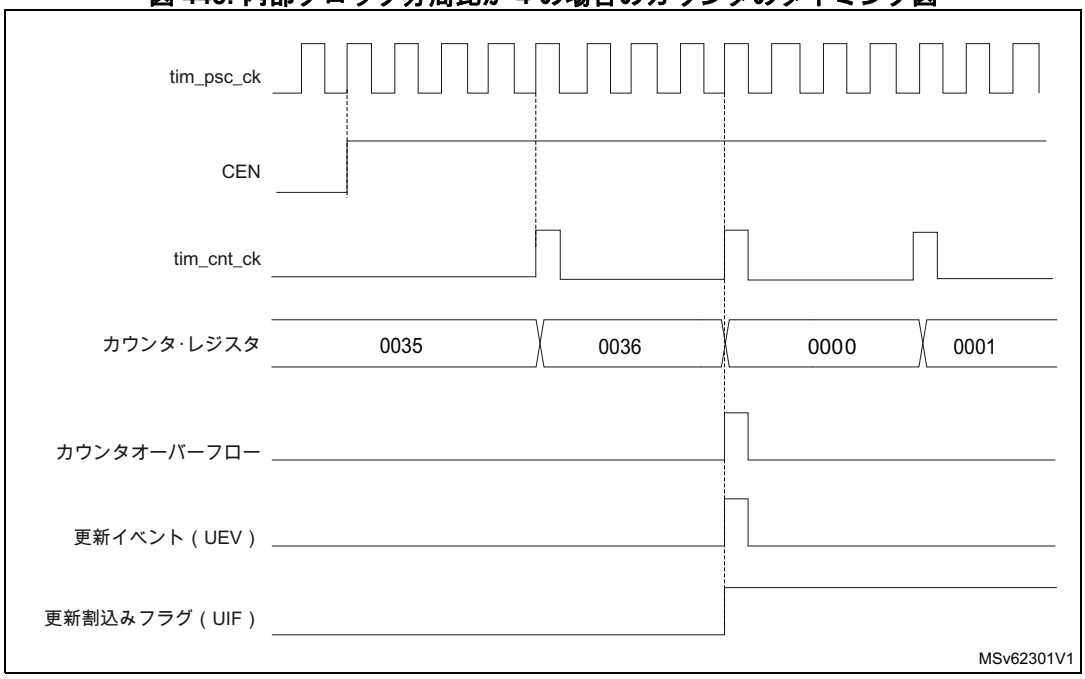


図 446. 内部クロック分周比が N の場合のカウンタのタイミング図

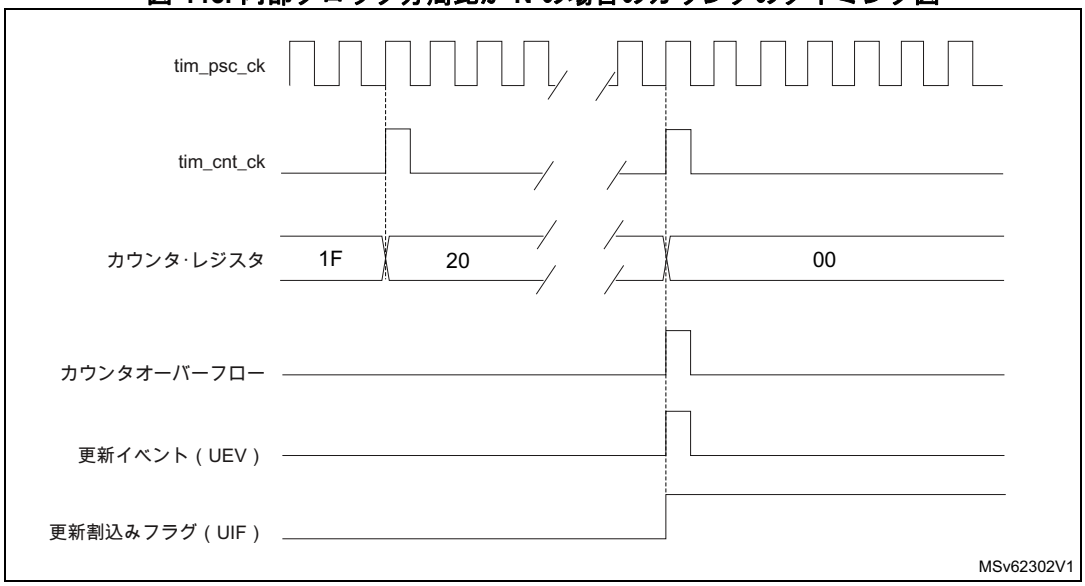


図 447. ARPE=0の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロードされない）

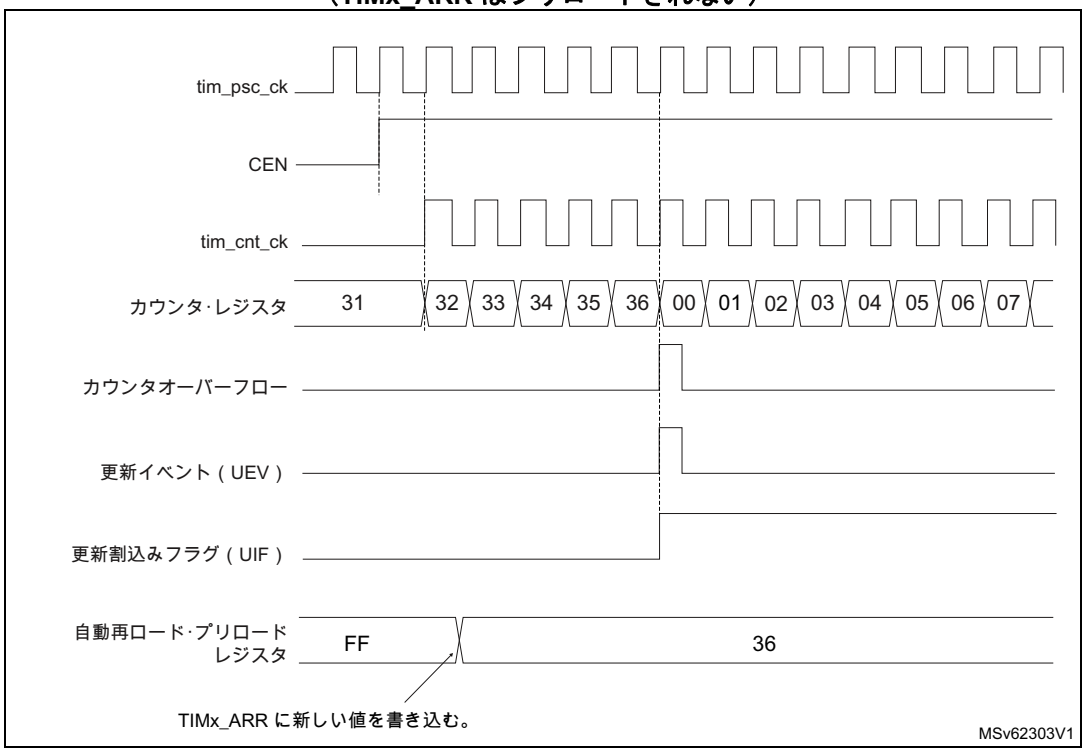
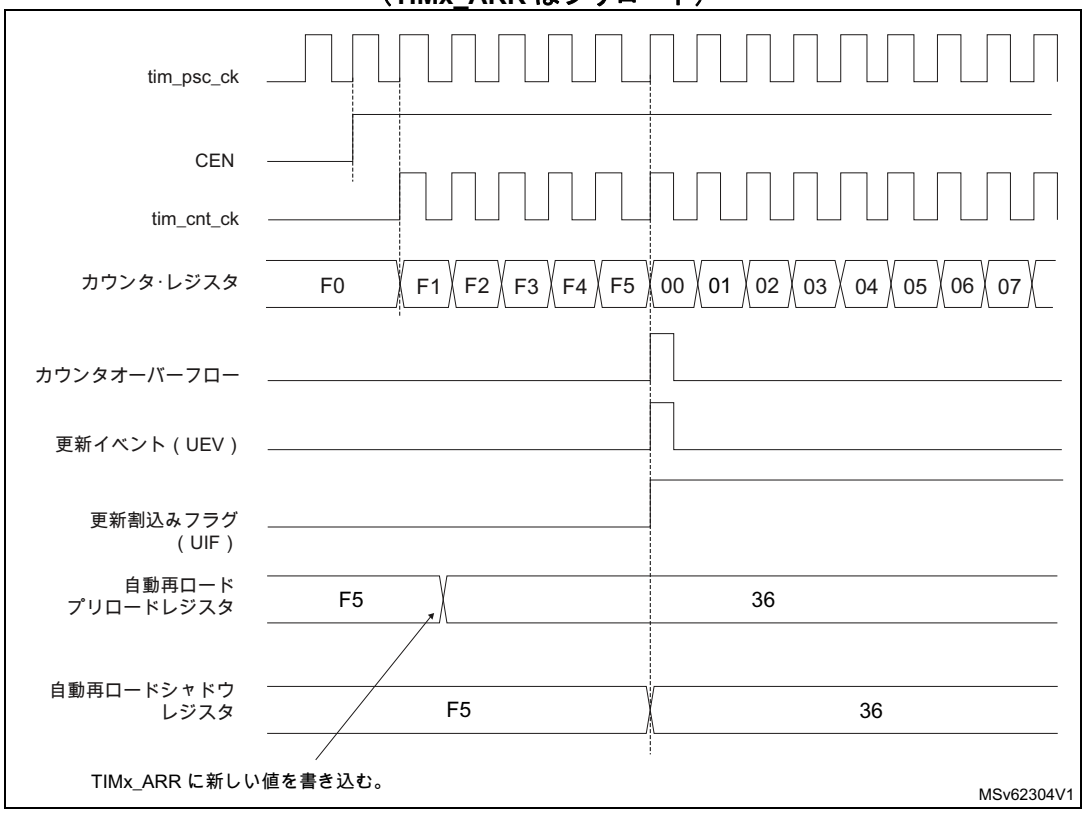


図 448. ARPE=1の場合の更新イベント時のカウンタのタイミング図（TIMx_ARR はプリロード）



30.4.5 繰り返しカウンタ

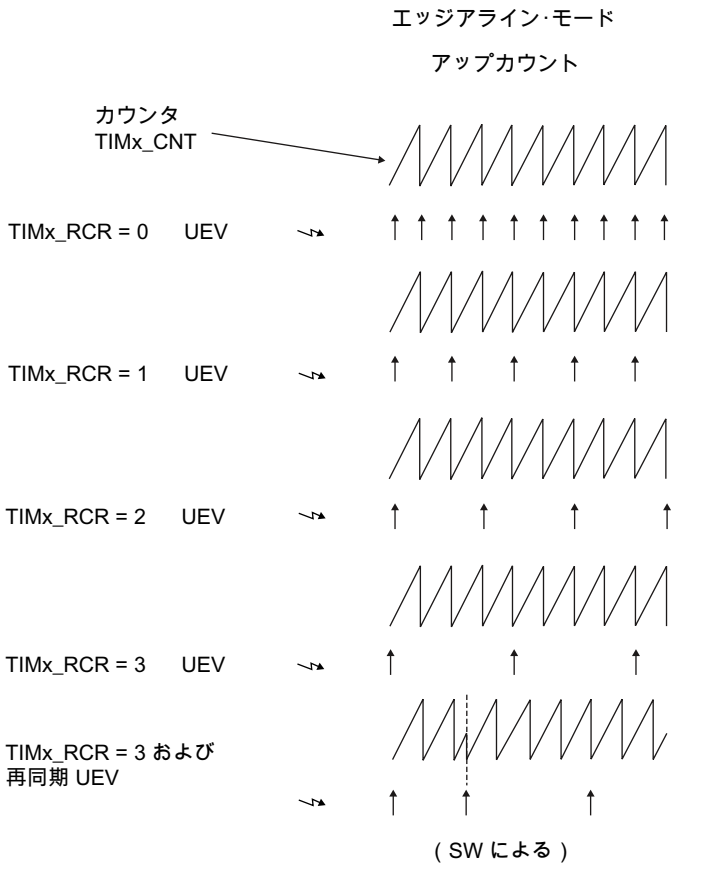
セクション 30.4.3：タイムベース·ユニットに、カウンタオーバーフローによって、どのように更新イベント（UEV）が生成されるかが説明されています。実際には、繰り返しカウンタが0 に達したときにのみ、更新イベントが生成されます。これは、PWM 信号を生成する際に役立ちます。

これは、TIMx_RCR 繰り返しカウンタレジスタの値を N とすると、N 回目のカウンタオーバーフローごとに、プリロードレジスタからシャドウレジスタ（TIMx_ARR 自動再ロードレジスタ、TIMx_PSC プリスケーラレジスタ、比較モードの TIMx_CCRx キャプチャ /比較レジスタ）へデータが転送されることを意味します。

繰り返しカウンタは、カウンタオーバーフローごとにデクリメントされます。

繰り返しカウンタは自動再ロードタイプです。繰り返しの回数は、TIMx_RCR レジスタの値によって定義されたとおりに維持されます（図 449 を参照してください）。ソフトウェアによって（TIMx_EGR レジスタの UG ビットをセットすることによって）、またはスレーブモードコントローラを介してハードウェアによって更新イベントが生成されると、繰り返しカウンタの値にかかわらず直ちにイベントが発生し、繰り返しカウンタに TIMx_RCR レジスタの内容が再ロードされます。

図 449. モードと TIMx_RCR レジスタの設定に応じた更新レートの例



30.4.6 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

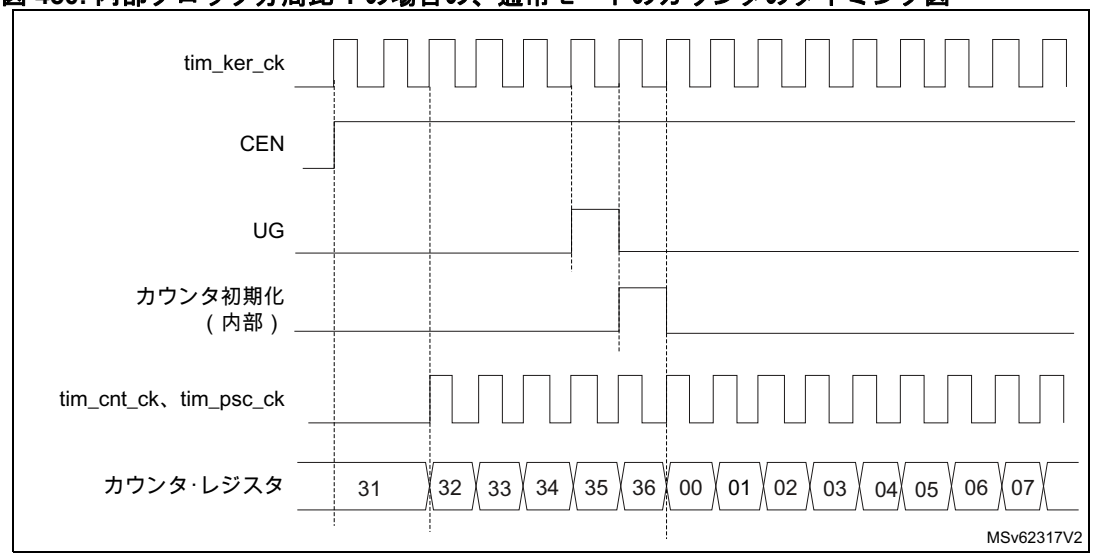
- 内部クロック（tim_ker_ck）
- 外部クロックモード 1：外部入力ピン（使用可能な場合、tim_ti1 または tim_ti2）
- 内部トリガ入力（tim_itrx）（TIM15 の場合のみ）：あるタイマを別のタイマのプリスケーラとして使用します。たとえば、TIM1 が TIM15 のプリスケーラとして機能するように設定できます。詳細については、タイマを使用して別のタイマを有効にするを参照してください。

内部クロックソース（tim_ker_ck）

スレーブモードコントローラが無効の場合（SMS=000）、CEN（TIMx_CR1 レジスタ）、および UG ビット（TIMx_EGR レジスタ）が実際の制御ビットとなり、ソフトウェアによってのみ変更できます（自動的にクリア状態に保たれる UG ビットを除きます）。CEN ビットに 1 が書き込まれるとすぐに、プリスケーラには内部クロック tim_ker_ck が供給されます。

図 450 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

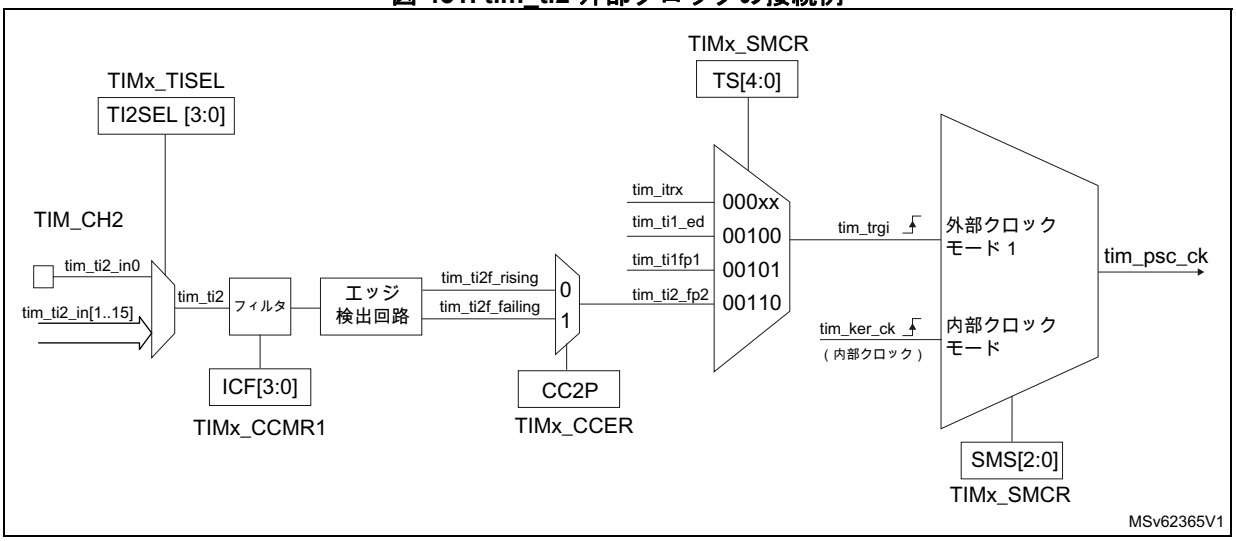
図 450. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 451. tim_ti2 外部クロックの接続例



たとえば、tim_ti2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

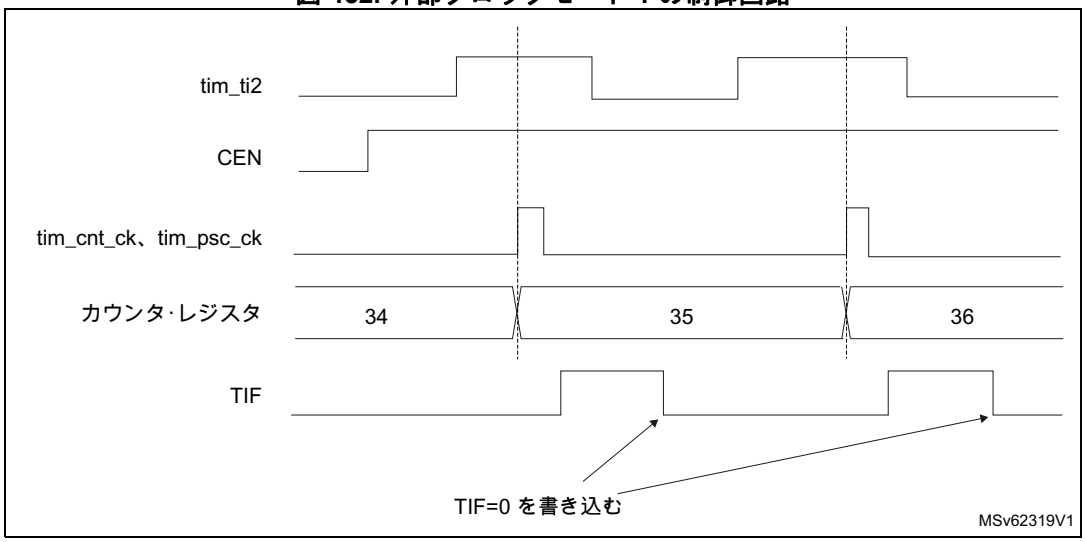
1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な tim_ti2_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャネル 2 が tim_ti2 入力の立ち上がりエッジを検出するように設定します。
3. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。
4. CC2P=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
5. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
6. TIMx_SMCR レジスタに TS=00110 を書き込むことによって、トリガ入力ソースとして tim_ti2 を選択します。
7. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

注： **キャプチャプリスケーラはトリガには使用されないので、設定する必要はありません。**

tim_ti2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際のカウンタクロックまでの間には、tim_ti2 入力の再同期回路による遅延があります。

図 452. 外部クロックモード 1 の制御回路



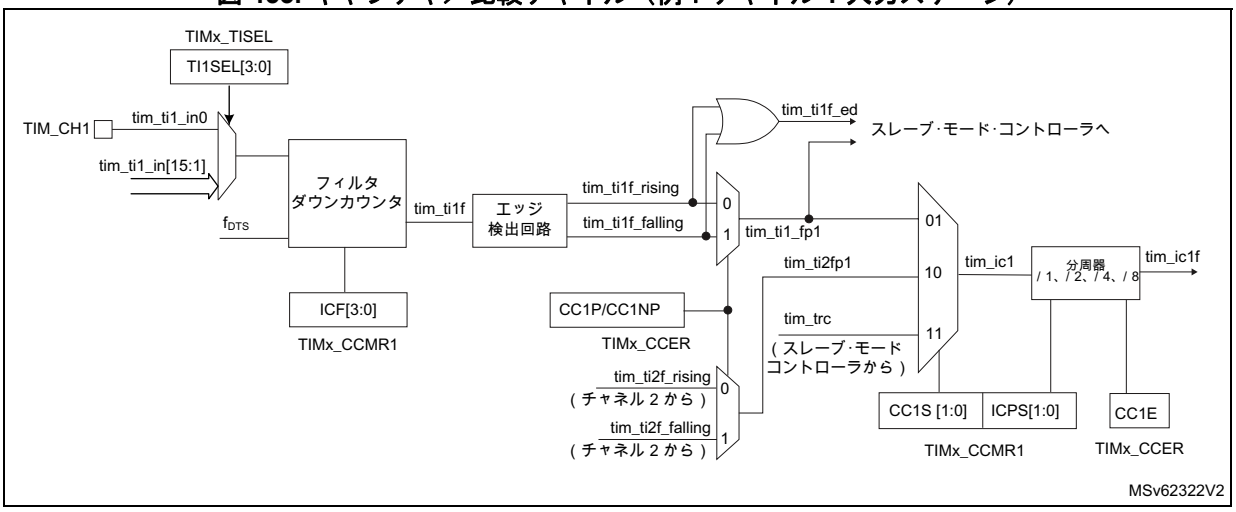
30.4.7 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入力ステージ（デジタルフィルタ、マルチプレクス、プリスケーラ）、および出力ステージ（コンパレータと出力制御）から構成されています。

図 453 から 図 456 に、1 つのキャプチャ／比較チャネルの概要を示します。

入力ステージは、対応する tim_tix 入力をサンプリングして、フィルタリングを行った信号 tim_tixf を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（tim_tixfpy）を生成します。この信号はプリスケーラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 453. キャプチャ／比較チャネル（例：チャネル 1 入力ステージ）



出力ステージは、その後、基準として使用される中間波形 tim_ocxref（アクティブハイ）を生成します。信号の極性は最終出力に影響を与えます。

図 454. キャプチャ／比較チャネル 1 メイン回路

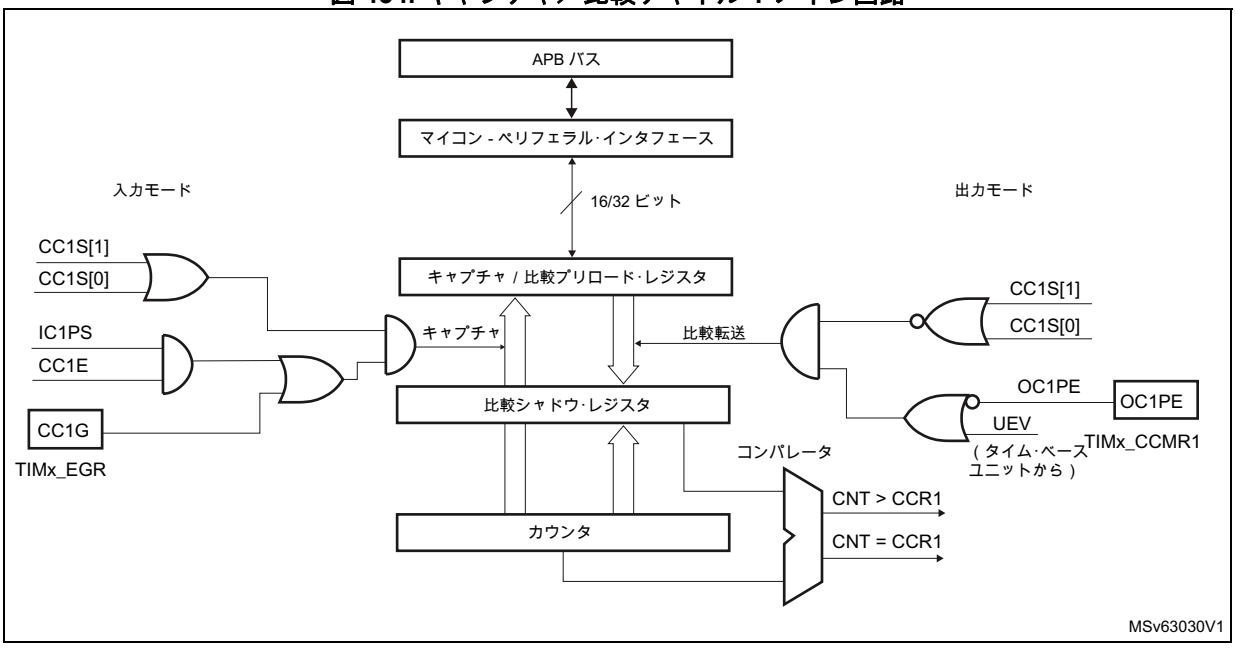


図 455. キャプチャ／比較チャネル（チャネル 1）の出力ステージ

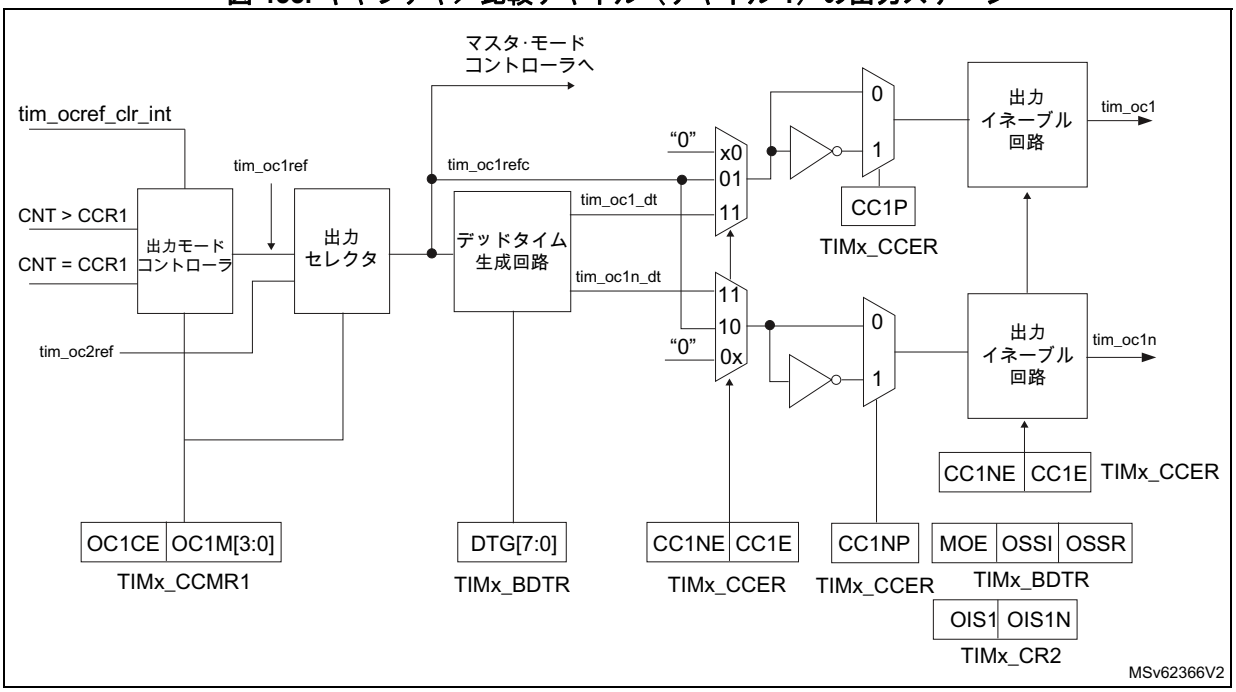
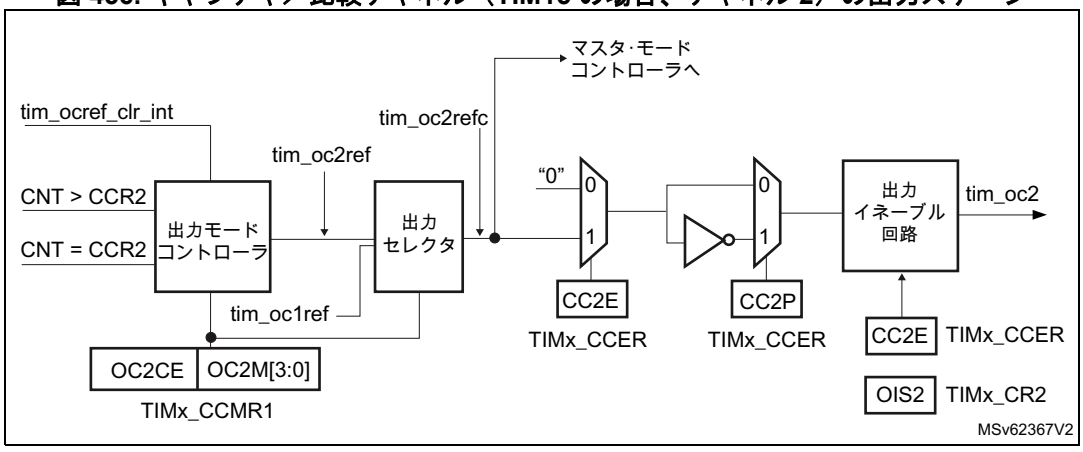


図 456. キャプチャ／比較チャネル（TIM15 の場合、チャネル 2）の出力ステージ



キャプチャ／比較ブロックは、1 つのプリロードレジスタと 1 つのシャドウレジスタで構成されています。書込みおよび読出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

30.4.8 入力キャプチャモード

入力キャプチャモードでは、対応する tim_icx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ（TIMx_CCRx）が使用されます。キャプチャが発生すると、対応する CCXIF フラグ（TIMx_SR レジスタ）がセットされ、割込みまたは DMA リクエストを送信できます（有効な場合）。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバキャプチャフラグ CCxOF（TIMx_SR レジスタ）がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、0 を書き込むとクリアされます。

次の例は、tim_ti1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な tim_ti1_in[15:1] ソース（内部または外部）を選択します。
2. アクティブ入力を選択します。TIMx_CCR1 は tim_ti1 入力とリンクされていなければならず、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャネルは入力に設定され、TIMx_CCR1 レジスタは読出し専用になります。
3. タイマに接続される信号に関して、適切な入力フィルタ時間をプログラムします（入力が tim_tix の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットによって）。入力信号の反転時、最低でも内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、tim_ti1 の遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。
4. tim_ti1 チャネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットに 0 を書き込みます（この場合、立ち上がりエッジの選択）。
5. 入力プリスケーラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいので、プリスケーラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
6. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
7. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE ビットをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読出し後、データ読出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： **IC 割込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。**

30.4.9 PWM 入力モード（TIM15 の場合のみ）

このモードでは、単一の tim_tix 入力に接続された PWM 信号の周期とデューティ･サイクルの両方を以下のように測定できます。

- TIMx_CCR1 レジスタは周期値（2 つの連続した立ち上がりエッジ間の間隔）を保持します。
- TIMx_CCR2 レジスタはパルス幅（2 つの連続した立ち上がりと立ち下がりエッジ間の間隔）を保持します。

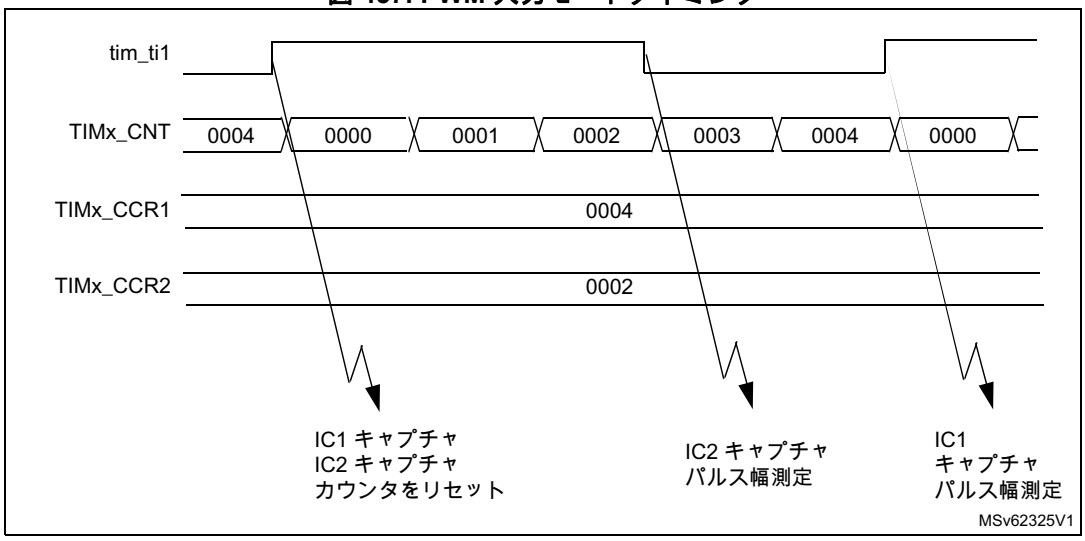
このモードは、入力キャプチャモードの特殊ケースです。設定手順は似ていますが、以下の違いがあります。

- 2 つの tim_icx 信号が同じ tim_tix 入力にマッピングされます。
- この 2 つの tim_icx 信号は、逆の極性のエッジでアクティブです。
- 2 つの tim_tixfpy 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、tim_ti1 に適用された PWM の周期（TIMx_CCR1 レジスタ）とデューティサイクル（TIMx_CCR2 レジスタ）を測定できます（手順は、tim_ker_ck 周波数とプリスケーラ値によって、若干異なることがあります）。

1. TIMx_TISEL レジスタの TI1SEL[3:0] ビットで、適切な tim_ti1_in[15:0] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR1のアクティブ入力を選択します。
3. CC1P ビットと CC1NP ビットに“0”を書き込むことによって（立ち上がりエッジでアクティブ）、tim_ti1fp1 のアクティブな極性を選択します（TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用）。
4. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって（tim_ti1 を選択）、TIMx_CCR2のアクティブ入力を選択します。
5. CC2P ビットと CC2NP ビットに“10”を書き込むことによって（立ち下がりエッジでアクティブ）、tim_ti1fp2 のアクティブ極性を選択します（TIMx_CCR2 のキャプチャに使用されます）。
6. TIMx_SMCR レジスタの TS ビットに 00101 を書き込むことによって（tim_ti1fp1 を選択）、有効なトリガ入力を選択します。
7. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
8. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。

図 457. PWM 入力モードタイミング



1. tim_ti1fp1 と tim_ti2fp2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2信号でのみ使用できます。

30.4.10 強制出力モード

このモード（TIMx_CCMRx レジスタの CCxS ビット = 00）では、各出力比較信号（tim_ocxref、そして tim_ocx/tim_ocxn）はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号（tim_ocxref/tim_ocx）を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 101 を書き込む必要があります。これにより、tim_ocxref は強制的にハイレベルになり（tim_ocxref は常にアクティブハイ）、tim_ocx は CCxP 極性ビットと逆の値になります。

例：CCxP=0（tim_ocx アクティブハイ）=> tim_ocx は強制的にハイレベルになります。

tim_ocxref 信号は、TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割込みや DMA リクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

30.4.11 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード（TIMx_CCMRx レジスタの OCxM ビット）と出力極性（TIMx_CCER レジスタの CCxP ビット）によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか（OCXM=000）、アクティブにセットされるか（OCxM=001）、非アクティブにセットされるか（OCxM=010）、または反転されます（OCxM=011）。
- 割込みステータスレジスタのフラグをセットします（TIMx_SR レジスタの CCxIF ビット）。
- 対応する割込みマスク（TIMx_DIER レジスタの CCXIE ビット）がセットされている場合は、割込みを生成します。
- 対応するイネーブルビット（TIMx_DIER レジスタの CCxDE ビット）がセットされている場合は、DMA リクエストを送信します（DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます）。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

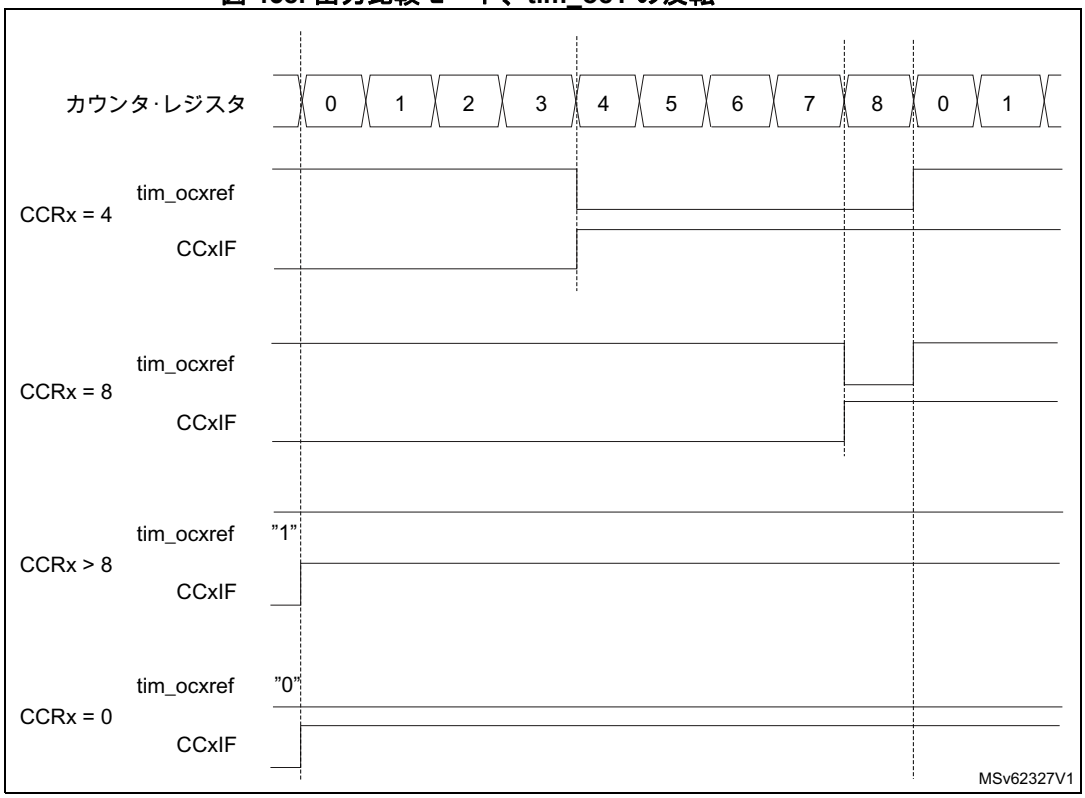
出力比較モードでは、更新イベント UEV は tim_ocxref および tim_ocx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます（ワンパルスモード）。

手順

1. カウンタクロックを選択します（内部、外部、プリスケーラ）。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例：
   - CNT と CCRx が一致したときに tim_ocx 出力ピンを反転するには、OCxM ビットに 011 を書き込みます。
   - プリロードレジスタを無効にするには、OCxPE ビットに 0 を書き込みます。
   - アクティブハイ極性を選択するには、CCxP ビットに 0 を書き込みます。
   - 出力を有効にするには、CCxE ビットに 1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限ります（OCxPE=0）。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を図 458 に示します。

図 458. 出力比較モード、tim_oc1 の反転



30.4.12 PWM モード

パルス幅変調（PWM）モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“110”（PWM モード 1）または“111”（PWM モード 2）を書き込むことによって、チャネルごとに選択できます（tim_ocx 出力ごとに 1 つの PWM）。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があり、また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも（アップカウントまたはセンターアラインモードで）有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

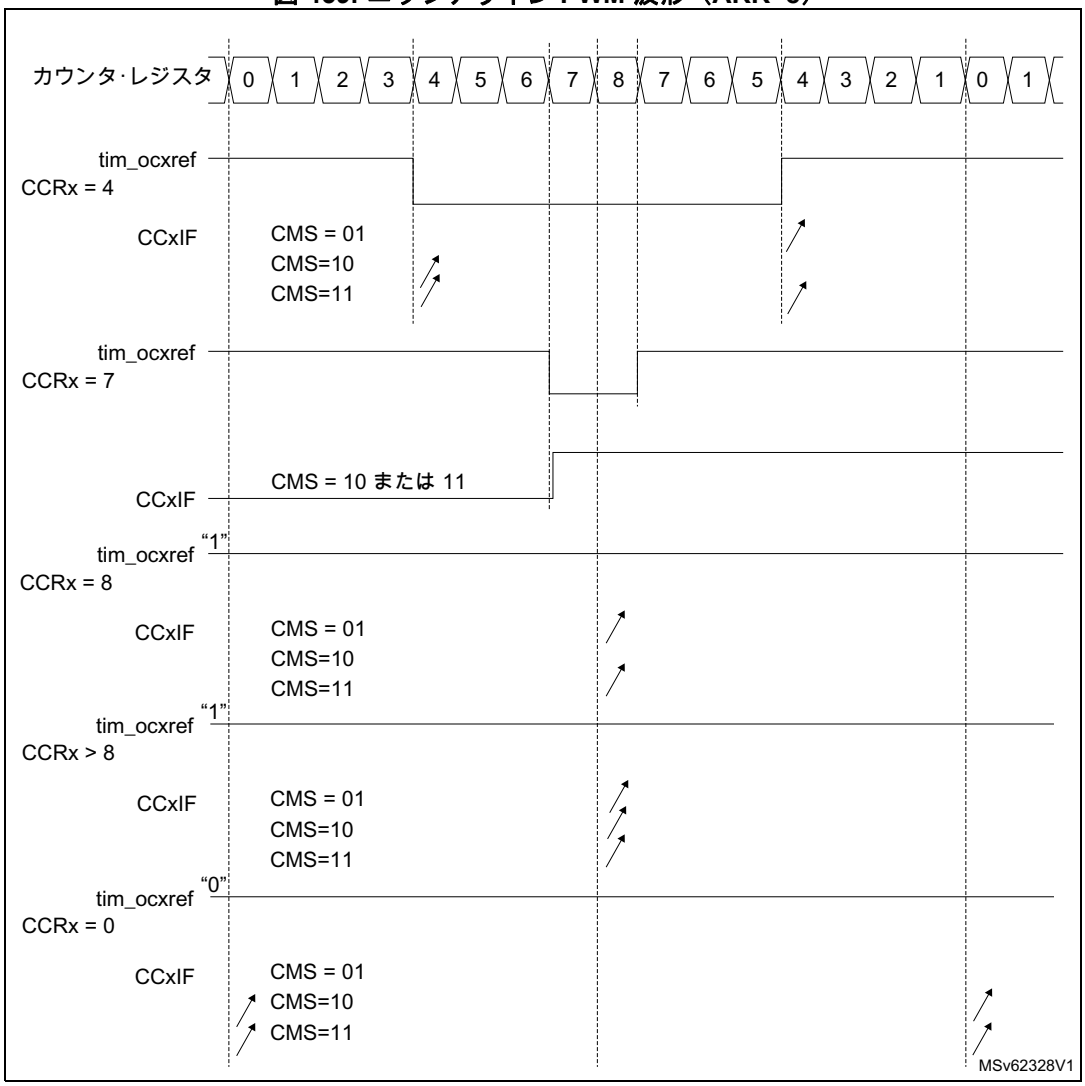
tim_ocx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。これは、アクティブハイまたはアクティブローとしてプログラムできます。tim_ocx 出力は、CCxE、CCxNE、MOE、OSSI、および OSSR ビット（TIMx_CCER および TIMx_BDTR レジスタ）の組み合わせによって有効になります。詳細については、TIMx_CCER レジスタの説明を参照してください。

PWM モード（1 または 2）では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx かどうかが判断されます（カウントの方向によります）。

TIM15/TIM16/TIM17 はアップカウント動作でのみ使用可能です。1319 ページのアップカウントモード を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 tim_ocxref は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、tim_ocxref は“1”に保持されます。比較値が 0 の場合、tim_ocxref は“0”に保持されます。図 459 に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 459. エッジアライン PWM 波形（ARR=8）

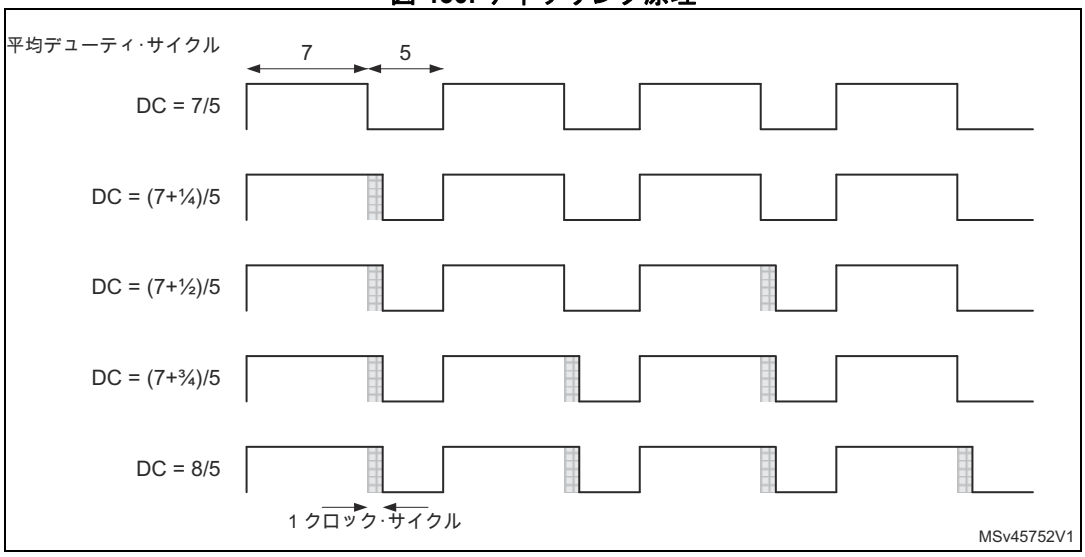


ディザリングモード

PWM モードの有効分解能は、TIMx_CR1 レジスタの DITHEN ビットを使用して、ディザリングモードを有効にすることで増加できます。これは、CCR（デューティサイクル分解能の増加）と ARR（PWM 周波数分解能の増加）の両方に適用されます。

動作原理は、事前に定義されたパターンで、16 個の連続した PWM 周期にわたって実際の CCR（または ARR）値をわずかに変更する（1 タイマクロック周期を追加する、またはしない）ことです。これにより、平均のデューティサイクルまたは PWM 周期を考えると、16 倍の分解能の向上が可能になります。下記の図 460 は、4 つの連続する PWM サイクルに適用されたディザリング原理を示します。

図 460. ディザリング原理

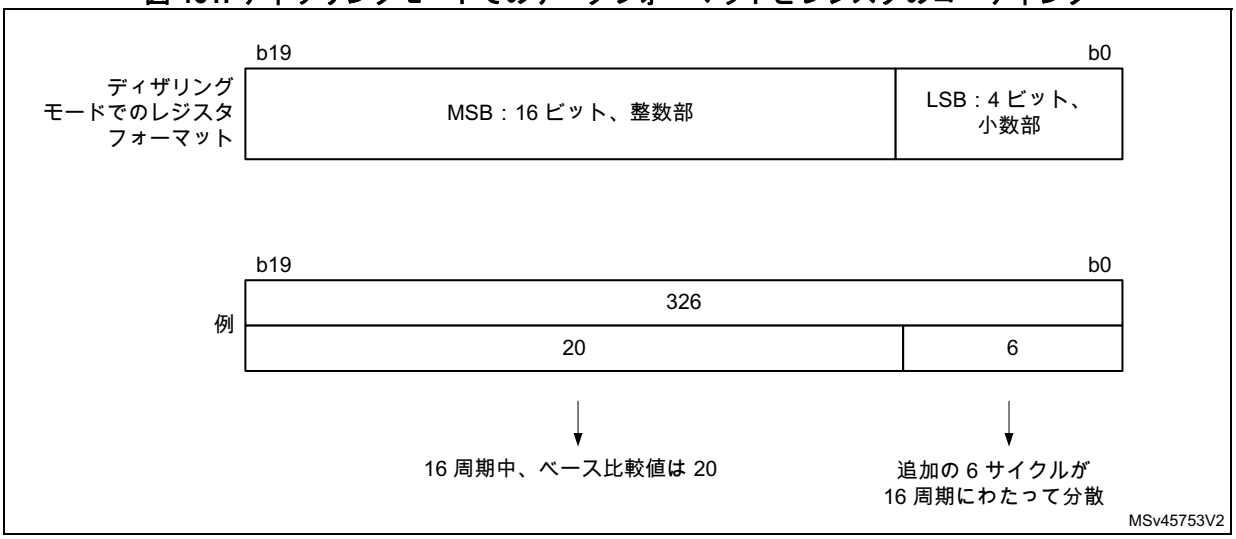


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図 461 を参照）。

- 下位 4 ビットは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット 19:4 に左シフトされ、ベース値用のコーディングになります。

注： **DITHEN ビットがセット／リセットされると、ARR と CCR の値が自動的に更新されます（たとえば、DITHEN = 0 で ARR = 0x05 の場合、DITHEN = 1 で ARR = 0x50 に更新されます）。DITHEN ビットをリセットするときは、次の手順に従う必要があります。**
1. CEN および ARPE ビットをリセットする必要があります。
2. ARR[3:0] ビットをリセットする必要があります。
3. DITHEN ビットをリセットする必要があります。
4. CCIF フラグをクリアする必要があります。
5. CEN ビットをセットすることができます（最終的に ARPE = 1 とともに）。

図 461. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

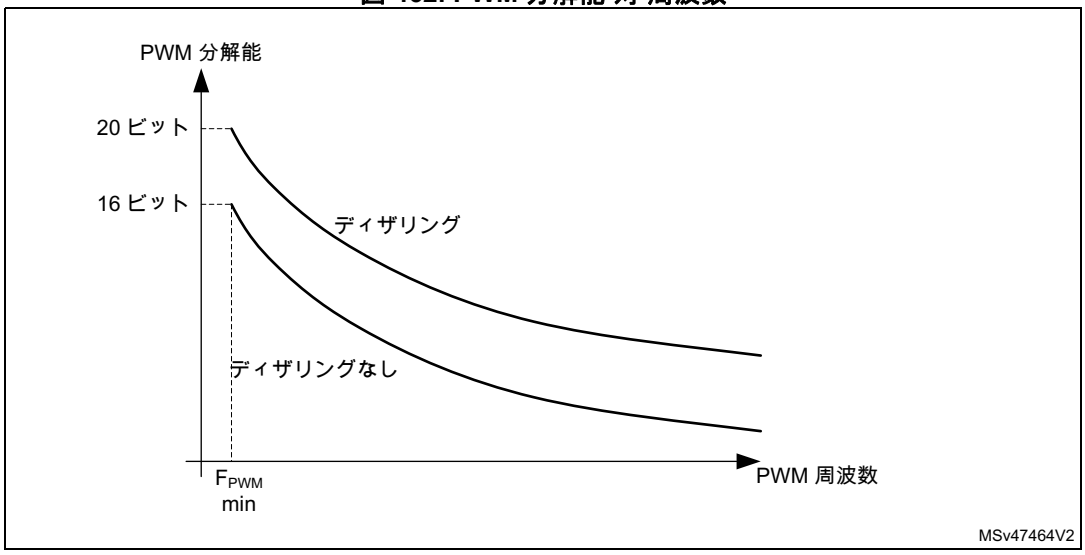
$$\text{ディザリングモード無効時：} F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモード有効時：} F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARR および TIMxCCRy の最大値は、ディザリングモードでは 0xFFFEF に制限されています（整数部では 65534、ディザリング部分では 15 に対応します）。**

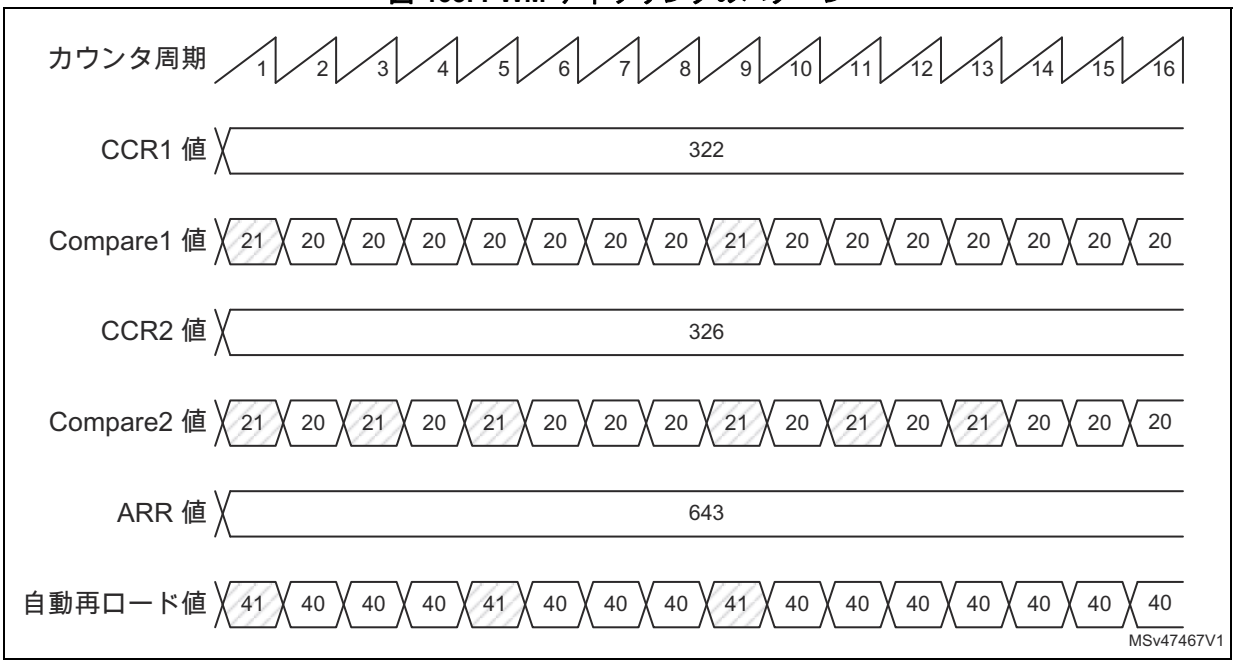
下記の図 462 に示すように、PWM 周波数に関係なく、ディザリングモードによって PWM 分解能を増加することができます。

図 462. PWM 分解能 対 周波数



下記の図 463 で説明するように、デューティサイクルや周期の変更は、連続した 16 周期にわたって広がります。

図 463. PWM ディザリングのパターン



自動再ロード値と比較値の増分は、下記の表 295 に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表 295. CCR および ARR レジスタの変更のディザリングパターン

-	PWM 周期															
LSB 値	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-

表 295. CCR および ARR レジスタの変更のディザリングパターン （続き）

-	PWM 周期															
LSB 値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

30.4.13 組み合わせ PWM モード（TIM15 のみ）

組み合わせ PWM モードでは、2 つのエッジアラインまたはセンターアライン PWM 信号を生成でき、それぞれのパルス間に遅延および位相シフトをプログラムできます。周波数が TIMx_ARR レジスタの値で決定されるのに対し、デューティサイクルや遅延は 2 つの TIMx_CCRx レジスタで決定されます。結果として得られる信号 tim_ocxrefc は、2 つの PWM 基準信号の OR または AND による論理結合から成ります。

- tim_oc1refc（または tim_oc2refc）は、TIMx_CCR1 および TIMx_CCR2 レジスタによって制御されます。

組み合わせ PWM モードは、TIMx_CCMRx レジスタの OCxM ビットに“1100”（組み合わせ PWM モード 1）または“1101”（組み合わせ PWM モード 2）を書き込むことによって、2 チャネルごとに選択できます（CCR レジスタペアごとに 1 つの tim_ocx 出力）。

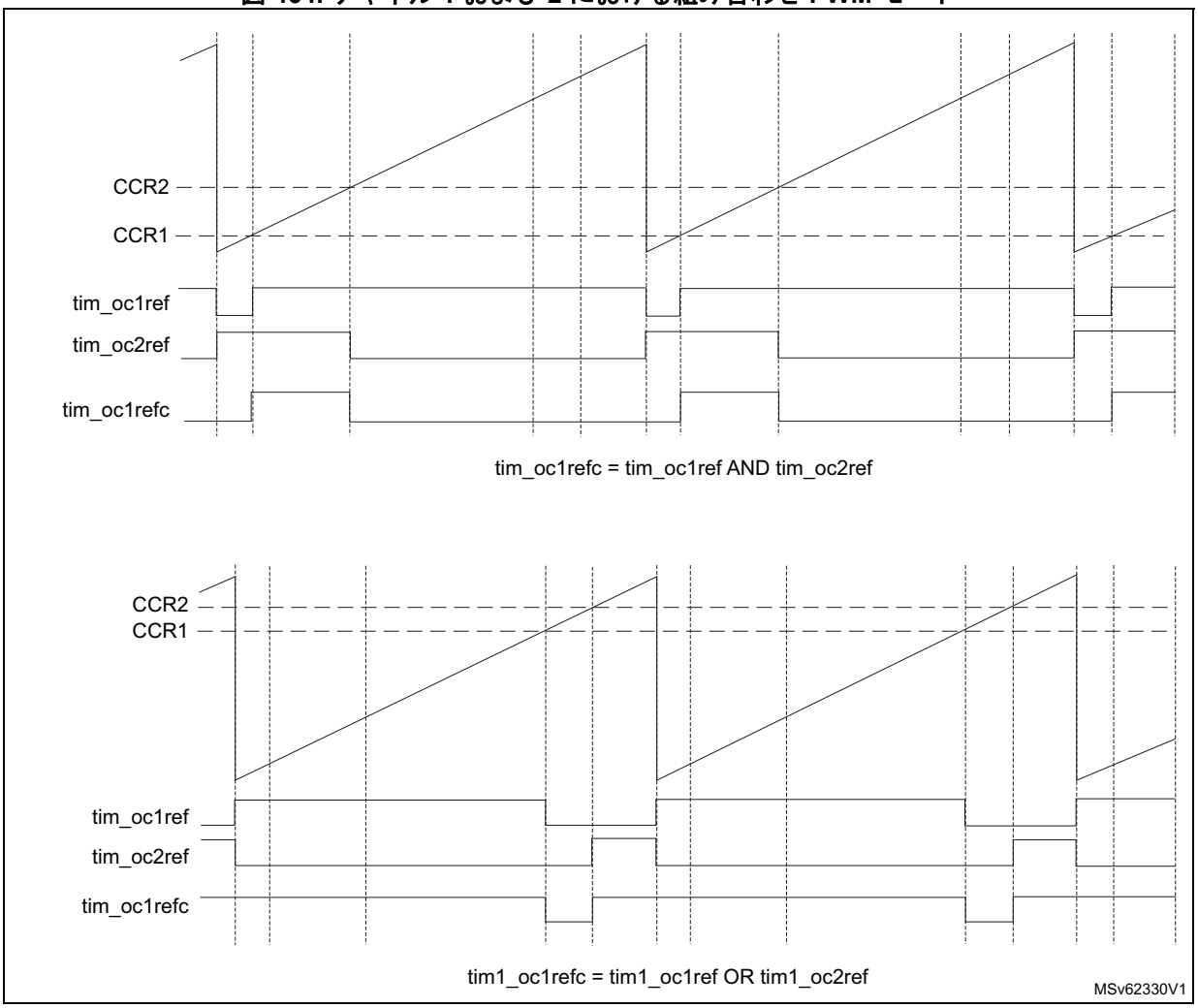
特定のチャネルが組み合わせ PWM チャネルとして使用されている場合、相補チャネルを反対の PWM モードに設定する必要があります（たとえば、1 つを組み合わせ PWM モード 1、もう 1 つを組み合わせ PWM モード 2 にします）。

注： **OCxM[3:0] ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

図 464 は、次の設定で取得可能な組み合わせ PWM モードを使用して生成される信号の例を表します。

- チャネル 1 が組み合わせ PWM モード 2 で設定されている場合
- チャネル 2 が PWM モード 1 で設定されている場合

図 464. チャネル 1 および 2 における組み合わせ PWM モード



30.4.14 相補出力とデッドタイム挿入

TIM15/TIM16/TIM17 汎用タイマは、1 つの相補信号を出力して、出力時のスイッチオフおよびスイッチオンを管理できます。

この時間は、通常、デッドタイムと呼ばれ、出力に接続されているデバイスとその特性（レベルシフタの内在的な遅延、電源スイッチによる遅延など）に応じて調整する必要があります。

出力（主出力 tim_ocx または相補 tim_ocxn）の極性は出力ごとに独自に選択できます。これは TIMx_CCER レジスタの CCxP ビットおよび CCxNP ビットへの書込みによって行います。

相補信号 tim_ocx および tim_ocxn は、TIMx_CCER レジスタの CCxE ビットと CCxNE ビット、TIMx_BDTR レジスタと TIMx_CR2 レジスタの MOE、OISx、OISxN、OSSI、および OSSR ビットといった複数の制御ビットの組み合わせによって有効になります。詳細については、表 302 : 1398 ページのブレーク機能を持つ相補 tim_oc1 および tim_oc1n チャネルの出力制御ビット 機能（TIM16/TIM17）を参照してください。特に、IDLE 状態に切り替わるとき（MOE が 0 になるとき）に、デッドタイムが挿入されます。

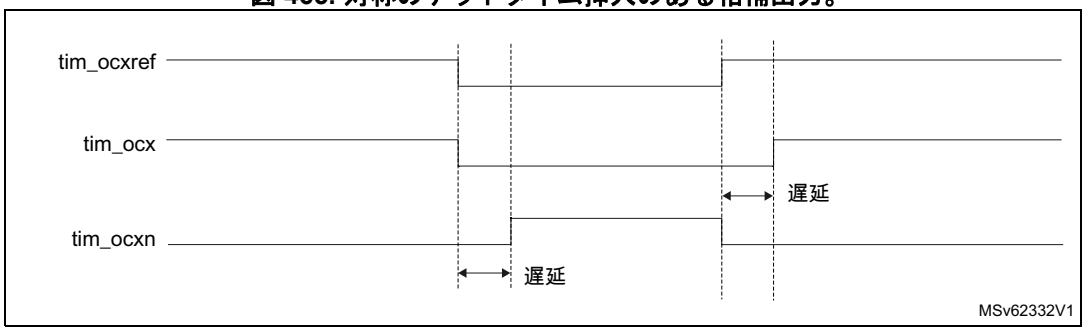
デッドタイム挿入は、CCxE ビットと CCxNE ビットの両方をセットし、ブレーク回路がある場合は、さらに MOE ビットをセットすることによって有効になります。各チャネルに 1 つの 10 ビットデッドタイムジェネレータがあります。この回路は、基準波形 tim_ocxref から tim_ocx と tim_ocxn の 2 つの出力を生成します。tim_ocx と tim_ocxn がアクティブハイの場合、

- tim_ocx 出力信号は基準信号と同じですが、立ち上がりエッジが基準の立ち上がりエッジより遅い点が異なります。
- tim_ocxn 出力信号は、立ち上がりエッジが基準波形の立ち下がりエッジから遅れている点を除けば、基準信号を反転させた波形と同じです。

遅延がアクティブ出力（tim_ocx または tim_ocxn）の幅より大きい場合、対応するパルスは生成されません。

以下の図は、デッドタイム生成回路の出力信号と基準信号 tim_ocxref との関係を示します。（これらの例では、CCxP=0、CCxNP=0、MOE=1、CCxE=1、および CCxNE=1 を想定しています。）

図 465. 対称のデッドタイム挿入のある相補出力。



TIMx_DTR2 の DTAE ビットによって、図 466 に示すように、デッドタイム値を、基準信号の立ち上がりエッジと立ち下がりエッジに対して変えることができます。

非対称モード（DTAE = 1）では、立ち上がりエッジ基準デッドタイムは TIMx_BDTR レジスタの DTG[7:0] ビットフィールドによって定義され、立ち下がりエッジ基準は TIMx_DTR2 レジスタの DTGF[7:0] ビットフィールドによって定義されます 。DTAE ビットは、カウンタを有効にする前に書き込む必要があり、CEN=1 の間は変更してはなりません。

プリロードメカニズムを使用して、PWM 運転中にデッドタイム値を動作中に更新することができます。デッドタイムビットフィールド DTG[7:0] と DTGF[7:0] は、TIMX_DTR2 レジスタで DTPE ビットがセットされるとプリロードされます。プリロード値は、次の更新イベントでアクティブレジスタにロードされます。

注： **カウンタが有効なときに DTPE ビットを有効にした場合、最後の更新以降に書き込まれた新しい値は破棄され、以前の値が使用されます。**

図 466. 非対称のデッドタイム

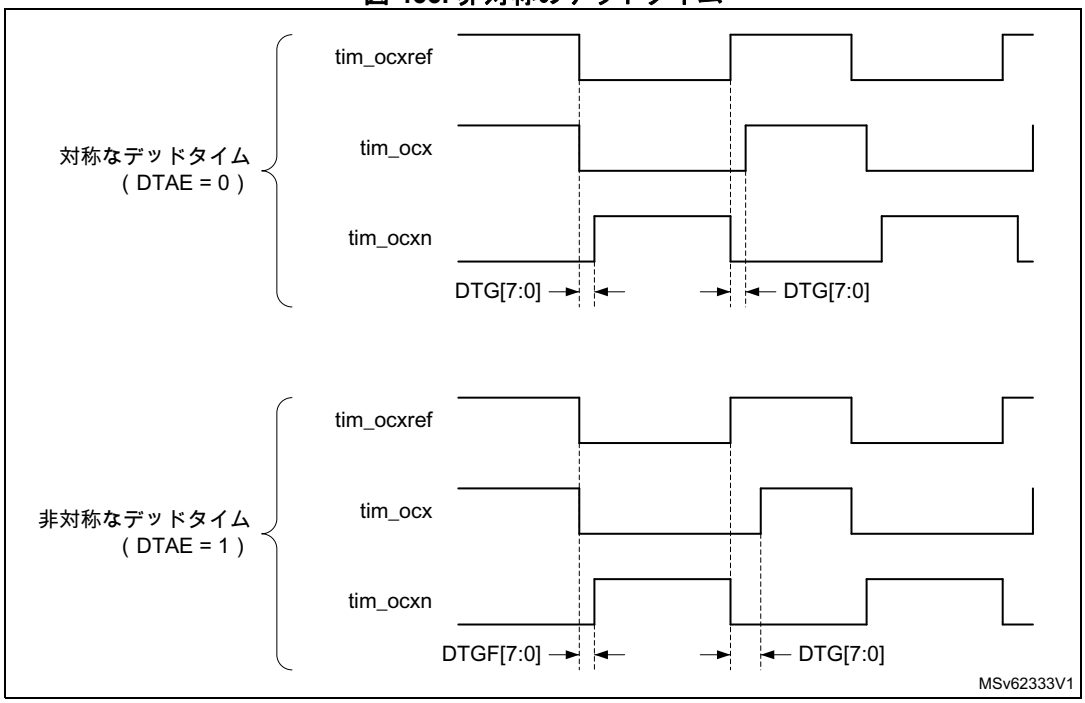


図 467. 負のパルスより長い遅延があるときのデッドタイムの波形

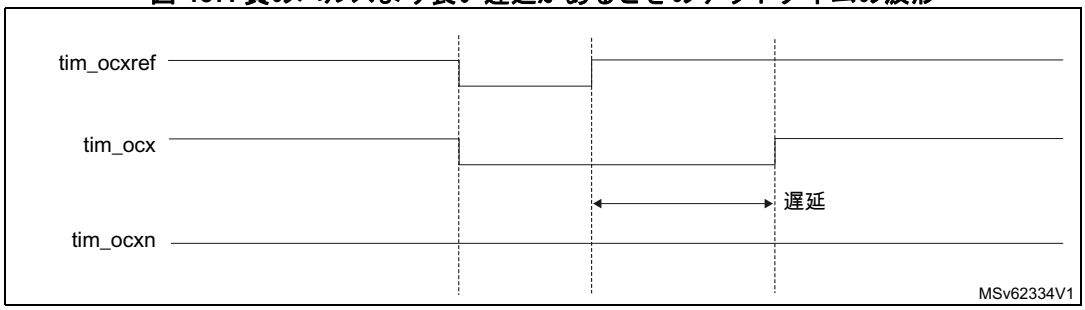
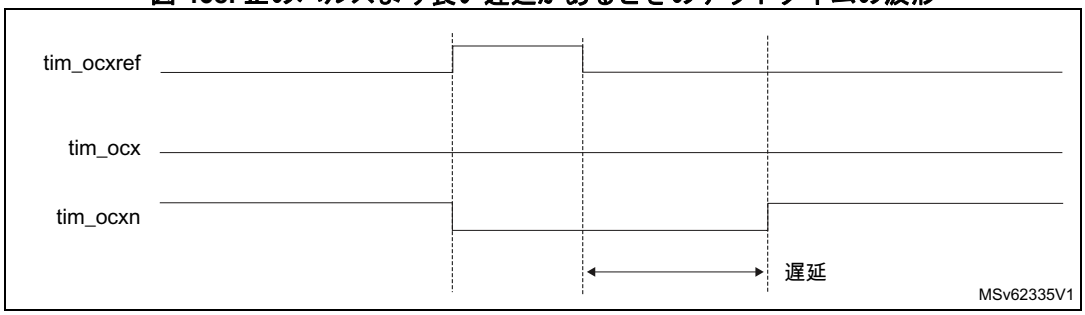


図 468. 正のパルスより長い遅延があるときのデッドタイムの波形



デッドタイム遅延は、各チャネルで同じであり、TIMx_BDTR レジスタの DTG ビットでプログラム可能です。遅延計算については、セクション 30.8.14：1402 ページのTIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）を参照してください。

tim_ocxref 信号の tim_ocx または tim_ocxn へのリダイレクト

出力モード（強制、出力比較、または PWM）では、TIMx_CCER レジスタの CCxE ビットおよび CCxNE ビットを設定することによって、tim_ocxref 信号を tim_ocx 出力または tim_ocxn 出力にリダイレクトできます。

これにより、特定の波形（PWM または静的アクティブレベルなど）を一方の出力に送信し、補信号をインアクティブレベルに固定することができます。他の例としては、両方の出力をインアクティブレベルにしたり、両方の出力をアクティブにして、デッドタイムのある相補出力とすることができます。

注： **tim_ocxn のみが有効なときには（CCxE=0、CCxNE=1）、相補にならず、tim_ocxref がハイレベルになるとすぐにアクティブになります。たとえば、CCxNP=0 の場合は、tim_ocxn=tim_ocxref になります。他方、tim_ocx と tim_ocxn の両方が有効なときには（CCxE=CCxNE=1）、tim_ocxref がハイになると tim_ocx はアクティブになり、tim_ocxn は補信号となり、tim_ocxref がローのときにアクティブになります。**

30.4.15 ブレーク機能の使用

ブレーク機能の目的は、タイマによって生成される PWM 信号によって駆動する電源スイッチを保護することです。ブレーク入力は通常、パワーステージおよび 3 相インバータの異常出力に接続されています。アクティブ化すると、ブレーク回路は PWM 出力を遮断し、強制的に事前定義された安全な状態に移行させます。

このブレーク･チャネルは、システム･レベル障害（クロック障害、ECC エラー、パリティ･エラーなど）とアプリケーション障害（入力ピンおよび内蔵比較回路）の両方を収集し、デッドタイムの持続時間経過後に、事前定義のレベル（アクティブまたはインアクティブ）に出力を強制できます。

ブレーク時の出力有効化信号と出力レベルは、いくつかの制御ビットで決まります。

- TIMx_BDTR レジスタの MOE ビット。ソフトウェアによって出力を有効／無効にすることができます。また、ブレーク または ブレーク2 イベント時にリセットされます。
- TIMx_BDTR レジスタの OSSI ビット。出力をインアクティブ状態で制御するか、GPIO コントローラへの制御を解除するかについて、タイマを定義します（通常、ハイインピーダンスモードにするため）。
- TIMx_CR2 レジスタの OISx および OISxN ビット。アクティブまたはインアクティブな出力遮断レベルをセットします。OISx および OISxN の値にかかわらず、一度に tim_ocx 出力と tim_ocxn 出力を両方ともアクティブレベルにセットすることはできません。詳細については、表 302：1398 ページのブレーク機能を持つ相補 tim_oc1 および tim_oc1n チャネルの出力制御ビット 機能（TIM16/TIM17）を参照してください。

リセットが終了すると、ブレーク回路は無効になり、MOE ビットはローになります。ブレーク機能は、TIMx_BDTR レジスタの BKE ビットをセットすることによって有効になります。ブレーク入力の極性は、同じレジスタの BKP ビットを設定することによって選択できます。BKE と BKP は、同時に変更できます。BKE および BKP ビットが書き込まれるとき、書込みが有効になるまでに 1 APB クロックサイクルの遅延が適用されます。そのため、書込み動作の後、ビットを正しく読み出すためには 1 APB クロックサイクル待つ必要があります。

MOE の立ち下がりエッジは非同期のことがあるので、実際の信号（出力に作用する信号）と同期制御ビット（TIMx_BDTR レジスタからアクセスできる）の間に、再同期回路が挿入されています。このため、非同期信号と同期信号の間に若干の遅延が発生します。特に、MOE がローになった後で 1 に設定された場合、MOE を正しく読み出すためには、遅延（ダミー命令）を挿入する必要があります。これは、書込みは非同期信号に対応しますが、読出しは同期信号を反映するからです。

ブレークは、次を持つ tim_brk 入力によって生成されます。

- プログラム可能な極性（TIMx_BDTR レジスタの BKP ビット）
- プログラム可能なイネーブルビット（TIMx_BDTR レジスタの BKE ビット）
- 誤ったイベントを避けるためのプログラム可能なフィルタ（TIMx_BDTR レジスタの BKF[3:0] ビット）

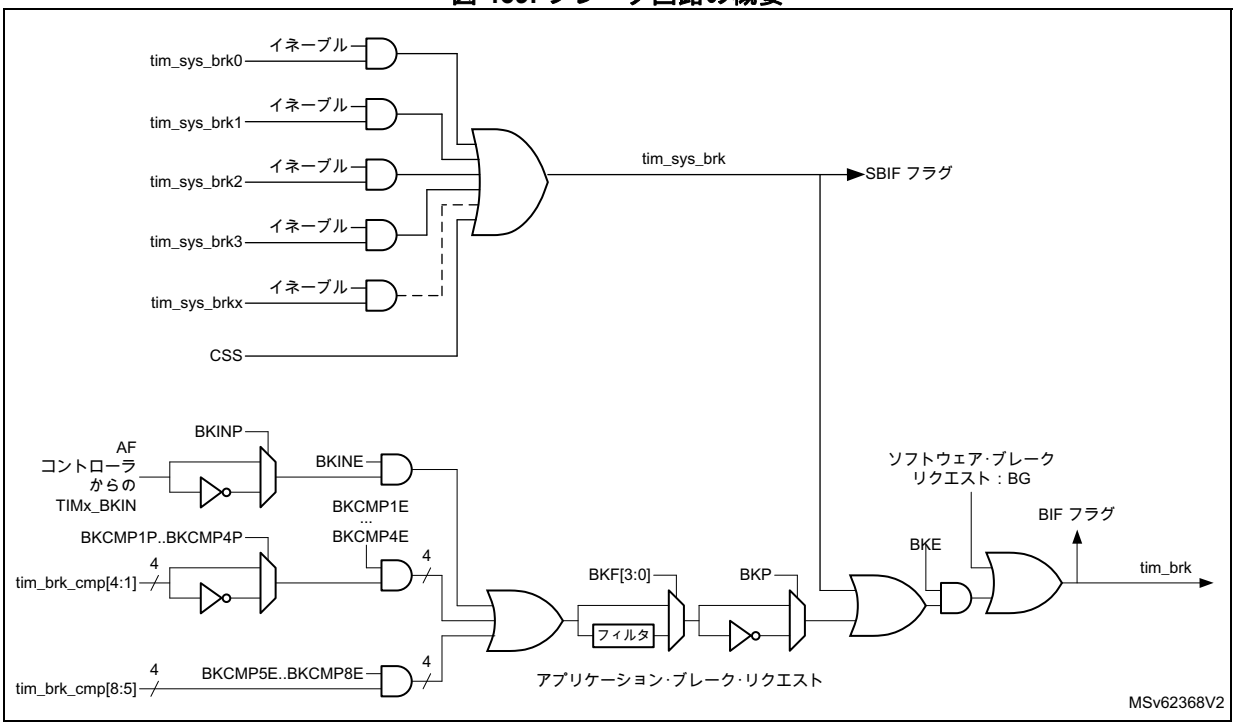
ブレークは、TIMx_AF1 レジスタを使用して、個別に有効化でき、プログラム可能なエッジ感度を持つ複数のソースから生成できます。

ブレーク（tim_brk）チャネルのソースは以下のいずれかです。

- TIMx_BKIN ピンの 1 つに接続された外部ソース（GPIO オルタネート機能選択レジスタの設定による）：極性選択機能およびオプションのデジタル･フィルタリング装備
- 内部ソース：
  - tim_brk_cmpx 入力から着信するもの（製品固有の実装についてはセクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照）
  - システムブレークリクエストから tim_sys_brk 入力へ着信するもの（製品固有の実装についてはセクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照）

ブレークイベントは、TIMx_EGR レジスタで BG ビットを使用して、ソフトウェアによって生成することもできます。以下の図 469 のとおり、すべてのソースはタイマの tim_brk 入力に入る前に論理和がとられます。

図 469. ブレーク回路の概要



注意： **非同期（クロックなし）動作は、プログラム可能なフィルタが無効な場合にのみ保証されます。有効になっている場合は、必ずブレークイベントが処理されるように、フェイルセーフクロックモード（たとえば、内部 PLL や CSS を使用）を使用する必要があります。**

ブレークが発生すると（ブレーク入力で選択されたレベル）、

- MOE ビットは非同期でクリアされ、出力をインアクティブ状態またはアイドル状態にできるほか、GPIO に制御を渡すこともできます（OSSI ビットで選択）。これは、マイクロコントローラのオシレータがオフでも機能します。
- 各出力チャネルは、MOE=0 になったとき、TIMx_CR2 レジスタの OISx ビットでプログラミングされたレベルで駆動されます。OSSI = 0 の場合、タイマは出力の制御を解放し（GPIO コントローラに渡されます）、それ以外の場合、イネーブル出力はハイのままです。
- 相補出力が使用されている場合は次のようになります。
  - 出力は、まずリセット状態のインアクティブ状態に置かれます（極性に依存します）。これは非同期に行われるので、タイマにクロックが供給されていないときでも機能します。
  - タイマクロックが供給されている場合、デッドタイム後に OISx および OISxN ビットでプログラミングされたレベルで出力を駆動するために、デッドタイムジェネレータが作動します。この場合でも、tim_ocx と tim_ocxn を同時にアクティブレベルに駆動することはできません。MOE の再同期により、デッドタイム時間が通常より少し長くなることに注意してください（約 2 tim_ker_ck クロックサイクル）。
  - OSSI = 0 の場合、タイマはイネーブル出力を解放し（GPIO コントローラに渡されることで強制的にハイ･インピーダンス状態になります）、それ以外の場合、イネーブル出力はハイのままか、CCxE ビットまたは CCxNE ビットがハイになったときにハイになります。
- ブレーク状態フラグ（TIMx_SR レジスタの BIF ビット）がセットされます。TIMx_DIER レジスタの BIE ビットがセットされている場合は、割込みを生成できます。TIMx_DIER レジスタの BDE ビットがセットされている場合、DMA リクエストを送信できます。
- TIMx_BDTR レジスタの AOE ビットがセットされている場合､MOE ビットは次の更新イベント UEV で再び自動的にセットされます。これを使用して、たとえば、レギュレーションを行うことができます。そうでない場合、MOE は、再び 1 が書き込まれるまでローのままです。この場合、セキュリティ目的で使用でき、パワー駆動回路、温度センサ、またはセキュリティコンポーネントからのアラームにブレーク入力を接続できます。

注： **AOE がセットされた状態で MOE が CPU によってリセットされると、出力はアイドル状態になり、OSSI の値に応じて非アクティブ･レベルまたはハイ･インピーダンスに強制的に設定されます。MOE と AOE の両方が CPU によってリセットされた場合は、出力が無効化され、TIMx_CR2 レジスタの OISx ビットにプログラムされたレベルに駆動されます。**

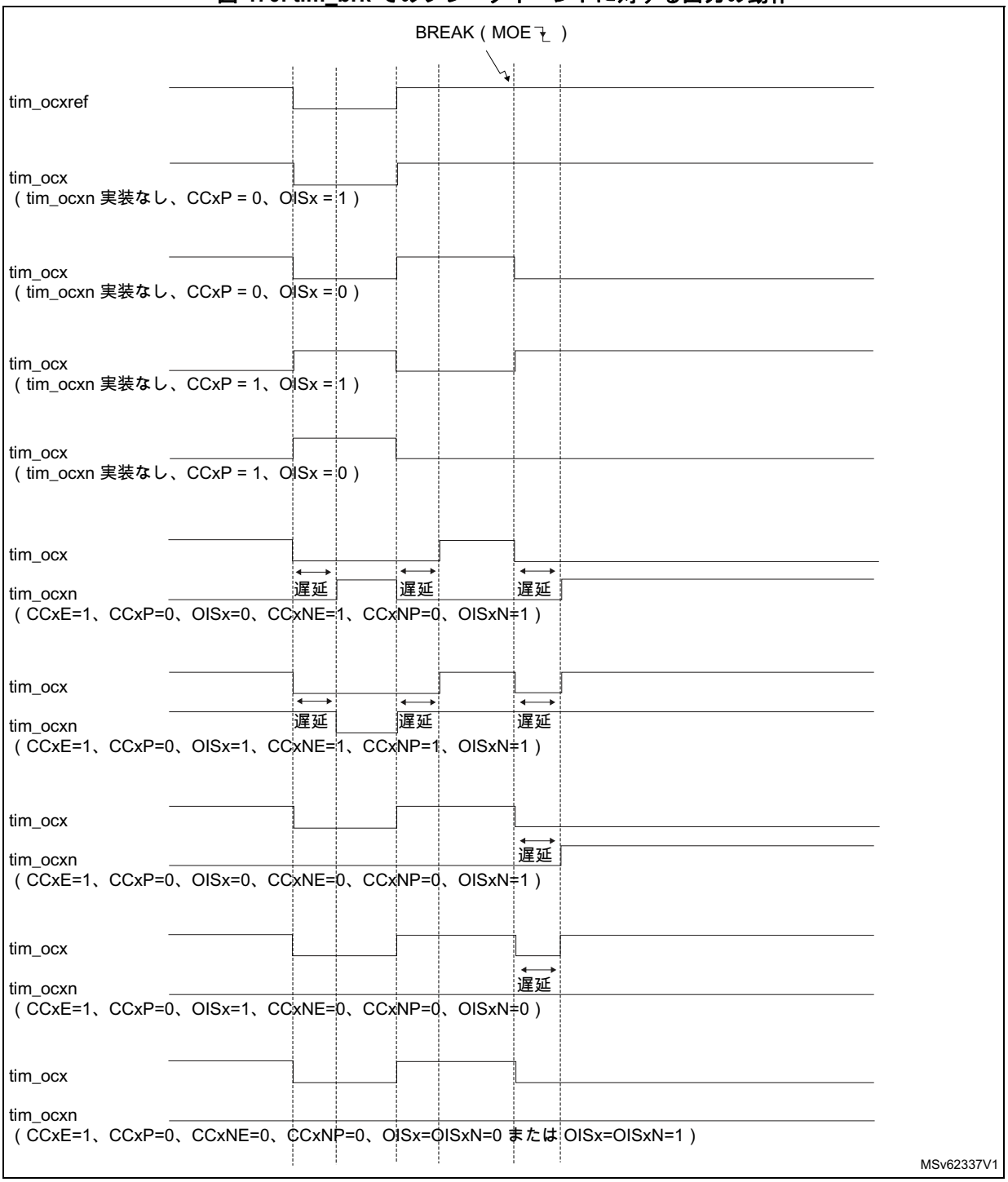
注： **ブレーク入力は、信号レベルに対して動作します。このため、ブレーク入力がアクティブな間は、MOE をセットできません（自動的にも、ソフトウェアによっても）。この間、ステータスフラグ BIF をクリアできません。**

ブレークは、tim_brk　入力によって生成でき、tim_brk　はプログラミング可能な極性を持ち、TIMx_BDTR レジスタの BKE がイネーブルビットです。

ブレーク入力と出力管理に加えて、アプリケーションに対する安全策として、ブレーク回路内に書込み保護機能を設けてあります。これにより､いくつかのパラメータ（デッドタイムの長さ､tim_ocx/tim_ocxn 極性､無効時の状態､OCxM 構成､ブレークイネーブルと極性）を固定することができます。TIMx_BDTR レジスタの LOCK ビットによって、保護を 3 つのレベルから選択することができます。セクション 30.8.14：TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）を参照してください。LOCK ビットは、MCU リセット後に 1 回だけ書き込むことができます。

図 470 に、ブレークに対する出力の動作例を示します。

図 470. tim_brk でのブレークイベントに対する出力の動作



30.4.16 双方向ブレーク入力

TIM15/TIM16/TIM17 には、図 471 に示すとおり、双方向ブレーク I/O が搭載されています。

次のものを使用できます。

- 入力および出力ステータスピンどちらにもなるユニークなピンを持ち、外部の MCU やゲートドライバに障害を通知するために使用できる幅広いレベルのグローバルブレーク信号
- 複数の内部および外部ブレークソースを統合する必要がある場合、互いに論理和をとり、ユニークなブレークイベントをトリガする内部ブレークソースおよび複数の外部オープンドレインソース

tim_brk 入力は、TIMxBDTR レジスタの BKBID ビットを使用して双方向モードに設定されます。BKBID プログラミングビットは、TIMxBDTR レジスタの LOCK ビットを使用して読出し専用モードにロックできます（LOCK レベル 1 以上）。

双方向モードでは、I/O をアクティブロー極性でオープンドレインモードに設定する必要があります（BKINP および BKP ビットを使用）。システム（例 : CSS）、オンチップペリフェラル、またはブレーク入力からのブレークリクエストによって、障害イベントを通知するブレーク入力のレベルが強制的にローとなります。安全性のため、極性ビットが正しくセット（アクティブハイ極性）されていない場合の双方向モードは禁止されています。

（BG ビットのセットによってトリガされる）ブレークソフトウェアイベントも、ブレーク I/O を強制的に 0 にして、タイマがブレーク状態に移行したことを外部コンポーネントに示します。ただし、これはブレークが有効になっている場合（BKE = 1）にのみ有効です。ソフトウェアブレークイベントが BKE = 0 で生成された場合、出力は安全な状態になり、ブレークフラグはセットされますが、TIM_BKIN I/O には効果がありません。

安全解除メカニズムは、システムが完全にロックされることを防ぎます（ブレーク入力でレベルがローになると、同じ入力でレベルをローに強制するブレークがトリガされます）。

BKDSRM ビットが 1 にセットされると、ブレーク出力が解放され、障害信号をクリアしてシステムを再設定できます。

ブレーク保護回路はどの時点においても無効にできません。

- ブレーク入力パスは常にアクティブです。BKDSRM ビットがセットされており、オープンドレイン制御が解放されている場合でもブレークイベントはアクティブです。これによって、ブレーク条件が存在する限り、PWM 出力の再開を防ぐことができます。
- 出力が有効になっている（MOE ビットがセットされている）限り、BKDSRM ビットでブレーク保護を解除できません（表 296 を参照）。

表 296. ブレーク保護解除条件

MOE	BKBID	BKDSRM	ブレーク保護状態
0	0	X	設定
0	1	0	設定
0	1	1	解除
1	X	X	設定

ブレーク回路の設定および再設定

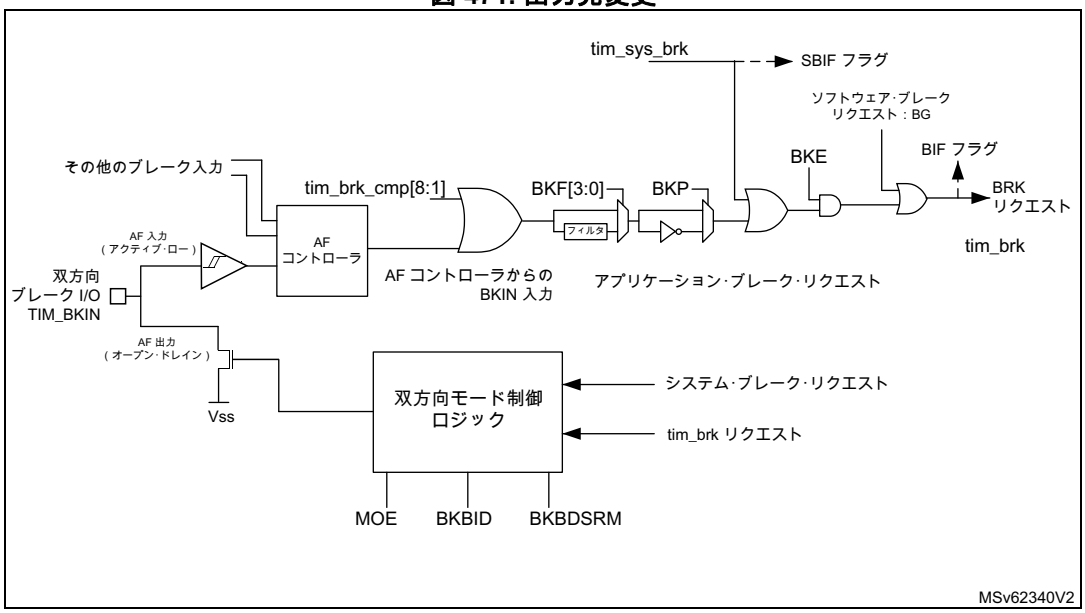
ブレーク回路（入力モードまたは双方向モード）はデフォルト（ペリフェラルリセット設定）で設定されています。

ブレークイベント後に保護を再び設定するには次の手順に従う必要があります。

- BKDSRM ビットをセットして、出力制御を解放する必要があります。
- ソフトウェアは、システムブレーク条件（ある場合）がなくなるまで待ってから、SBIF ステータスフラグをクリアする必要があります（または再設定の前にシステムでクリアします）。
- ソフトウェアは、BKDSRM ビットをハードウェアによってクリアされるまでポーリングする必要があります（アプリケーションブレーク条件がなくなるとき）。

この時点から、ブレーク回路は設定されアクティブになり、MOE ビットをセットして、PWM 出力を再び有効にできます。

図 471. 出力先変更

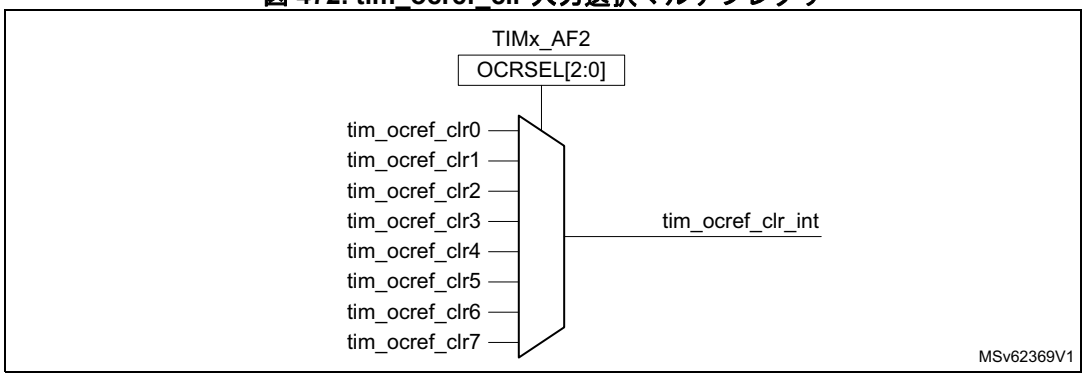


30.4.17 外部イベントによる tim_ocxref 信号のクリア

特定のチャネルの tim_ocxref 信号は tim_ocref_clr_int 入力にハイレベルを適用するとクリアされます（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）。tim_ocxref は、次の更新イベント（UEV）が発生するまでローのままになります。この機能は、出力比較モードと PWM モードでのみ使用可能です。強制モードでは動作しません。

tim_ocref_clr_int 入力は、以下の図 472 に示すように、複数の入力から選択できます。

図 472. tim_ocref_clr 入力選択マルチプレクサ



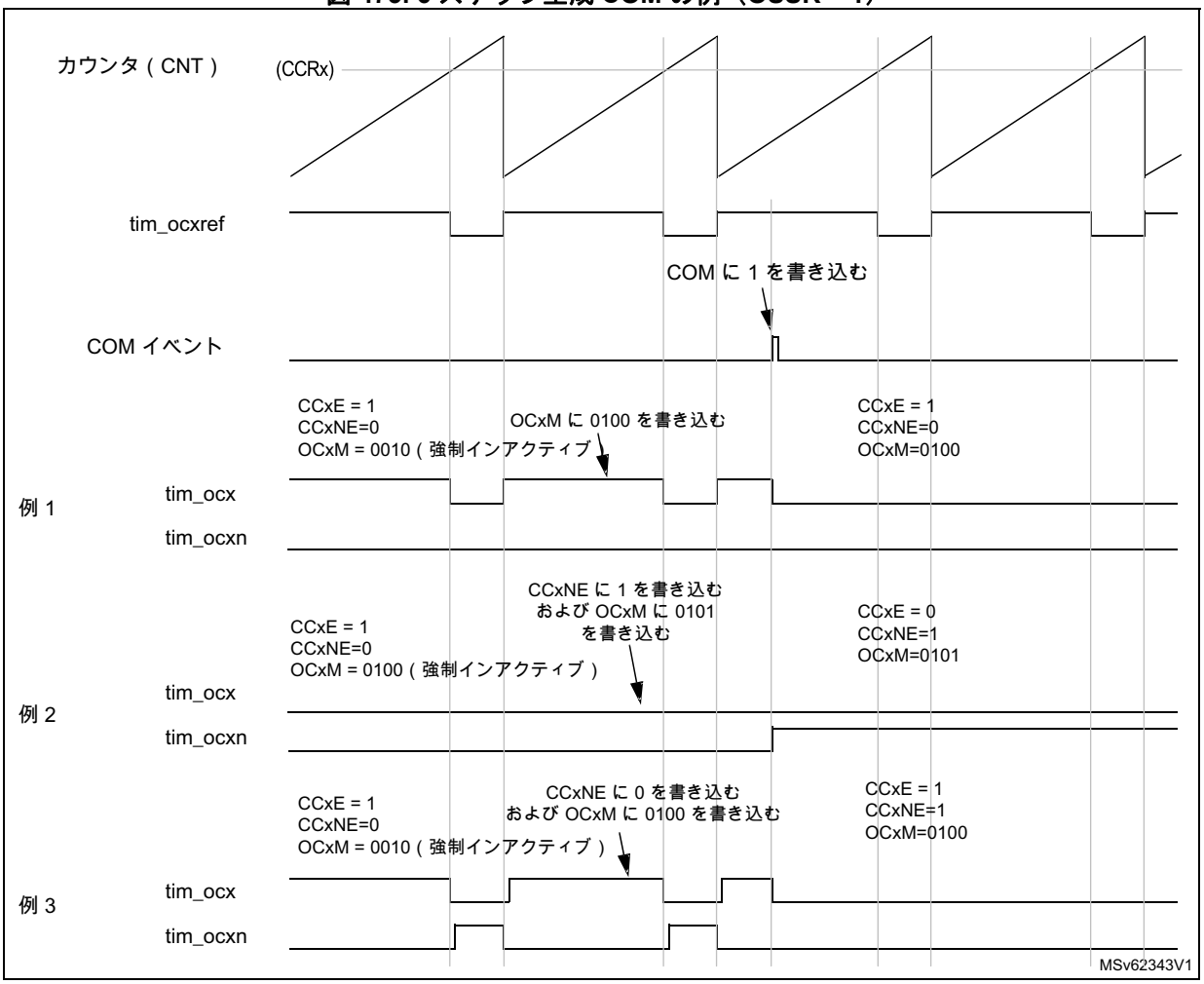
30.4.18 6 ステップ PWM 生成

チャネルで相補出力を使用しているときは、OCxM、CCxE、CCxNE の各ビットでプリロード･ビットを使用できます。プリロードビットは、COM 転流イベントでシャドウビットにコピーされます。これにより、次のステップの構成をあらかじめプログラミングして、すべてのチャネルの構成を同時に変更することができます。COM は、TIMx_EGR レジスタの COM ビットをセットすることによってソフトウェアによって、またはハードウェアによって（tim_trgi 立ち上がりエッジで）生成することができます。

フラグは、COM イベントが発生したときにセットされ（TIM x _SR レジスタの COMIF ビット）、これによって割込み（TIMx_DIER レジスタの COMIE ビットがセットされている場合）または DMA リクエスト（TIMx_DIER レジスタの COMDE ビットがセットされている場合）を生成できます。

図 473 は、COM イベント発生時の tim_ocx と tim_ocxn 出力の動作を、3 種類の設定をプログラムした場合について示した例です。

図 473. 6 ステップ生成 COM の例（OSSR = 1）



30.4.19 ワンパルスモード

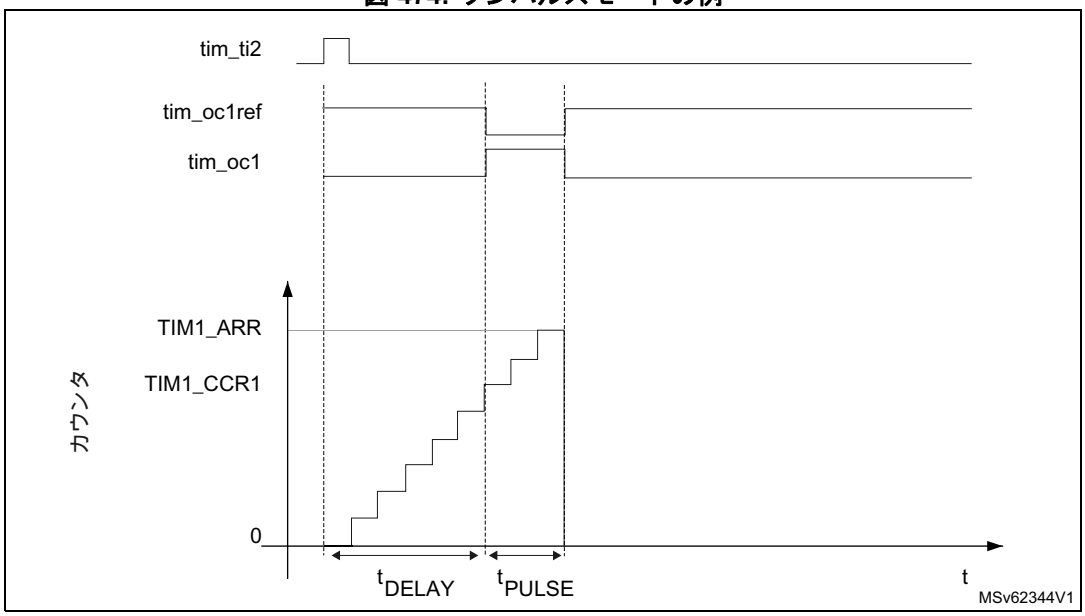
ワンパルスモード（OPM：One Pulse Mode）は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に（タイマがトリガを待っているときに）、設定が次のようでなければなりません。

- CNT < CCRx ≤ ARR （特に、0 < CCRx）

図 474. ワンパルスモードの例



たとえば、tim_ti2 入力ピンで立ち上がりエッジが検出されたときに、tim_oc1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

tim_ti2fp2 をトリガ 1 として使用します。

1. TIMx_TISEL レジスタの TI2SEL[3:0] ビットで、適切な tim_ti2_in[15:1] ソース（内部または外部）を選択します。
2. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、tim_ti2fp2 を tim_ti2 にマッピングします。
3. tim_ti2fp2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P=“0”と CC2NP =“0”を書き込みます。
4. tim_ti2fp2 をスレーブモードコントローラのトリガ（tim_trgi）として構成するために、TIMx_SMCR レジスタの TS=“00110”を書き込みます。
5. tim_ti2fp2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110”（トリガモード）を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます（クロック周波数とカウンタプリスケーラを考慮に入れて）。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差（TIMx_ARR - TIMx_CCR1）によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタに OC1M=111 を書き込むことによって、PWM モード 2 を有効にする必要があります。オプションで、TIMx_CCMR1 レジスタに OC1PE='1' を書き込み、TIMx_CR1 レジスタの ARPE ビットを書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込む必要があります。次に、UG ビットをセットすることによって更新を生成し、tim_ti2 で外部トリガイベントを待つ必要があります。この例では、CC1P に“0”を書き込みます。

必要なパルスは 1 つだけなので、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、カウンタは次の更新イベント時に停止します（カウンタが自動再ロード値に達して、“0”に戻る時点）。

特殊ケース：tim_ocx 高速イネーブル

ワンパルスモードでは、tim_tix 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延（t_{DELAY} min）が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、tim_ocxref（および tim_ocx）は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

30.4.20 再トリガ可能なワンパルス･モード（OPM）（TIM15 のみ）

このモードでは、トリガに応じてカウンタを開始して、プログラム可能な長さのパルスを生成できます。ただし、セクション 30.4.19 : で説明する再トリガ不可能なワンパルス･モードと、次のような違いがあります。

- パルスはトリガが発生し次第開始します（プログラム可能な遅延はありません）。
- パルスは、前のトリガが完了する前に新しいトリガが発生すると拡張されます。

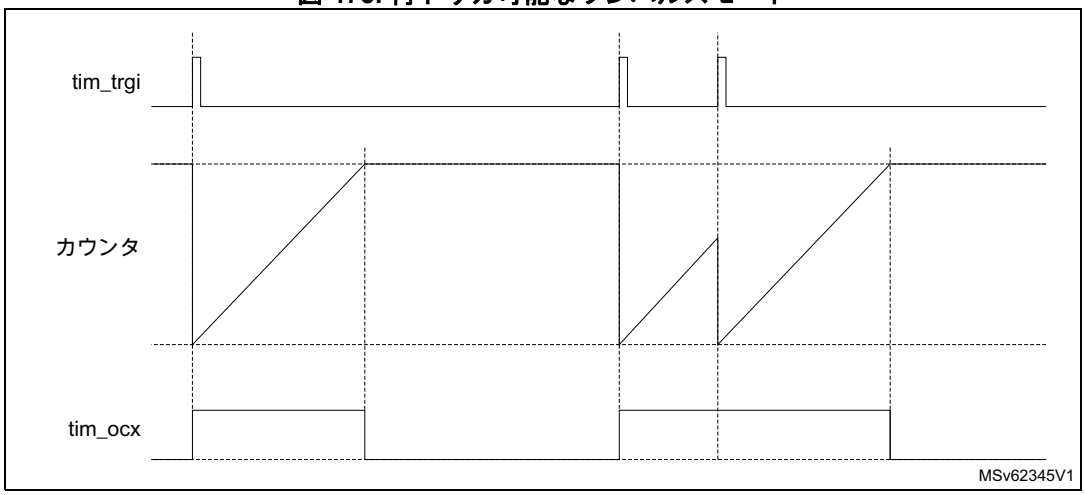
タイマはスレーブモードである必要があり、このときビットは TIMx_SMCR レジスタで SMS[3:0] =“1000”（リセットモードとトリガモードの組み合わせ）、および再トリガ可能な OPM モード 1 または 2 で OCxM[3:0] が“1000”または“1001”にセットされています。

タイマをアップカウントモードで設定した場合、対応する CCRx を 0 にセットする必要があります（ARR レジスタによってパルス長がセットされます）。タイマをダウンカウントモードで設定した場合、CCRx は ARR 以上である必要があります。

注： **OCxM[3:0] および SMS[3:0]ビットフィールドは互換性を確保するために 2 つのパーツに分割され、最上位ビットと 3 つの最下位ビットとは隣接していません。**

このモードをセンターアライン PWM モードと組み合わせて使用することはできません。TIMx_CR1 では、CMS[1:0] = 00 にする必要があります。

図 475. 再トリガ可能なワンパルスモード



30.4.21 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

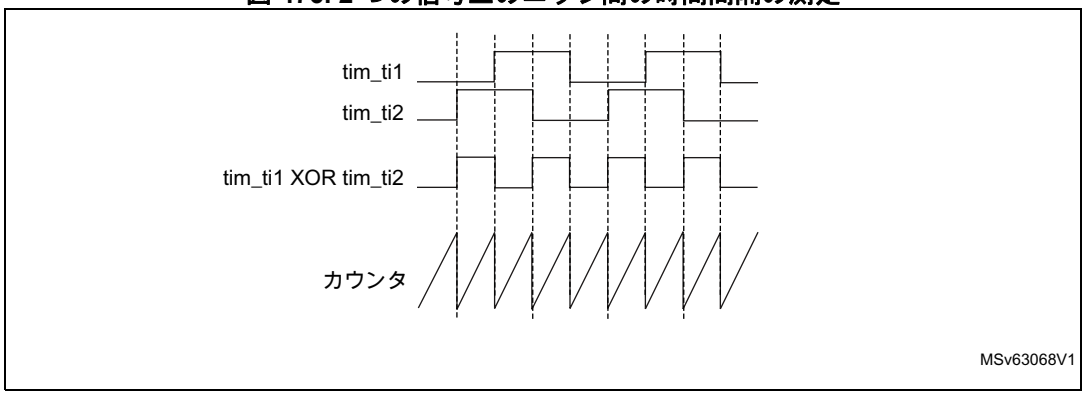
UIF と UIFCPY フラグのアサートの間には、遅延はありません。

30.4.22 タイマ入力 XOR 機能（TIM15 のみ）

TIMx_CR2 レジスタの TI1S ビットによって、チャネル 1 の入力フィルタを 2 つの入力ピン tim_ti1 および tim_ti2 を結合する XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。次の図 476 に示すように、2 つの入力信号上のエッジ間の間隔を測定するのに便利です。

図 476. 2 つの信号上のエッジ間の時間間隔の測定



30.4.23 外部トリガ同期（TIM15 のみ）

TIM タイマは、タイマの同期または連結のために、内部で互いにリンクされます。

TIM15 タイマは、いくつかのモードで外部トリガを使用して同期できます。すなわち、リセットモード、ゲートモード、トリガモード、リセット + トリガ モード、およびゲート + リセットモードです。

スレーブモード：リセットモード

カウンタとそのプリスケーラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ（TIMx_ARR、TIMx_CCRx）が更新されます。

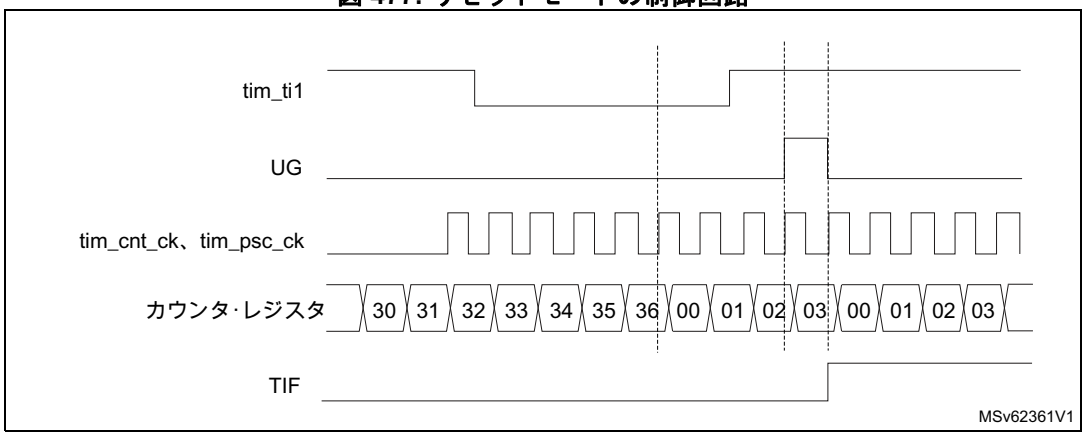
次の例では、tim_ti1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. tim_ti1 の立ち上がりエッジを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01）。TIMx_CCER レジスタで CC1P= 0 と CC1NP = 0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
2. TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、tim_ti1 の立ち上がりエッジまで通常の動作を行います。tim_ti1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ（TIMx_SR レジスタの TIF ビット）、有効な場合は割込みリクエストまたは DMA リクエストを送信できます（TIMx_DIER レジスタの TIE および TDE ビットに依存）。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。tim_ti1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、tim_ti1 入力の同期回路によるものです。

図 477. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

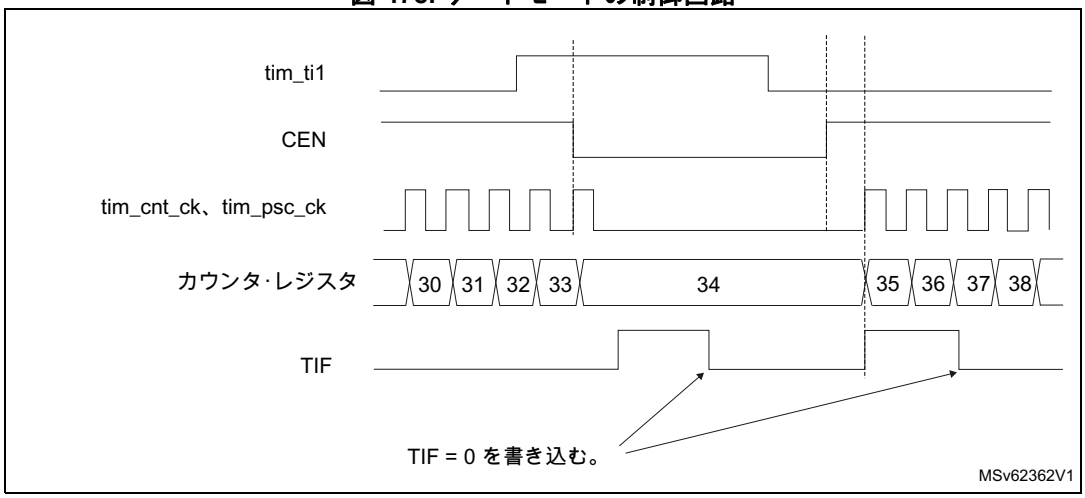
次の例では、アップカウンタは tim_ti1 入力がローのときだけカウントします。

1. tim_ti1 のローレベルを検出するように、チャネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタで CC1P = 1 と CC1NP = 0 を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=00101 を書き込むことによって、入力ソースとして tim_ti1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、tim_ti1 がローになると内部クロックでカウントを開始して、tim_ti1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

tim_ti1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、tim_ti1 入力の再同期回路によるものです。

図 478. ゲートモードの制御回路



スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

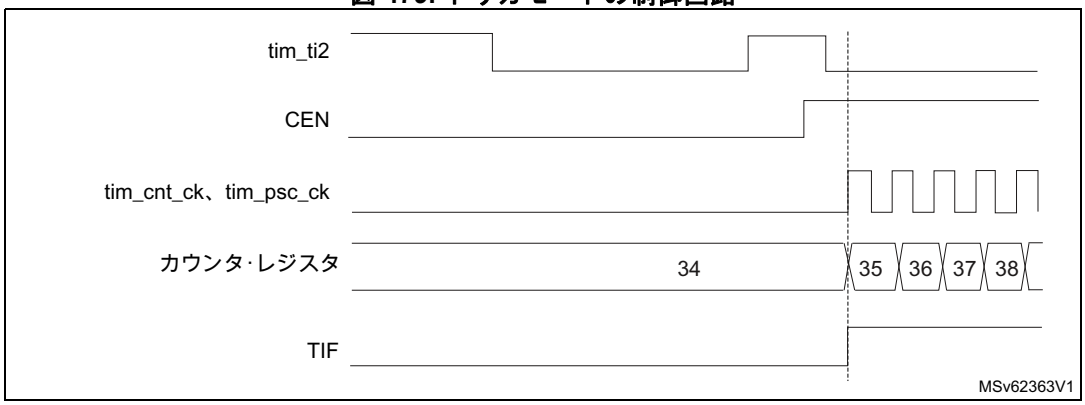
次の例では、アップカウンタは、tim_ti2 入力の立ち上がりエッジに応じて開始します。

1. tim_ti2 の立ち上がりエッジを検出するように、チャネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケーラはトリガには使用されないので、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P に「1」、CC2NP に「0」を書き込んで、極性を有効にします（そして、ローレベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=00110 を書き込むことによって、入力ソースとして tim_ti2 を選択します。

tim_ti2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

tim_ti2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、tim_ti2 入力の再同期回路によるものです。

図 479. トリガモードの制御回路



実行時エンコーダモード更新のためのスレーブモード選択プリロード

SMS[3:0] ビットをプリロードすることができます。これは、TIMx_SMCR レジスタの SMSPE イネーブルビットをセットすることで有効になります。SMS[3:0] プリロードからアクティブ値へ転送するトリガは、カウンタがオーバーフローするときに発生する更新イベント（UEV）です。

30.4.24 スレーブ·モード - リセット·モードとトリガ·モードの組み合わせ（TIM15 のみ）

この場合、選択したトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

このモードはワンパルスモードで使用します。

30.4.25 スレーブモード - リセットモードとゲートモードの組み合わせ（TIM15 のみ）

カウンタクロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

このモードによって、範囲外の PWM 信号（最大期待値を超えるデューティサイクル）を検出できます。

30.4.26 タイマ同期（TIM15 のみ）

タイマの同期やチェーン動作のために、TIMx タイマを内部で互いにリンクできます。詳細については、セクション 29.4.23：タイマの同期を参照してください。

注： **tim_trgo 信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

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

チャネルが 1 つのみのタイマにはマスタ･モードがありません。しかし、OC1 出力信号で、他のいくつかのタイマ（本書の他のセクションで説明しているタイマを含む）をトリガできます。デバイスに搭載されたタイマのうち、トリガを送信してスレーブとして使用できるものを特定するには、「TIMx 内部トリガ接続」にある表を参照してください。

OC1 信号のパルス幅は、トリガ送信先タイマの 2 クロック･サイクル以上にプログラムし、スレーブ･タイマが確実にトリガを検出できるようにする必要があります。

たとえば、トリガ送信先タイマの CK_INT のクロックが、トリガ送信元タイマの 4 分の 1 の速度である場合、OC1 のパルス幅は 8 クロック･サイクルに設定します。

30.4.28 ADC トリガ（TMI15 のみ）

タイマでは、リセット･イベント、イネーブル･イベント、比較イベントなどのさまざまな内部信号で ADC トリガ･イベントを生成できます。

注： **tim_trgo 信号を受信するスレーブ･ペリフェラル（タイマ、ADC など）のクロックは、マスタ･タイマからイベントを受信する前に有効化する必要があり、マスタ･タイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

30.4.29 DMA バースト･モード

TIMx タイマには、1 つのイベントで複数の DMA リクエストを生成する機能があります。主な目的は、いくつかのタイマレジスタをソフトウェアのオーバヘッドなく複数回再プログラムできるようにすることです。複数のレジスタを連続して一定の時間間隔で読み出すために使用することもできます。

DMA コントローラの転送先は一意で、仮想レジスタ TIMx_DMAR を示している必要があります。特定のタイマイベントで、タイマは一連の DMA リクエスト（バースト）を開始します。TIMx_DMAR レジスタへの各書込みは、実際にタイマレジスタの 1 つにリダイレクトされます。

TIMx_DCR レジスタの DBL[4:0] ビットによって、DMA バースト長がセットされます。タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します。つまり、転送数（ハーフワード単位またはバイト単位）を示します。

TIMx_DCR レジスタの DBA[4:0] ビットは、DMA 転送の DMA ベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMCR

たとえば、更新イベント時に CCRx レジスタ値の内容を更新するためにタイマ DMA バースト機能を使用します（x = 2、3、4）。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャネルを次のように設定します。
   - DMA チャネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
   - DMA チャネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
   - 転送データ数 = 3 とします（下の注を参照）。
   - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって､DCR レジスタを設定します。DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします（DIER レジスタのUDE ビットをセット）。
4. TIMx を有効化
5. DMA チャネルを有効化注：

この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

注： **null 値を予約済みレジスタに書き込むことができます。**

30.4.30 TIM15/TIM16/TIM17DMA リクエスト

表 297 に示すように、TIM15/TIM16/TIM17 は DMA リクエストを生成できます。

表 297. DMA リクエスト

DMA リクエスト信号	DMA 略称	DMA リクエスト	イネーブル制御ビット
tim_upd_dma	TIM_UP	更新	UDE
tim_cc1_dma	TIM_CH1	キャプチャ／比較 1	CC1DE
tim_com_dma(1)	TIM_COM	転流（COM）	COMDE
tim_trg_dma（1）	TIM_TRIG	トリガ	TDE

1. TIM15 でのみ使用できます。

30.4.31 デバッグ･モード

マイクロコントローラがデバッグ･モードになると（FPU 搭載 Cortex®-M4 コアは停止状態）、TIMx カウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

安全のため、カウンタが停止すると、出力は無効になります（MOE ビットのリセット時と同じ）。強制的にハイインピーダンスにするために、出力を強制的にインアクティブ状態にするか（OSSI ビット = 1）、GPIO コントローラで制御することができます（OSSI ビット = 0）。

詳細については、デバッグセクションを参照してください。

30.5 TIM15/TIM16/TIM17 低電力モード

表 298. 低消費電力モードが TIM15/TIM16/TIM17 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスは SLEEP モードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

30.6 TIM15/TIM16/TIM17 割込み

表 299 に示すように、TIM15/TIM16/TIM17 は複数の割込みを生成できます。

表 299. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
TIM	更新	UIF	UIE	UIF に 0 を書き込む	可能	不可
	キャプチャ／比較 1	CC1IF	CC1IE	CC1IF に 0 を書き込む	可能	不可
	キャプチャ／比較 2(1)	CC2IF	CC2IE	CC2IF に 0 を書き込む	可能	不可
	転流（COM）	COMIF	COMIE	COMIF に 0 を書き込む	可能	不可
	トリガ (1)	TIF	TIE	TIF に 0 を書き込む	可能	不可
	ブレーク	BIF	BIE	BIF に 0 を書き込む	可能	不可

1. TIM15 でのみ使用できます。

30.7 TIM15 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

30.7.1 TIM15 制御レジスタ 1（TIM15_CR1）

アドレス･オフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 DITHEN：ディザリング･イネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHEN ビットは、CEN ビットがリセットされているときのみ変更できます。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIM15_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIM15_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}= t_{tim_ker_ck}$

01：$t_{DTS}= 2*t_{tim_ker_ck}$

10：$t_{DTS}= 4*t_{tim_ker_ck}$

11：予約済みです。

ビット 7 **ARPE**：自動再ロード･プリロード･イネーブル

0：TIM15_ARR レジスタはバッファされません。

1：TIM15_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルス･モード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエスト･ソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディスエーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 :UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1 :UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタ･イネーブル

0：カウンタは無効です。

1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

30.7.2 TIM15 制御レジスタ 2（TIM15_CR2）

アドレス･オフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
					rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OIS2**：出力アイドル状態 2（tim_oc2 出力）

0：MOE = 0 のとき、tim_oc2=0

1：MOE = 0 のとき、tim_oc2=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BKR レジスタの LOCK ビット）、変更できません。

ビット 9 **OIS1N**：出力アイドル状態 1（tim_oc1n 出力）

0：MOE=0 のとき、デッドタイム後に tim_oc1n=0

1：MOE=0 のとき、デッドタイム後に tim_oc1n=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BKR レジスタの LOCK ビット）、変更できません。

ビット 8 **OIS1**：出力アイドル状態 1（tim_oc1 出力）

0：MOE=0 のとき、デッドタイム後に tim_oc1=0

1：MOE=0 のとき、デッドタイム後に tim_oc1=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BKR レジスタの LOCK ビット）、変更できません。

ビット 7 **TI1S**：tim_ti1 選択

0：tim_ti1_in[15:0] マルチプレクサ出力が tim_ti1 入力に接続されます。

1：tim_ti1_in[15:0] および tim_ti2_in[15:0] マルチプレクサ出力が tim_ti1 入力に接続されます（XOR 接続）。

ビット 6:4 **MMS[2:0]**：マスタ･モード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（tim_trgo）。組み合わせは、次のとおりです。

000： **リセット** - TIM15_EGR レジスタの UG ビットがトリガ出力（tim_trgo）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、tim_trgo 信号は実際のリセットより遅延します。

001： **イネーブル** - カウンタ･イネーブル信号 CNT_EN がトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタ･イネーブル信号は、ゲート･モードに設定されているとき、CEN 制御ビットとトリガ入力との論理積（AND）によって生成されます。カウンタ･イネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブ･モードが選択されている場合を除き、tim_trgo には遅延が存在します（TIM15_SMCR レジスタの MSM ビットの説明を参照してください）。

010： **更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

011： **パルス比較** - CC1IF フラグがすでにハイであっても、それがセットされるキャプチャまたは比較一致が発生すると、トリガ出力として正のパルスがただちに送信されます（tim_trgo）。

100： **比較** - tim_oc1refc 信号がトリガ出力（tim_trgo）として使用されます。

101： **比較** - tim_oc2refc 信号がトリガ出力（tim_trgo）として使用されます。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。

1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。

1：キャプチャ /比較制御ビットがプリロードされているときに（CCPC = 1）、COMG ビットをセットすることによって、または tim_trgi の立ち上がりエッジで更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0：CCxE、CCxNE、および OCxM ビットはプリロードされません。

1：CCxE、CCxNE、および OCxM の各ビットがプリロードされます。書込み後のこれらのビットは、転流イベント（COM）が発生したときにのみ更新されます（CCUS ビットの状態に応じて、COMG ビットのセットまたは tim_trgi の立ち上がりエッジ検出で更新）。

注： このビットは、相補出力を持つチャネルでのみ機能します。

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

アドレス･オフセット：0x08

リセット値：0x0000 0000

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

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 19:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **MSM**：マスタ／スレーブ･モード

0：影響なし。

1：トリガ入力（tim_trgi）に対するイベントの影響は、現在のタイマとそのスレーブとの間で（tim_trgo を通じて）完全な同期状態が得られるまで遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 21、20、6、5、4 **TS[4:0]**：トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

00000：内部トリガ 0（tim_itr0）
00001：内部トリガ 1（tim_itr1）
00010：内部トリガ 2（tim_itr2）
00011：内部トリガ 3（tim_itr3）
00100：tim_ti1 エッジ検出回路（tim_ti1f_ed）
00101：フィルタタイマ入力 1（tim_ti1fp1）
00110：フィルタタイマ入力 2（tim_ti2fp2）
00111：予約済みです。
01000：内部トリガ 4（tim_itr4）
01001：内部トリガ 5（tim_itr5）
01010：内部トリガ 6（tim_itr6）
01011：内部トリガ 7（tim_itr7）
01100：内部トリガ 8（tim_itr8）
01101：内部トリガ 9（tim_itr9）
01110：内部トリガ 10（tim_itr10）
10000：内部トリガ 12（tim_itr12）
10001：内部トリガ 13（tim_itr13）
10010：内部トリガ 14（tim_itr14）
10011：内部トリガ 15（tim_itr15）
その他：予約済みです。

各タイマにおける tim_itrx の意味の詳細については、セクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

注： 設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 16、2、1、0 **SMS[3:0]**：スレーブモード選択

外部信号が選択されると、トリガ信号（tim_trgi）のアクティブエッジが、外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照）。

0000：スレーブモードは無効です。CEN =“1”の場合、プリスケーラは内部クロックによって直接クロック供給されます。

0001：予約済みです。

0010：予約済みです。

0011：予約済みです。

0100：リセット･モード - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

0101：ゲート･モード - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

0110：トリガ･モード - カウンタは、トリガ tim_trgi の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

0111：外部クロック･モード 1 - 選択されたトリガ（tim_trgi）の立ち上がりエッジがカウンタのクロックとして供給されます。

1000：リセット･モードとトリガ･モードの組み合わせ - 選択されたトリガ入力（tim_trgi）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成してカウンタを開始します。

1001：ゲート･モードとリセット･モードの組み合わせ - カウンタ･クロックは、トリガ入力（tim_trgi）がハイのときに有効になります。トリガがローになると、カウンタは停止し、リセットされます。カウンタの開始と停止の両方が制御されます。

その他：予約済み。

注： トリガ入力として tim_ti1f_ed が選択されている場合（TS=00100）、ゲートモードを使用することはできません。tim_ti1f_ed は tim_ti1f の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

注： tim_trgo 信号を受信するスレーブペリフェラル（タイマ、ADC など）のクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。

30.7.4 TIM15 DMA／割込み有効化レジスタ（TIM15_DIER）

アドレス･オフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	rw	rw				rw	rw	rw	rw	rw			rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TDE**：トリガ DMA リクエスト･イネーブル

0：トリガ DMA リクエストは無効です。

1：トリガ DMA リクエストは有効です。

ビット 13 **COMDE**：COM DMA リクエスト･イネーブル

0：COM DMA リクエストは無効です。

1：COM DMA リクエストは有効です。

ビット 12:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル

0：CC1 DMA リクエストは無効です。

1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエスト･イネーブル
0：更新 DMA リクエストは無効です。
1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル
0：ブレーク割込みは無効です。
1：ブレーク割込みは有効です。

ビット 6 **TIE**：トリガ割込みイネーブル
0：トリガ割込みは無効です。
1：トリガ割込みは有効です。

ビット 5 **COMIE**：COM 割込みイネーブル
0：COM 割込み無効
1：COM 割込み有効

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE**：キャプチャ／比較 2 割込みイネーブル
0：CC2 割込みは無効です。
1：CC2 割込みは有効です。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル
0：CC1 割込みは無効です。
1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル
0：更新割込みは無効です。
1：更新割込みは有効です。

30.7.5 TIM15 ステータスレジスタ（TIM15_SR）

アドレス･オフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
					rc_w0	rc_w0		rc_w0	rc_w0	rc_w0			rc_w0	rc_w0	rc_w0

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF**：キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。
0：オーバキャプチャは検出されていません。
1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIM15_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割込みフラグ
このフラグは、ブレーク入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。
0：ブレークイベントは発生していません。
1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 **TIF**：トリガ割込みフラグ

このフラグは、TRG トリガイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、tim_trgi 入力でアクティブエッジが検出されたとき、またはゲートモードが選択されている場合は、両方のエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0：トリガイベントは発生していません。

1：トリガ割込みが保留中です。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IF**：キャプチャ／比較 2 割込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタ·オーバーフロー時（アップカウント·モードおよびアップ／ダウンカウント·モードの場合）、またはアンダーフロー時（ダウンカウント·モードの場合）に CC1IF ビットはハイになります。センターアライン·モードでのフラグの設定には 3 つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0　　の場合の更新）、および TIM15_CR1 レジスタの UDIS=0 の場合。

– TIM15_CR1 レジスタの URS=0 かつ UDIS=0 の場合に、TIM15_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

– TIM15_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって CNT が再初期化されたとき（セクション 30.7.3：TIM15 スレーブモード制御レジスタ（TIM15_SMCR）を参照）。

30.7.6 TIM15 イベント生成レジスタ（TIM15_EGR）

アドレス･オフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
								w	w	rw			w	w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIM15_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 5 **COMG：**キャプチャ／比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2G**：キャプチャ /比較 2 イベント生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIM15_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

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

アドレス･オフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC2F [3:0]				IC2PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 **IC2F[3:0]**：入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]**：入力キャプチャ 2 プリスケーラ

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。

10：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。

11：CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIM15_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIM15_CCER レジスタの CC2E="0"）のときのみ、書込みができます。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、tim_ti1 入力をサンプリングする周波数と、tim_ti1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング

0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6

0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8

0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6

0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8

1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6

1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8

1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5

1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6

1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8

1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5

1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6

1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（tim_ic1）に作用するプリスケーラの分周比を定義します。プリスケーラは、CC1E="0"（TIM15_CCER レジスタ）になるとリセットされます。

00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01：キャプチャは、2 イベントごとに行われます。

10：キャプチャは、4 イベントごとに行われます。

11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

10：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。

11：CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIM15_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注： CC1S ビットは、チャネルがオフ（TIM15_CCER レジスタの CC1E="0"）のときのみ、書込みができます。

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

アドレス･オフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M [3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2 CE	OC2M [2:0]			OC2 PE	OC2 FE	CC2S [1:0]		OC1 CE	OC1M [2:0]			OC1 PE	OC1 FE	CC1S [1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OC2CE**：出力比較 2 クリアイネーブル

ビット 24、14:12 **OC2M[3:0]**：出力比較 2 モード

ビット 11 **OC2PE**：出力比較 2 プリロードイネーブル

ビット 10 **OC2FE**：出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]**：キャプチャ／比較 2 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC2 チャネルは出力として設定されます。

01：CC2 チャネルは入力として設定され、tim_ic2 は tim_ti2 に配置されます。

10：C2 チャネルは入力として設定され、tim_ic2 は tim_ti1 に配置されます。

11：CC2 チャネルは入力として設定され、tim_ic2 は tim_trc にマッピングされます。このモードは、TS ビット（TIM15_SMCR レジスタ）を通じて内部トリガ入力が選択されている場合のみ機能します。

注： CC2S ビットは、チャネルがオフ（TIM15_CCER レジスタの CC2E="0"）のときのみ、書込みができます。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル

0：tim_oc1ref は tim_ocref_clr_int 入力の影響を受けません。

1：tim_oc1ref は tim_ocref_clr_int 入力のハイレベルが検出されるとクリアされます。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、tim_oc1 および tim_oc1n が導き出される出力基準信号 tim_oc1ref の動作を定義します。tim_oc1ref はアクティブハイですが、tim_oc1 および tim_oc1n のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIM15_CCR1 とカウンタ TIM15_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブ･レベルに設定します。カウンタ TIM15_CNT がキャプチャ／比較レジスタ 1（TIM15_CCR1）と一致すると、tim_oc1ref 信号は強制的にハイになります。

0010：一致時にチャネル 1 をインアクティブ･レベルに設定します。カウンタ TIM15_CNT がキャプチャ／比較レジスタ 1（TIM15_CCR1）と一致すると、tim_oc1ref 信号は強制的にローになります。

0011：トグル - TIM15_CNT = TIM15_CCR1 のとき、tim_oc1ref はトグルします。

0100：強制非アクティブレベル - tim_oc1ref は強制的にローになります。

0101：強制アクティブレベル - tim_oc1ref は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIM15_CNT<TIM15_CCR1 の場合はアクティブになり、それ以外の場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIM15_CNT<TIM15_CCR1 の場合はインアクティブに、それ以外の場合はアクティブになります。

1000：再トリガ可能な OPM モード 1 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはインアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。

1001：再トリガ可能な OPM モード 2 - アップカウント･モードでは、tim_trgi 信号でトリガ･イベントを検出するまでチャネルはインアクティブです。その後、PWM モード 2 と同様に比較が行われ、チャネルは次の更新時に再びインアクティブになります。ダウンカウントモードでは、tim_trgi 信号でトリガイベントを検出するまでチャネルはアクティブです。その後、PWM モード 1 と同様に比較が行われ、チャネルは次の更新時に再びアクティブになります。

1010：予約済みです。

1011：予約済みです。

1100：組み合わせ PWM モード 1 - tim_oc1ref は、PWM モード 1 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理和（OR）です。

1101：組み合わせ PWM モード 2 - tim_oc1ref は、PWM モード 2 と同様に動作します。tim_oc1refc は tim_oc1ref と tim_oc2ref の論理積（AND）です。

1110：予約済み。

1111：予約済み。

注：　これらのビットは、LOCK レベル 3 がプログラムされていて（TIM15_BDTR レジスタの LOCK ビット）、CC1S = "00"（チャネルは出力に設定）のときには、変更できません。

PWM モードでは、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、tim_ocxref のレベルが変化します。

相補出力を持つチャネルでは、このビット･フィールドはプリロードされます。TIM15_CR2 レジスタのCCPC ビットがセットされた場合、OC1M アクティブビットは、COM が生成されたときにのみプリロードから新しい値をとります。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIM15_CCR1 のプリロードレジスタは無効です。TIM15_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIM15_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIM15_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注：　これらのビットは、LOCK レベル 3 がプログラムされていて（TIM15_BDTR レジスタの LOCK ビット）、CC1S = "00"（チャネルは出力に設定）のときには、変更できません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1　出力に対して、比較一致のように働きます。このため、tim_ocx は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

10：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti2 に配置されます。

11：CC1 チャネルは入力として設定され、tim_ic1 は tim_trc にマッピングされます。このモードは、TS ビット（TIM15_SMCR レジスタ）で内部トリガ入力が選択されている場合のみ機能します。

注：　CC1S ビットは、チャネルがオフ（TIM15_CCER レジスタの CC1E="0"）のときのみ、書込みができます。

30.7.9 TIM15 キャプチャ／比較有効レジスタ（TIM15_CCER）

アドレス･オフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
								rw		rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP**：キャプチャ／比較 2 相補出力極性

CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P**：キャプチャ／比較 2 出力極性

CC1P の説明を参照してください。

ビット 4 **CC2E**：キャプチャ／比較 2 出力イネーブル

CC1E の説明を参照してください。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：tim_oc1n はアクティブハイです。

1：tim_oc1n はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、tim_ti1fp1 と tim_ti2fp1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注：　このビットは、LOCK レベル 2 または 3 がプログラムされた状態で（TIM15_BDTR レジスタの LOCK ビット）CC1S = “00”（チャネルを出力として設定）になるとただちに書込みできなくなります。

相補出力を持つチャネルでは、このビットはプリロードされます。TIM15_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - tim_oc1n はアクティブではありません。tim_oc1n のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E の各ビットによって決まります。

1：オン - tim_oc1n 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

CC1 チャネルが出力として設定されている場合：

0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）

1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0： 非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1： 反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1： 非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0： この設定は予約済みです。使用できません。

注： このビットは、LOCK レベル 2 または 3 がプログラムされた時点で書込みできなくなります（TIM15_BDTR レジスタの LOCK ビット）。

相補出力を持つチャネルでは、このビットはプリロードされます。TIM15_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 300 を参照してください。

表 300. ブレーク機能を持つ相補 tim_ocx および tim_ocxn チャネルの出力制御ビット（TIM15）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CCxE ビット	CCxNE ビット	tim_ocx 出力状態	tim_ocxn 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0 tim_ocxn=0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocx=0	tim_ocxref + 極性 tim_ocxn = tim_ocxref XOR CCxNP
		0	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref XOR CCxP	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_ocxn=0
		X	1	1	tim_ocxref + 極性 + デッドタイム	tim_ocxref に対する相補（OCREF ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） tim_ocx=CCxP	tim_ocxref + 極性 tim_ocxn=tim_ocxref XOR CCxNP
		1	1	0	tim_ocxref + 極性 tim_ocx=tim_ocxref xor CCxP	オフ状態（インアクティブ状態で出力有効） tim_ocxn=CCxNP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイインピーダンス）	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期に：tim_ocx=CCxP、tim_ocxn=CCxNP クロックが存在する場合：デッドタイム後、tim_ocx=OISx および tim_ocxn=OISxN。ただし、OISx と OISxN は、アクティブ状態における tim_ocx と tim_ocxn の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OISx、OISxN、CCxP、および CCxNP ビットはクリアされたままでなければなりません。

注：　相補チャネル tim_ocx と tim_ocxn に接続されている外部入出力ピンの状態は、tim_ocx チャネルと tim_ocxn チャネルの状態、GPIO の制御レジスタ、およびオルタネート機能選択レジスタに依存します。

30.7.10 TIM15 カウンタ（TIM15_CNT）

アドレス･オフセット：0x24

リセット値：0x0000 0000

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

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

ビット 31 **UIFCPY**：UIF コピー
このビットは TIM15_ISR レジスタの UIF ビットのコピーで読出し専用です。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）
このレジスタは、カウンタ値を保持します。
ディザリングモード（DITHEN = 1）
このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分は得られません。

30.7.11 TIM15 プリスケーラ（TIM15_PSC）

アドレス･オフセット：0x28

リセット値：0x0000

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

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（$f_{tim_cnt_ck}$）は $f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。
PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIM15_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

30.7.12 TIM15 の自動再ロードレジスタ（TIM15_ARR）

アドレス･オフセット：0x2C

リセット値：0x0000 FFFF

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

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

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **ARR[19:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 30.4.3 : 1317 ページのタイムベース･ユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

30.7.13 TIM15 繰り返しカウンタレジスタ（TIM15_RCR）

アドレス･オフセット：0x30

リセット値：0x0000

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

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタの再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰り返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返しカウンタは、繰り返し更新イベント UEV でのみ、REP 値で再ロードされるので、TIM15_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

- エッジアラインモードでは、PWM 周期の数
- センターアラインモードでは、PWM の 1/2 周期の数

30.7.14 TIM15 のキャプチャ／比較レジスタ 1（TIM15_CCR1）

アドレス･オフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR1[19:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIM15_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIM15_CNT と比較され、tim_oc1 出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1 チャネルが入力として設定されている場合：

CR1 は、最後の入力キャプチャ 1 イベント（tim_ic1）によって転送されたカウンタ値です。TIMx_CCR1 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

30.7.15 TIM15 のキャプチャ／比較レジスタ 2（TIM15_CCR2）

アドレス･オフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR2[19:0]**：キャプチャ／比較 2 値

CC2 チャネルが出力として設定されている場合：

CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。

TIM15_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIM15_CNT と比較され、tim_oc2 出力に送信される値を含みます。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] に比較値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] に整数部を保持します。CCR2[3:0] ビットフィールドにはディザリング部分が含まれます。

CC2 チャネルが入力として設定されている場合：

CCR2 は、最後の入力キャプチャ 1 イベント（tim_ic2）によって転送されたカウンタ値です。TIMx_CCR2 レジスタは読出し専用レジスタで、プログラムできません。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR2[15:0] にキャプチャ値を保持します。CCR2[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR2[19:4] にキャプチャを保持します。CCR2[3:0] ビットはリセットされています。

30.7.16 TIM15 ブレークおよびデッドタイムレジスタ（TIM15_BDTR）

アドレス･オフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： **BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書込みがロックされるので、TIM15_BDTR レジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。**

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 tim_brk は入力モードです。

1：ブレーク入力 tim_brk は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 tim_brk が設定されます。

1：ブレーク入力 tim_brk は解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 25:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk 入力信号をサンプリングする周波数と、tim_brk に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、tim_brk は非同期で動作します。

0001：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 2

0010：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 4

0011：$f_{SAMPLING} = f_{tim_ker_ck}$、N = 8

0100：$f_{SAMPLING} = f_{DTS}/2$、N = 6

0101：$f_{SAMPLING} = f_{DTS}/2$、N = 8

0110：$f_{SAMPLING} = f_{DTS}/4$、N = 6

0111：$f_{SAMPLING} = f_{DTS}/4$、N = 8

1000：$f_{SAMPLING} = f_{DTS}/8$、N = 6

1001：$f_{SAMPLING} = f_{DTS}/8$、N = 8

1010：$f_{SAMPLING} = f_{DTS}/16$、N = 5

1011：$f_{SAMPLING} = f_{DTS}/16$、N = 6

1100：$f_{SAMPLING} = f_{DTS}/16$、N = 8

1101：$f_{SAMPLING} = f_{DTS}/32$、N = 5

1110：$f_{SAMPLING} = f_{DTS}/32$、N = 6

1111：$f_{SAMPLING} = f_{DTS}/32$、N = 8

注： このビットは、LOCK レベル 1 がプログラムされているとき（TIM15_BDTR レジスタの LOCK ビット）には変更できません。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、tim_brk 入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OSSI ビットに応じて、tim_ocx および tim_ocxn 出力が無効か、強制的にアイドル状態になります。

1：tim_ocx および tim_ocxn 出力は、それぞれのイネーブルビット（TIM15_CCER レジスタの CCxE、CCxNE ビット）がセットされている場合は有効です。

詳細については、tim_ocx/tim_ocxn イネーブルの説明を参照してください（セクション 30.7.9：1371 ページのTIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。

1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（ブレーク入力がアクティブでない場合）。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 tim_brk はアクティブ･ローです。

1：ブレーク入力 tim_brk はアクティブ･ハイです。

注：　このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロック･サイクルの遅延が生じます。

ビット 12 **BKE**：ブレーク･イネーブル

0：ブレーク入力（tim_brk および tim_sys_brk クロック障害イベント）は無効です。

1：ブレーク入力（tim_brk および tim_sys_brk クロック障害イベント）は有効です。

このビットは、LOCK レベル 1 がプログラムされているとき（TIM15_BDTR レジスタの LOCK ビット）には変更できません。

注：　このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。

詳細については、tim_ocx/tim_ocxn イネーブルの説明を参照してください（セクション 30.7.9：1371 ページのTIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：インアクティブのとき、tim_ocx/tim_ocxn 出力は無効です（タイマが出力の制御を解放し、制御が GPIO に引き継がれることで出力は強制的にハイ･インピーダンスになります）。

1：インアクティブのとき、CCxE = 1 または CCxNE = 1 になると、tim_ocx/tim_ocxn 出力は、インアクティブ･レベルで有効になります（出力は引き続きタイマで制御されます）。

注：　このビットは、LOCK レベル 2 がプログラムされているとき（TIM15_BDTR レジスタの LOCK ビット）には変更できません。

ビット 10 **OSSI**：アイドル･モードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。

詳細については、tim_ocx/tim_ocxn イネーブルの説明を参照してください（セクション 30.7.9：1371 ページのTIM15 キャプチャ／比較有効レジスタ（TIM15_CCER））。

0：インアクティブのとき、tim_ocx/tim_ocxn 出力は無効です（tim_ocx/tim_ocxn イネーブル出力信号 = 0）。

1：インアクティブのとき、tim_ocx/tim_ocxn 出力は、CCxE = 1 または CCxNE = 1 になると、まず強制的にアイドル･レベルになります（tim_ocx/tim_ocxn イネーブル出力信号 = 1）

注：　このビットは、LOCK レベル 2 がプログラムされているとき（TIM15_BDTR レジスタの LOCK ビット）には変更できません。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCK オフ - どのビットも書込み保護されません。

01：LOCK レベル 1 ー TIM15_BDTR レジスタの DTG ビット、TIM15_CR2 レジスタの OISx および OISxN ビット、および TIM15_BDTR レジスタの BKBID/BKE/BKP/AOE ビットは、書込みができなくなります。

10：LOCK レベル 2 ー LOCK レベル 1 で書き込めなくなるビットのほかに、CC 極性ビット（関連するチャネルが CCxS ビットによって出力に設定されている場合は、TIM15_CCER レジスタの CCxP/CCxNP ビット）、OSSR ビット、OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 ー LOCK レベル 2 で書き込めなくなるビットのほかに、CC 制御ビット（関連するチャネルが CCxS ビットによって出力に設定されている場合は、TIM15_CCMRx レジスタの OCxM および OCxPE ビット）も書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIM15_BDTR レジスタに書込みが行われると、それらの内容は次のリセットまで凍結されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。

DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$

DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$

DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$

DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns （250 ns 単位）

32 µs から 63 µs （1 µs 単位）

64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされている（TIM15_BDTR レジスタの LOCK ビット）とき、変更できません。

30.7.17 TIM15 タイマデッドタイムレジスタ 2（TIM15_DTR2）

アドレス･オフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DTPE**：デッドタイム･プリロード･イネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 16 **DTAE**：デッドタイム非対称イネーブル

0：立ち上がりおよび立ち下がりエッジでのデッドタイムは同一で、DTG[7:0] レジスタで定義されます。

1：立ち上がりエッジのデッドタイムは DTG[7:0] レジスタで定義され、立ち下がりエッジのデッドタイムは DTGF[7:0] ビットで定義されます。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジ･ジェネレータのセットアップ

このビットフィールドでは、立ち下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされている（TIM15_BDTR レジスタの LOCK ビット）とき、変更できません。

30.7.18 TIM15 入力選択レジスタ（TIM15_TISEL）

アドレス･オフセット：0x5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **TI2SEL[3:0]**：tim_ti2_in[15:0] の入力を選択します。

0000：TIM15_CH2 入力（tim_ti2_in0）

0001：tim_ti2_in1

.....

1111：tim_ti2_in15

相互接続一覧については、セクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：tim_ti1_in[15:0] の入力を選択します。

0000：TIM15_CH1 入力（tim_ti1_in0）

0001：tim_ti1_in1

.....

1111：tim_ti1_in15

相互接続一覧については、セクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

30.7.19 TIM15 オルタネート機能レジスタ 1 （TIM15_AF1）

アドレス･オフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

製品固有の実装については[セクション 30.4.2: TIM15/TIM16/TIM17 のピンと内部信号]を参照してください。

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **BKCMP4P**：tim_brk_cmp4 入力極性

このビットは、tim_brk_cmp4 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp4 入力はアクティブハイです。

1：tim_brk_cmp4 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 12 **BKCMP3P**：tim_brk_cmp3 入力極性

このビットは、tim_brk_cmp3 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp3 入力はアクティブハイです。

1：tim_brk_cmp3 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 11 **BKCMP2P**：tim_brk_cmp2 入力極性

このビットは、tim_brk_cmp2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2 入力はアクティブハイです。

1：tim_brk_cmp2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：tim_brk_cmp1 入力極性

このビットは、tim_brk_cmp1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1 入力はアクティブハイです。

1：tim_brk_cmp1 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：TIMx_BKIN 入力の極性

このビットは、TIMx_BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN 入力はアクティブ･ハイです。

1：TIMx_BKIN 入力はアクティブ･ローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKCMP8E**：tim_brk_cmp8 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp8 を有効化します。mdf_brkx 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp8 入力は無効です。

1：tim_brk_cmp8 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7 **BKCMP7E**：tim_brk_cmp7 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp7 を有効化します。COMP7 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp7 入力は無効です。

1：tim_brk_cmp7 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 6 **BKCMP6E**：tim_brk_cmp6 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp6 を有効化します。tim_brk_cmp6 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp6 入力は無効です。

1：tim_brk_cmp6 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 5 **BKCMP5E**：tim_brk_cmp5 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp5 を有効化します。tim_brk_cmp5 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp5 入力は無効です。

1：tim_brk_cmp5 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 4 **BKCMP4E**：tim_brk_cmp4 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp4 を有効化します。tim_brk_cmp4 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp4 入力は無効です。

1：tim_brk_cmp4 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 3 **BKCMP3E**：tim_brk_cmp3 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp3 を有効化します。tim_brk_cmp3 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp3 入力は無効です。

1：tim_brk_cmp3 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 2 **BKCMP2E**：tim_brk_cmp2 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp2 を有効化します。tim_brk_cmp2 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp2 入力は無効です。

1：tim_brk_cmp2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：tim_brk_cmp1 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp1 を有効化します。tim_brk_cmp1 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp1 入力は無効です。

1：tim_brk_cmp1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：TIMx_BKIN 入力イネーブル

このビットは、タイマの tim_brk 入力に対して TIMx_BKIN オルタネート機能入力を有効化します。TIMx_BKIN 入力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：TIMx_BKIN 入力は無効です。

1：TIMx_BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

30.7.20 TIM15 オルタネート機能レジスタ 2 （TIM15_AF2）

アドレス･オフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **OCRSEL[2:0]**：ocref_clr ソース選択

これらのビットは、ocref_clr 入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

010：tim_ocref_clr2

011：tim_ocref_clr3

100：tim_ocref_clr4

101：tim_ocref_clr5

110：tim_ocref_clr6

111：tim_ocref_clr7

製品固有の実装についてはセクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIM15_BDTR レジスタの LOCK ビット）、変更できません。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

30.7.21 TIM15 DMA 制御レジスタ（TIM15_DCR）

アドレス･オフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、DMA 転送長（タイマは、TIM15_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
.....
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIM15_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIM15_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIM15_CR1
00001：TIM15_CR2
00010：TIM15_SMCR
.....

30.7.22 完全転送の TIM15 DMA アドレス（TIM15_DMAR）

アドレス･オフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIM15_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIM15_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIM15_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスは DMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL は TIM15_DCR 内で設定）。

30.7.23 TIM15 レジスタマップ

TIM15 レジスタは、次の表のように、16 ビットのアドレス可能レジスタとしてマップされます。

表 301. TIM15 レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIM15_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD [1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	**TIM15_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS2	OIS1N	OIS1	TI1S	MMS[2:0]			CCDS	CCUS	Res.	CCPC
	リセット値																						0	0	0	0	0	0	0	0	0		0
0x08	**TIM15_SMCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TS [4:3]		Res.	Res.	Res.	SMS[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MSM	TS[2:0]			Res.	SMS[2:0]		
	リセット値											0	0				0									0	0	0	0		0	0	0
0x0C	**TIM15_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDE	COMDE	Res.	Res.	Res.	CC1DE	UDE	BIE	TIE	COMIE	Res.	Res.	CC2IE	CC1IE	UIE
	リセット値																		0	0				0	0	0	0	0			0	0	0
0x10	**TIM15_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2OF	CC1OF	Res.	BIF	TIF	COMIF	Res.	Res.	CC2IF	CC1IF	UIF
	リセット値																						0	0		0	0	0			0	0	0
0x14	**TIM15_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	TG	COMG	Res.	Res.	CC2G	CC1G	UG
	リセット値																									0	0	0			0	0	0
0x18	**TIM15_CCMR1** 入力キャプチャ・モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC2F [3:0]				IC2 PSC [1:0]		CC2S [1:0]		IC1F [3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	TIM15_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC2M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	OC2CE	OC2M [2:0]			OC2PE	OC2FE	CC2S [1:0]		OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
	リセット値								0								0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	**TIM15_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2NP	Res.	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
	リセット値																									0		0	0	0	0	0	0
0x24	**TIM15_CNT**	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIM15_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 301. TIM15 レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	**TIM15_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x30	**TIM15_RCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	**TIM15_CCR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38	**TIM15_CCR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38 ～ 0x40	予約済みです。	Res.																															
0x44	**TIM15_BDTR**	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DT[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48 ～ 0x50	予約済みです。	Res.																															
0x54	**TIM15_DTR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x58	予約済みです。	Res.																															
0x5C	**TIM15_TISEL**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI2SEL [3:0]				Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																					0	0	0	0					0	0	0	0
0x60	**TIM15_AF1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE
	リセット値																			0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x64	**TIM15_AF2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCR SEL[2:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x68 - 0x3D8	予約済みです。	Res.																															
0x3DC	**TIM15_DCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x3E0	**TIM15_DMAR**	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

30.8 TIM16/TIM17 レジスタ

レジスタの説明で使用されている略語のリストについては、セクション 1.2 : を参照してください。

30.8.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 16〜17）

アドレス･オフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	CKD[1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw		rw	rw	rw				rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DITHEN**：ディザリング･イネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHEN ビットは、CEN ビットがリセットされているときのみ変更できます。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD[1:0]**：クロック分周

このビットフィールドは、タイマクロック（tim_ker_ck）周波数と、デッドタイムジェネレータとデジタルフィルタ（tim_tix）によって使用されるデッドタイムおよびサンプリングクロック（t_{DTS}）との間の分周比を示します。

00：$t_{DTS}=t_{tim_ker_ck}$

01：$t_{DTS}=2*t_{tim_ker_ck}$

10：$t_{DTS}=4*t_{tim_ker_ck}$

11：予約済みです。

ビット 7 **ARPE**：自動再ロード･プリロード･イネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルス･モード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエスト･ソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

1：カウンタ･オーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディスエーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー
– UG ビットのセット
– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCRx）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタ･イネーブル

0：カウンタは無効です。
1：カウンタは有効です。

注： 外部クロックおよびゲートモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

30.8.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 16～17）

アドレス･オフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **OIS1N**：出力アイドル状態 1（tim_oc1n 出力）

0：MOE=0 のとき、デッドタイム後に tim_oc1n=0
1：MOE=0 のとき、デッドタイム後に tim_oc1n=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKR レジスタの LOCK ビット）。

ビット 8 **OIS1**：出力アイドル状態 1（tim_oc1 出力）

0：MOE=0 のとき、デッドタイム後に tim_oc1=0
1：MOE=0 のとき、デッドタイム後に tim_oc1=1

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BKR レジスタの LOCK ビット）。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CCDS**：キャプチャ／比較 DMA 選択

0：CCx DMA リクエストは、CCx イベントが発生すると送信されます。
1：CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2 **CCUS**：キャプチャ /比較制御更新選択

0：キャプチャ /比較制御ビットがプリロードされるときには（CCPC=1）、COMG ビットをセットすることによってのみ更新されます。
1：キャプチャ /比較制御ビットがプリロードされているときに（CCPC = 1）、COMG ビットをセットすることによって、または tim_trgi の立ち上がりエッジで（使用可能な場合）更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **CCPC**：キャプチャ /比較プリロード制御

0 : CCxE、CCxNE、および OCxM ビットはプリロードされません。

1 : CCxE、CCxNE、および OCxM ビットがプリロードされます。書込みの後、これらのビットは COM ビットがセットされたときにのみ更新されます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

30.8.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 16〜17）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
						rw	rw	rw		rw				rw	rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1DE**：キャプチャ／比較 1 DMA リクエストイネーブル

0：CC1 DMA リクエストは無効です。

1：CC1 DMA リクエストは有効です。

ビット 8 **UDE**：更新 DMA リクエスト･イネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7 **BIE**：ブレーク割込みイネーブル

0：ブレーク割込みは無効です。

1：ブレーク割込みは有効です。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIE**：COM 割込みイネーブル

0：COM 割込み無効

1：COM 割込み有効

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IE**：キャプチャ／比較 1 割込みイネーブル

0：CC1 割込みは無効です。

1：CC1 割込みは有効です。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

30.8.4 TIMx ステータスレジスタ（TIMx_SR）（x = 16～17）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CC1OF**：キャプチャ／比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。これは、“0”を書き込むことによってソフトウェアによってクリアされます。

0：オーバキャプチャは検出されていません。

1：CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BIF**：ブレーク割込みフラグ

このフラグは、tim_brk 入力がアクティブになると、ハードウェアによってセットされます。ブレーク入力がアクティブでない場合、ソフトウェアによってクリアできます。

0：ブレークイベントは発生していません。

1：ブレーク入力でアクティブレベルが検出されました。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMIF**：COM 割込みフラグ

このフラグは、COM イベント時にハードウェアによってセットされます（キャプチャ /比較制御ビット - CCxE、CCxNE、OCxM - が更新されたとき）。ソフトウェアによってクリアされます。

0：COM イベントは発生していません。

1：COM 割込みがペンディング中です。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1IF**：キャプチャ／比較 1 割込みフラグ

このフラグはハードウェアによってセットされます。ソフトウェアによって（入力キャプチャモードまたは出力比較モード）、または TIMx_CCR1 レジスタを読み出すことによって（入力キャプチャモードのみ）クリアされます。

0：比較一致／入力キャプチャは発生していません。

1：比較一致または入力キャプチャが発生しました。

チャネル CC1 が出力として設定されている場合：カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致したとき、このフラグがセットされます。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時（アップカウントおよびアップ／ダウンカウントモードの場合）、またはアンダーフロー時（ダウンカウントモードの場合）に CC1IF ビットはハイになります。センターアラインモードでのフラグの設定には 3つのオプションがあります。詳細については、TIMx_CR1 レジスタの CMS ビットを参照してください。

CC1 チャネルが入力として設定されている場合：このビットは、TIMx_CCR1 レジスタでカウンタ値がキャプチャされた（TIMx_CCER の CC1P および CC1NP ビット設定で定義されたエッジ感度に従って、IC1 でエッジが検出された）ときに設定されます。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

– 繰り返しカウンタ値に関するオーバーフロー（繰り返しカウンタ=0 の場合の更新）、および TIMx_CR1 レジスタの UDIS=0 の場合。

– TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

30.8.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 16～17）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
								w		w				w	w

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **BG**：ブレーク生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：ブレークイベントが生成されます。MOE ビットがクリアされ、BIF フラグがセットされます。有効な場合は、関連する割込みまたは DMA 転送が発生します。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **COMG**：キャプチャ／比較制御更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：CCPC ビットがセットされているときには、CCxE、CCxNE、および OCxM ビットを更新できます。

注： このビットは、相補出力を持つチャネルでのみ機能します。

ビット 4:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1G**：キャプチャ /比較 1 イベント生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：チャネル 1 でキャプチャ／比較イベントが生成されます。

CC1 チャネルが出力として設定されている場合：

CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。

CC1 チャネルが入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響はありません。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

30.8.6 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 16～17）

アドレス･オフセット：0x18

リセット値：0x0000 0000

同じレジスタを入力キャプチャ･モード（このセクション）または出力比較モード（次のセクション）で使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IC1F [3:0]				IC1PSC [1:0]		CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

入力キャプチャモード

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **IC1F[3:0]**：入力キャプチャ 1 フィルタ

このビットフィールドは、tim_ti1 入力をサンプリングする周波数と、tim_ti1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個の連続イベント発生が必要です。

0000：フィルタなし、f_{DTS} でサンプリング
0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2
0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4
0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N =
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}/16$、N = 5
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

ビット 3:2 **IC1PSC[1:0]**：入力キャプチャ 1 プリスケーラ

このビットフィールドは、CC1 入力（tim_ic1）に作用するプリスケーラの分周比を定義します。
プリスケーラは、CC1E = 0（TIMx_CCER レジスタ）になるとリセットされます。
00：プリスケーラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。
01：キャプチャは、2 イベントごとに行われます。
10：キャプチャは、4 イベントごとに行われます。
11：キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

その他：予約済みです。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

30.8.7 TIMx キャプチャ／比較モードレジスタ 1 [オルタネート]（TIMx_CCMR1）（x = 16～17）

アドレス･オフセット：0x18

リセット値：0x0000 0000

同じレジスタを出力比較モード（このセクション）または入力キャプチャ･モード（前のセクション）に使用できます。チャネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力キャプチャモードと出力比較モードで異なる機能を持ちます。両方のモードを個別に組み合わせることができます（例：入力キャプチャモードでのチャネル 1 と出力比較モードでのチャネル 2）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M [3]
															rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1CE	OC1M [2:0]			OC1PE	OC1FE	CC1S [1:0]	
								rw	rw	rw	rw	rw	rw	rw	rw

出力比較モード：

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **OC1CE**：出力比較 1 クリアイネーブル

0：tim_oc1ref は tim_ocref_clr 入力の影響を受けません。

1：tim_oc1ref は tim_ocref_clr 入力のハイレベルが検出されるとクリアされます。

ビット 16、6:4 **OC1M[3:0]**：出力比較 1 モード

これらのビットは、tim_oc1 および tim_oc1n が導き出される出力基準信号 tim_oc1ref の動作を定義します。tim_oc1ref はアクティブハイですが、tim_oc1 および tim_oc1n のアクティブレベルは CC1P および CC1NP ビットに依存します。

0000：停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません。

0001：一致時にチャネル 1 をアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にハイになります。

0010：比較が一致した場合、チャネル 1 をインアクティブ･レベルに設定します。カウンタ TIMx_CNT がキャプチャ /比較レジスタ 1（TIMx_CCR1）と一致すると、tim_oc1ref 信号は強制的にローになります。

0011：反転 - TIMx_CNT = TIMx_CCR1 のとき、tim_oc1ref は反転します。

0100：強制非アクティブレベル - tim_oc1ref は強制的にローになります。

0101：強制アクティブレベル - tim_oc1ref は強制的にハイになります。

0110：PWM モード 1 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

0111：PWM モード 2 - チャネル 1 は、TIMx_CNT<TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。

その他：予約済みです。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、tim_oc1ref のレベルが変化します。

ビット 3 **OC1PE**：出力比較 1 プリロードイネーブル

0：TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書込み可能であり、新しい値はただちに有効になります。

1：TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注： これらのビットは、LOCK レベル 3 がプログラムされていて（TIMx_BDTR レジスタの LOCK ビット）、かつ CC1S=“00”（チャネルは出力に設定）のときには、変更できません。

ビット 2 **OC1FE**：出力比較 1 高速イネーブル

このビットは、トリガイベントとタイマ出力の遷移の間の待ち時間を短縮します。開始トリガの後、できるだけ早く出力パルスを開始するには、1 パルスモード（TIMx_CR1 レジスタの OPM ビットのセット）で使用する必要があります。

0：CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1：トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、tim_ocx は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 **CC1S[1:0]**：キャプチャ／比較 1 選択

このビットフィールドは、チャネルの方向（入力／出力）と、使用される入力を定義します。

00：CC1 チャネルは出力として設定されます。

01：CC1 チャネルは入力として設定され、tim_ic1 は tim_ti1 に配置されます。

その他：予約済みです。

注： CC1S ビットは、チャネルがオフ（TIMx_CCER レジスタの CC1E=0）のときにのみ書込み可能です。

30.8.8 TIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17）

アドレスオフセット：0x20

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **CC1NP**：キャプチャ／比較 1 相補出力極性

CC1 チャネルが出力として設定されている場合：

0：tim_oc1n はアクティブハイです。

1：tim_oc1n はアクティブローです。

CC1 チャネルが入力として設定されている場合：

このビットは、tim_ti1fp1 の極性を定義するために CC1P と組み合わせて使用されます。CC1P の説明を参照してください。

注： このビットは、LOCK レベル 2 または 3 がプログラムされた状態で（TIMx_BDTR レジスタの LOCK ビット）、CC1S = “00”（チャネルは出力として設定）になるとただちに書込みできなくなります。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1NP アクティブビットは、転流イベント（COM）が発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 2 **CC1NE**：キャプチャ /比較 1 相補出力イネーブル

0：オフ - tim_oc1n はアクティブではありません。tim_oc1n のレベルは、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E の各ビットによって決まります。

1：オン - tim_oc1n 信号は、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1E ビットにより、対応する出力ピンに出力されます。

ビット 1 **CC1P**：キャプチャ／比較 1 出力極性

0：OC1 のアクティブハイ（出力モード）／エッジ感度を選択（入力モード、下記を参照）

1：OC1 のアクティブロー（出力モード）／エッジ感度を選択（入力モード、下記を参照）

CC1 チャネルが入力として設定されているときは、CC1NP/CC1P の両ビットは、トリガまたはキャプチャ動作に対する TI1FP1 と TI2FP1 のアクティブ極性を選択します。

CC1NP=0、CC1P=0：非反転/立ち上がりエッジ。この回路は TIxFP1 の立ち上がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=0、CC1P=1：反転／立ち下がりエッジ。この回路は TIxFP1 の立ち下がりエッジに反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されます（ゲートモードまたはエンコーダモードでのトリガ動作）。

CC1NP=1、CC1P=1：非反転／両エッジ／この回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し（リセットモード、外部クロックモード、またはトリガモードにおけるキャプチャまたはトリガ動作）、TIxFP1 は反転されません（ゲートモードでのトリガ動作）。この設定をエンコーダモードで使用することはできません。

CC1NP=1、CC1P=0：この設定は予約済みです。使用できません。

注：　このビットは、LOCK レベル 2 または 3 がプログラムされた直後は書込みできません（TIMx_BDTR レジスタの LOCK ビット）。

相補出力を持つチャネルでは、このビットはプリロードされます。TIMx_CR2 レジスタの CCPC ビットがセットされた場合、CC1P アクティブビットは、転流イベントが発生したときにのみ、プリロードされたビットから新しい値を取り込みます。

ビット 0 **CC1E**：キャプチャ／比較 1 出力イネーブル

0：キャプチャモードは無効／OC1 は非アクティブ（下記を参照）。

1：キャプチャモードは有効／OC1 信号は、対応する出力ピンに出力されます。

CC1 チャネルが出力として設定されている場合、OC1 レベルは、CC1E ビットの状態にかかわらず、MOE、OSSI、OSSR、OIS1、OIS1N、および CC1NE ビットに依存します。詳細については、表 302 を参照してください。

表 302. ブレーク機能を持つ相補 tim_oc1 および tim_oc1n チャネルの出力制御ビット機能（TIM16/TIM17）

制御ビット					出力状態(1)	
MOE ビット	OSSI ビット	OSSR ビット	CC1E ビット	CC1NE ビット	tim_oc1 出力状態	tim_oc1n 出力状態
1	X	X	0	0	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1 = 0 tim_oc1n = 0	
		0	0	1	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1 = 0	tim_oc1ref + 極性 tim_oc1n = tim_oc1ref XOR CC1NP
		0	1	0	tim_oc1ref + 極性 tim_oc1=tim_oc1ref XOR CC1P	出力無効（タイマによって駆動されない：ハイインピーダンス） tim_oc1n = 0
		X	1	1	tim_oc1ref + 極性 + デッドタイム	tim_oc1ref に対する相補（tim_oc1ref ではなく）+ 極性 + デッドタイム
		1	0	1	オフ状態（インアクティブ状態で出力有効） tim_oc1=CC1P	tim_oc1ref + 極性 tim_oc1n=tim_oc1ref XOR CC1NP
		1	1	0	tim_oc1ref + 極性 tim_oc1=tim_oc1ref XOR CC1P	オフ状態（インアクティブ状態で出力有効） tim_oc1n=CC1NP
0	0	X	X	X	出力無効（タイマによって駆動されない：ハイインピーダンス）	
	1		0	0		
			0	1	オフ状態（インアクティブ状態で出力有効） 非同期に：tim_oc1=CC1P、tim_oc1n=CC1NP クロックが存在する場合：デッドタイム後、tim_oc1=OIS1 および tim_oc1n=OIS1N。ただし、OIS1 と OIS1N は、アクティブ状態における tim_oc1 と tim_oc1n の両方に対応しないことを前提とします。	
			1	0		
			1	1		

1. チャネルの両方の出力が使用されないとき（GPIO コントローラが制御を引き継いだ場合）、OIS1、OIS1N、CC1P、および CC1NP ビットはクリアされたままでなければなりません。

注： 相補チャネル tim_oc1 および tim_oc1n に接続されている外部入出力ピンの状態は、tim_oc1 および tim_oc1n チャネルの状態と、GPIO の制御レジスタおよびオルタネート機能選択レジスタに依存します。

30.8.9 TIMx カウンタ（TIMx_CNT）（x = 16～17）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされている場合、ビット 31 は予約済みです。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）

このレジスタは、カウンタ値を保持します。

ディザリングモード（DITHEN = 1）

このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分は得られません。

30.8.10 TIMx プリスケーラ（TIMx_PSC）（x = 16～17）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数（ftim_cnt_ck）は $f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。

PSC は、更新イベントごとにアクティブプリスケーラレジスタにロードされる値を含みます（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、またはリセットモードに設定されているトリガコントローラを通じて、カウンタがクリアされる場合も含まれます）。

30.8.11 TIMx 自動再ロードレジスタ（TIMx_ARR）（x = 16～17）

アドレスオフセット：0x2C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **ARR[19:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARR の更新と動作の詳細については、セクション 30.4.3 : 1317 ページのタイムベース･ユニットを参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

非ディザリングモード（DITHEN = 0）

このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

30.8.12 TIMx 繰り返しカウンタレジスタ（TIMx_RCR）（x = 16～17）

アドレスオフセット：0x30

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **REP[7:0]**：繰り返しカウンタの再ロード値

このビットフィールドによって、プリロードレジスタが有効なときの比較レジスタの更新レート（プリロードレジスタからアクティブレジスタへの周期的な転送）を定義します。また、更新割込み生成の頻度も定義します（この割込みが有効な場合）。

繰り返しダウンカウンタがゼロに達すると、更新イベントが生成され、REP 値からカウントをリスタートします。繰り返しカウンタは、繰り返し更新イベント UEV でのみ、REP 値で再ロードされるので、TIMx_RCR レジスタへの書込みは、次の繰り返し更新イベントまで有効になりません。

PWM モードでは、（REP+1）はエッジアラインモードで PWM 周期の数を意味します。

- エッジアラインモードでは、PWM 周期の数
- センターアラインモードでは、PWM の 1/2 周期の数

30.8.13 TIMx キャプチャ／比較レジスタ 1（TIMx_CCR1）（x = 16～17）

アドレス･オフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **CCR1[19:0]**：キャプチャ／比較 1 値

CC1 チャネルが出力として設定されている場合：

CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、tim_oc1 出力に送信される値を含みます。

非ディザリング･モード（DITHEN = 0）

このレジスタは、CCR1[15:0] に比較値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] に整数部を保持します。CCR1[3:0] ビットフィールドにはディザリング部分が含まれます。

CC1 チャネルが入力として設定されている場合：

CCR1 は、最後の入力キャプチャ 1 イベント（tim_ic1）によって転送されたカウンタ値です。

非ディザリングモード（DITHEN = 0）

このレジスタは、CCR1[15:0] にキャプチャ値を保持します。CCR1[19:16] ビットはリセットされています。

ディザリングモード（DITHEN = 1）

このレジスタは、CCR1[19:4] にキャプチャを保持します。CCR1[3:0] ビットはリセットされています。

30.8.14 TIMx ブレークおよびデッドタイムレジスタ（TIMx_BDTR）（x = 16～17）

アドレスオフセット：0x44

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	BKBID	Res.	BK DSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]			
			rw		rw							rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

注： BKBID、BKDSRM、BKF[3:0]、AOE、BKP、BKE、OSSI、OSSR、および DTG[7:0] ビットは、LOCK 設定に応じて書込みがロックされるので、TIMx_BDTR レジスタへの最初の書込みアクセス時に、これらすべてを設定しなければならないことがあります。

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28 **BKBID**：ブレーク双方向

0：ブレーク入力 tim_brk は入力モードです。

1：ブレーク入力 tim_brk は双方向モードです。

双方向モード（BKBID ビットが 1 にセット）では、ブレーク入力が入力モードとオープンドレイン出力モード両方で設定されます。アクティブなブレークイベントで、ブレーク入力の低ロジックレベルをアサートし、外部デバイスに対する内部ブレークイベントを表します。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **BKDSRM**：ブレーク解除

0：ブレーク入力 tim_brk が設定されます。

1：ブレーク入力 tim_brk は解除されます。

このビットは、アクティブなブレークソースがない場合、ハードウェアによってクリアされます。

BKDSRM ビットは、双方向出力制御（ハイインピーダンス状態でのオープンドレイン出力）を解放するためにソフトウェアでセットしてから、ハードウェアによってリセットされ、障害状態がなくなったことを示すまでポーリングする必要があります

注： このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 25:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **BKF[3:0]**：ブレークフィルタ

このビットフィールドは、tim_brk 入力をサンプリングする周波数と、tim_brk に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000：フィルタなし、tim_brk は非同期で動作します。
0001：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 2
0010：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 4
0011：$f_{SAMPLING}= f_{tim_ker_ck}$、N = 8
0100：$f_{SAMPLING}= f_{DTS}/2$、N = 6
0101：$f_{SAMPLING}= f_{DTS}/2$、N = 8
0110：$f_{SAMPLING}= f_{DTS}/4$、N = 6
0111：$f_{SAMPLING}= f_{DTS}/4$、N = 8
1000：$f_{SAMPLING}= f_{DTS}/8$、N = 6
1001：$f_{SAMPLING}= f_{DTS}/8$、N = 8
1010：$f_{SAMPLING}= f_{DTS}$、N = 16
1011：$f_{SAMPLING}= f_{DTS}/16$、N = 6
1100：$f_{SAMPLING}= f_{DTS}/16$、N = 8
1101：$f_{SAMPLING}= f_{DTS}/32$、N = 5
1110：$f_{SAMPLING}= f_{DTS}/32$、N = 6
1111：$f_{SAMPLING}= f_{DTS}/32$、N = 8

このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15 **MOE**：メイン出力イネーブル

このビットは、tim_brk 入力がアクティブとなると、ハードウェアによって非同期にクリアされます。ソフトウェアによって、または、AOE ビットに応じて自動的にセットされます。出力として設定されたチャネルに対してのみ有効です。

0：OSSI ビットに応じて、tim_oc1 および tim_oc1n 出力が無効か、強制的にアイドル状態になります。
1：tim_oc1 および tim_oc1n 出力は、それぞれのイネーブルビット（TIMx_CCER レジスタの CC1E、CC1NE ビット）がセットされている場合は有効です。

詳細については、tim_oc1/tim_oc1n イネーブルの説明を参照してください（セクション 30.8.8：1396 ページのTIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

ビット 14 **AOE**：自動出力イネーブル

0：MOE はソフトウェアによってのみセットできます。
1：MOE は、ソフトウェアによって、または次の更新イベント時に自動的にセットできます（tim_brk 入力がアクティブでない場合）。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 13 **BKP**：ブレーク極性

0：ブレーク入力 tim_brk はアクティブ･ローです。
1：ブレーク入力 tim_brk はアクティブ･ハイです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 12 **BKE**：ブレーク･イネーブル

0：ブレーク入力（tim_brk および tim_sys_brk イベント）は無効です。
1：ブレーク入力（tim_brk および tim_sys_brk イベント）は有効です。

注： このビットは、LOCK レベル 1 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

このビットへの書込み操作では、書込みが有効になるまでに 1 APB クロックサイクルの遅延が生じます。

ビット 11 **OSSR**：RUN モードのオフ状態の選択

このビットは、MOE=1 のとき、相補出力を持ち、出力として設定されているチャネルで使用されます。OSSR は、相補出力がタイマに実装されていない場合には、実装されません。
詳細については、tim_oc1/tim_oc1n イネーブルの説明を参照してください（セクション 30.8.8：1396 ページのTIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

0：インアクティブのとき、tim_oc1/tim_oc1n 出力は無効です（タイマが出力の制御を解放し、制御が GPIO に引き継がれることで出力は強制的にハイ･インピーダンスになります）。

1：インアクティブのとき、CC1E = 1 または CC1NE = 1 になると、tim_oc1/tim_oc1n 出力は、インアクティブ･レベルで有効になります（出力は引き続きタイマで制御されます）。

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 10 **OSSI**：アイドル･モードのオフ状態の選択

このビットは、MOE=0 のとき、出力として設定されているチャネルで使用されます。
詳細については、tim_oc1/tim_oc1n イネーブルの説明を参照してください（セクション 30.8.8：1396 ページのTIMx キャプチャ／比較有効レジスタ（TIMx_CCER）（x = 16～17））。

0：インアクティブのとき、tim_oc1/tim_oc1n 出力は無効です（tim_oc1/tim_oc1n イネーブル出力信号 = 0）。

1：インアクティブのとき、tim_ocx/tim_ocxn 出力は、CC1E = 1 または CC1NE = 1 になると、まず強制的にアイドル･レベルになります（tim_oc1/tim_oc1n イネーブル出力信号 = 1）

注： このビットは、LOCK レベル 2 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 9:8 **LOCK[1:0]**：ロック設定

これらのビットは、ソフトウェアエラーに対する書込み保護を提供します。

00：LOCK オフ - どのビットも書込み保護されません。

01：LOCK レベル 1 ＝ TIMx_BDTR レジスタの DTG ビット、TIMx_CR2 レジスタの OISx および OISxN ビット、および TIMx_BDTR レジスタの BKBID/BKE/BKP/AOE ビットは、書込みができなくなります。

10：LOCK レベル 2 ＝ LOCK レベル 1 に加えて、CC 極性ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCER レジスタの CCxP/CCxNP ビット）と OSSR および OSSI ビットも書き込めなくなります。

11：LOCK レベル 3 ＝ LOCK レベル 2 に加えて、CC 制御ビット（関連するチャネルが CCxS ビットを通じて出力に設定されている場合は、TIMx_CCMRx レジスタの OCxM および OCxPE ビット）が書き込めなくなります。

注： LOCK ビットは、リセット後に一度だけ書込みができます。いったん TIMx_BDTR レジスタに書込みが行われると、その内容は次のリセットまで停止されます。

ビット 7:0 **DTG[7:0]**：デッドタイムジェネレータのセットアップ

これらのビットでは、相補出力の間に挿入されるデッドタイムの長さを指定します。デッドタイムの時間（DT）は、次の式で与えられます。
DTG[7:5]=0xx => DT=DTG[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$
DTG[7:5]=10x => DT=(64+DTG[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$
DTG[7:5]=110 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$
DTG[7:5]=111 => DT=(32+DTG[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$
例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。
0 から 15875 ns（125 ns 単位）
16 µs から 31750 ns （250 ns 単位）
32 µs から 63 µs （1 µs 単位）
64 µs から 126 µs （2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

30.8.15 TIMx タイマ･デッドタイム･レジスタ 2 （TIMx_DTR2）（x = 16～17）

アドレス･オフセット：0x054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DTPE**：デッドタイム･プリロード･イネーブル

0：デッドタイム値はプリロードされません。

1：デッドタイム値プリロードは有効です。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 16 **DTAE**：デッドタイム非対称イネーブル

0：立ち上がりおよび立ち下がりエッジでのデッドタイムは同一で、DTG[7:0] レジスタで定義されます。

1：立ち上がりエッジのデッドタイムは DTG[7:0] レジスタで定義され、立ち下がりエッジのデッドタイムは DTGF[7:0] ビットで定義されます。

注： このビットは、LOCK レベル 1、2、または 3 がプログラムされているときには変更できません（TIMx_BDTR レジスタの LOCK ビット）。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DTGF[7:0]**：デッドタイム立ち下がりエッジ･ジェネレータのセットアップ

このビットフィールドでは、立ち下がりエッジにおいて、相補出力の間に挿入されるデッドタイムの長さを指定します。

DTGF[7:5]=0xx => DTF=DTGF[7:0]x t_{dtg}、ここで $t_{dtg}=t_{DTS}$.

DTGF[7:5]=10x => DTF=(64+DTGF[5:0])xt_{dtg}、ここで $T_{dtg}=2xt_{DTS}$.

DTGF[7:5]=110 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=8xt_{DTS}$.

DTGF[7:5]=111 => DTF=(32+DTGF[4:0])xt_{dtg}、ここで $T_{dtg}=16xt_{DTS}$.

例：T_{DTS}=125ns（8MHz）の場合、可能なデッドタイムの値は、以下のとおりです。

0 から 15875 ns（125 ns 単位）

16 µs から 31750 ns（250 ns 単位）

32 µs から 63 µs（1 µs 単位）

64 µs から 126 µs（2 µs 単位）

注： このビットフィールドは、LOCK レベル 1、2、または 3 がプログラムされているとき、変更できません（TIMx_BDTR レジスタの LOCK ビット）。

30.8.16 TIMx 入力選択レジスタ（TIMx_TISEL）（x = 16～17）

アドレスオフセット：0x5C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **TI1SEL[3:0]**：tim_ti1_in[15:0] の入力を選択します。

0000：TIMx_CH1 入力（tim_ti1_in0）

0001：tim_ti1_in1

.....

1111：tim_ti1_in15

相互接続一覧については、セクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

30.8.17 TIMx オルタネート機能レジスタ 1 （TIMx_AF1）（x = 16～17）

アドレス･オフセット：0x060

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	BK CMP4P	BK CMP3P	BK CMP2P	BK CMP1P	BKINP	BK CMP8E	BK CMP7E	BK CMP6E	BK CMP5E	BK CMP4E	BK CMP3E	BK CMP2E	BK CMP1E	BKINE
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

製品固有の実装についてはセクション 30.4.2:TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **BKCMP4P**：tim_brk_cmp4 入力極性

このビットは、tim_brk_cmp4 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp4 入力はアクティブハイです。

1：tim_brk_cmp4 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 12 **BKCMP3P**：tim_brk_cmp3 入力極性

このビットは、tim_brk_cmp3 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp3 入力はアクティブハイです。

1：tim_brk_cmp3 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 11 **BKCMP2P**：tim_brk_cmp2 入力極性

このビットは、tim_brk_cmp2 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp2 入力はアクティブハイです。

1：tim_brk_cmp2 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 10 **BKCMP1P**：tim_brk_cmp1 入力極性

このビットは、tim_brk_cmp1 入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：tim_brk_cmp1 入力はアクティブハイです。

1：tim_brk_cmp1 入力はアクティブローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 9 **BKINP**：TIMx_BKIN 入力の極性

このビットは、TIMx_BKIN オルタネート機能入力の感度を選択します。BKP 極性ビットとともにプログラムする必要があります。

0：TIMx_BKIN 入力はアクティブ･ハイです。

1：TIMx_BKIN 入力はアクティブ･ローです。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 8 **BKCMP8E**：tim_brk_cmp8 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp7 を有効化します。mdf_brkx 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp8 入力は無効です。

1：tim_brk_cmp8 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 7 **BKCMP7E**：tim_brk_cmp7 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp7 を有効化します。tim_brk_cmp7 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp7 入力は無効です。

1：tim_brk_cmp7 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 6 **BKCMP6E**：tim_brk_cmp6 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp6 を有効化します。tim_brk_cmp6 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp6 入力は無効です。

1：tim_brk_cmp6 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 5 **BKCMP5E**：tim_brk_cmp5 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp5 を有効化します。tim_brk_cmp5 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp5 入力は無効です。

1：tim_brk_cmp5 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 4 **BKCMP4E**：tim_brk_cmp4 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp4 を有効化します。tim_brk_cmp4 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp4 入力は無効です。

1：tim_brk_cmp4 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 3 **BKCMP3E**：tim_brk_cmp3 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp3 を有効化します。tim_brk_cmp3 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp3 入力は無効です。

1：tim_brk_cmp3 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 2 **BKCMP2E**：tim_brk_cmp2 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp2 を有効化します。tim_brk_cmp2 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp2 入力は無効です。

1：tim_brk_cmp2 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 1 **BKCMP1E**：tim_brk_cmp1 イネーブル

このビットは、タイマの tim_brk 入力に対して tim_brk_cmp1 を有効化します。tim_brk_cmp1 出力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：tim_brk_cmp1 入力は無効です。

1：tim_brk_cmp1 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 0 **BKINE**：TIMx_BKIN 入力イネーブル

このビットは、タイマの tim_brk 入力に対して TIMx_BKIN オルタネート機能入力を有効化します。TIMx_BKIN 入力は、ほかの tim_brk ソースとの「論理和」がとられます。

0：TIMx_BKIN 入力は無効です。

1：TIMx_BKIN 入力は有効です。

注： このビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

30.8.18 TIMx オルタネート機能レジスタ 2 （TIMx_AF2）（x = 16～17）

アドレス･オフセット：0x064

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCRSEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **OCRSEL[2:0]**：tim_ocref_clr ソース選択

これらのビットは、tim_ocref_clr 入力ソースを選択します。

000：tim_ocref_clr0

001：tim_ocref_clr1

010：tim_ocref_clr2

011：tim_ocref_clr3

100：tim_ocref_clr4

101：tim_ocref_clr5

110：tim_ocref_clr6

111：tim_ocref_clr7

製品固有の実装についてはセクション 30.4.2：TIM15/TIM16/TIM17 のピンと内部信号を参照してください。

注： これらのビットは、LOCK レベル 1 がプログラムされている場合（TIMx_BDTR レジスタの LOCK ビット）、変更できません。

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

30.8.19 TIMx オプション･レジスタ 1（TIMx_OR1）（x = 16～17）

アドレス･オフセット：0x68

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSE32EN
															rw

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **HSE32EN**：32 分周された HSE イネーブル

このビットは、tim_ti1_in3 用に 32 分周された HSE を有効にします。詳細については、表 289：tim_ti1 入力マルチプレクサへの相互接続を参照してください。

0：32 分周された HSE は無効です。

1：32 分周された HSE は有効です。

30.8.20 TIMx DMA 制御レジスタ（TIMx_DCR）（x = 16～17）

アドレスオフセット：0x3DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

ビット 31:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのフィールドは、転送数など DMA 転送長（タイマは、TIMx_DMAR アドレスに対して読出しまたは書込みアクセスが行われるときにバースト転送を認識します）を指定します。転送は、ハーフワードまたはバイトです（以下の例を参照）。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
.....
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのフィールドは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読出し／書込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
.....

例：次の転送を考えます：DBL = 7 転送 かつDBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

30.8.21 TIM16/TIM17 完全転送用の DMA アドレス（TIMx_DMAR）（x = 16～17）

アドレスオフセット：0x3E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DMAB[31:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読出しまたは書込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

（TIMx_CR1 アドレス）+ （DBA + DMA インデックス）x 4

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスはDMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL はTIMx_DCR 内で設定）。

30.8.22 TIM16/TIM17 レジスタマップ

TIM16/TIM17 レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 303. TIM16/TIM17レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIFREMAP	Res.	CKD [1:0]		ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																					0		0	0	0				0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OIS1N	OIS1	Res.	Res.	Res.	Res.	CCDS	CCUS	Res.	CCPC
	リセット値																							0	0					0	0		0
0x08	予約済みです。	Res.																															
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1DE	UDE	BIE	Res.	COMIE	Res.	Res.	Res.	CC1IE	UIE
	リセット値																							0	0	0		0				0	0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC1OF	Res.	BIF	Res.	COMIF	Res.	Res.	Res.	CC1IF	UIF
	リセット値																							0		0		0				0	0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	BG	Res.	COMG	Res.	Res.	Res.	CC1G	UG
	リセット値																									0		0				0	0
0x18	**TIMx_CCMR1** 入力キャプチャ・モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	IC1F [3:0]				IC1 PSC [1:0]		CC1S [1:0]	
	リセット値																									0	0	0	0	0	0	0	0
	TIMx_CCMR1 出力比較モード	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OC1M[3]	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	OC1CE	OC1M [2:0]				OC1PE	OC1FE	CC1S [1:0]	
	リセット値																0									0	0	0	0	0	0	0	0
0x1C	予約済みです。	Res.																															
0x20	**TIMx_CCER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	Res.	CC1NP	CC1NE	CC1P	CC1E
	リセット値																													0	0	0	0
0x24	**TIMx_CNT**	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

表 303. TIM16/TIM17レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x30	TIMx_RCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REP[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x34	TIMx_CCR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CCR1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x38 - 0x40	予約済みです。	Res.																															
0x44	TIMx_BDTR	Res.	Res.	Res.	BKBID	Res.	BKDSRM	Res.	Res.	Res.	Res.	Res.	Res.	BKF[3:0]				MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK [1:0]		DT[7:0]							
	リセット値				0		0							0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48 ~ 0x50	予約済みです。	Res.																															
0x54	TIMx_DTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTPE	DTAE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DTGF[7:0]							
	リセット値															0	0									0	0	0	0	0	0	0	0
0x58	予約済みです。	Res.																															
0x5C	TIMx_TISEL	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TI1SEL [3:0]			
	リセット値																													0	0	0	0
0x60	TIMx_AF1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BKCMP4P	BKCMP3P	BKCMP2P	BKCMP1P	BKINP	BKCMP8E	BKCMP7E	BKCMP6E	BKCMP5E	BKCMP4E	BKCMP3E	BKCMP2E	BKCMP1E	BKINE	
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	1	
0x64	TIMx_AF2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OCR SEL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値														0	0	0																
0x68	TIMx_OR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	HSE32EN
	リセット値																																0
0x6C ~ 0x3D8	予約済みです。	Res.																															
0x3DC	TIMx_DCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBL[4:0]					Res.	Res.	Res.	DBA[4:0]				
	リセット値																				0	0	0	0	0				0	0	0	0	0
0x3E0	TIMx_DMAR	DMAB[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、セクション 2.2 : を参照してください。

31 基本タイマ （TIM6/TIM7）

31.1 TIM6/TIM7 の概要

基本タイマ（TIM6/TIM7）は、プログラム可能なプリスケーラによって駆動される 16 ビットの自動再ロード･カウンタで構成されています。

タイムベースを生成する汎用タイマとして使用できます。

また、DACのトリガとして使用することもできます。その場合は、タイマのトリガ出力を使用します。

タイマは全面的に独立していて、どのようなリソースも共有しません。

31.2 TIM6/TIM7 の主な特徴

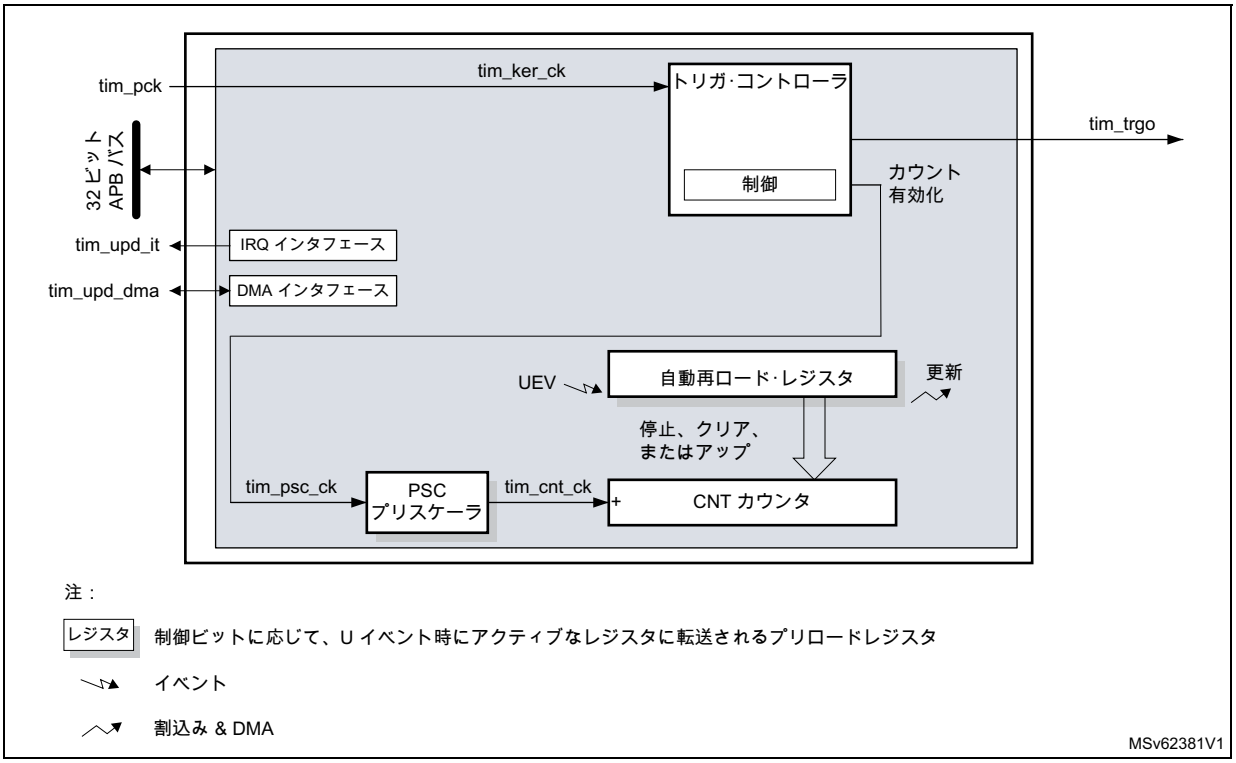
基本タイマ（TIM6/TIM7）の機能は、次のとおりです。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケーラ（動作中も変更可能）で、カウンタクロック周波数を 1 から 65535 の間の値で分周可能
- DAC をトリガする同期回路
- 更新イベント時の割込み／DMA 生成：カウンタオーバーフロー

31.3 TIM6/TIM7 機能詳細

31.3.1 TIM6/TIM7 ブロック図

図 480. 基本タイマブロック図



31.3.2 TIM6/TIM7 内部信号

このセクションの以下の表は、TIM の入力および出力の一覧です。

表 304. TIM 内部入力／出力信号

内部信号名	信号タイプ	説明
tim_pclk	入力	タイマ APB クロック
tim_ker_ck	入力	タイマカーネルクロック。このクロックは、tim_pclk（同じソースから導出）と同期する必要があります。クロック比 tim_ker_ck/tim_pclk は整数 1、2、3、・・・16（最大値）でなければなりません。
tim_trgo	出力	内部トリガ出力。このトリガは他のオンチップ·ペリフェラル（DAC）をトリガできます。
tim_upd_it	出力	タイマ更新イベント割込み
tim_upd_dma	出力	タイマ更新 DMA リクエスト

31.3.3 TIM6/TIM7 クロック

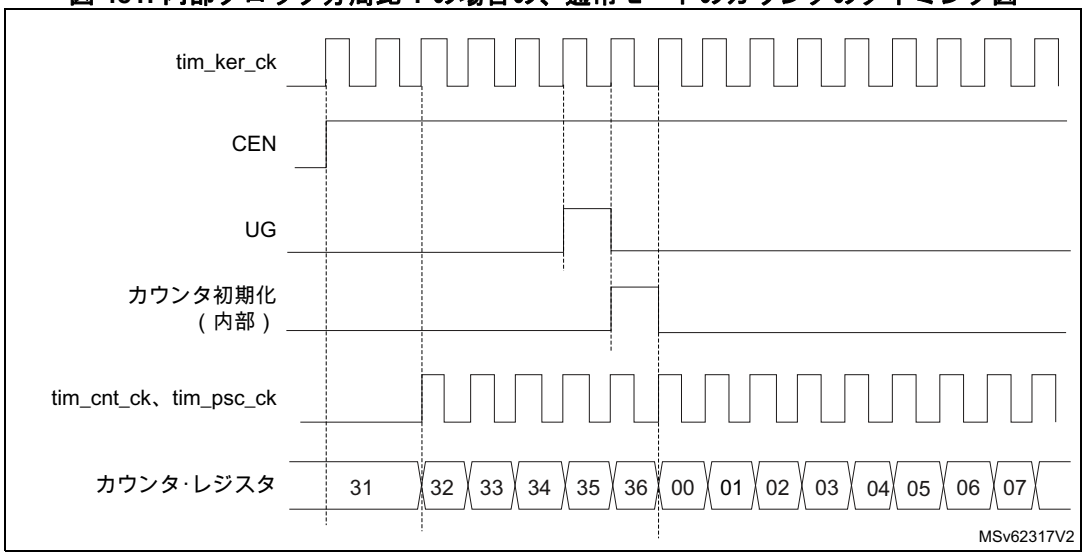
タイマバスインタフェースは tim_pclk APB クロックによってクロック供給されます。

カウンタクロック tim_ker_ck は tim_pclk 入力に接続されます。

TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます（ただし、自動的にクリア状態が保持される UG ビットを除く）。CEN ビットに 1 が書き込まれるとすぐに、プリスケーラには内部クロック tim_ker_ck が供給されます。

図 481 に、プリスケーラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

図 481. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



31.3.4 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケーラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケーラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ（TIMx_CNT）
- プリスケーラレジスタ（TIMx_PSC）
- 自動再ロードレジスタ（TIMx_ARR）

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット（ARPE）に応じて、常時または更新イベント UEV ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット（CEN）がセットされているときのみ、プリスケーラ出力 tim_cnt_ck から供給されます。

実際のカウンタイネーブル信号 tim_cnt_en は、CEN ビットセットの 1 クロックサイクル後にセットされることにご注意ください。

プリスケーラの説明

プリスケーラは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビットレジスタ（TIMx_PSC レジスタ）を使って制御される 16 ビットカウンタをベースとしています。TIMx_PSC 制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケーラ比は、次の更新イベントで有効になります。

図 482 と 図 483 に、プリスケーラ比を動作中に変更したときのカウンタの動作の例を示します。

図 482. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

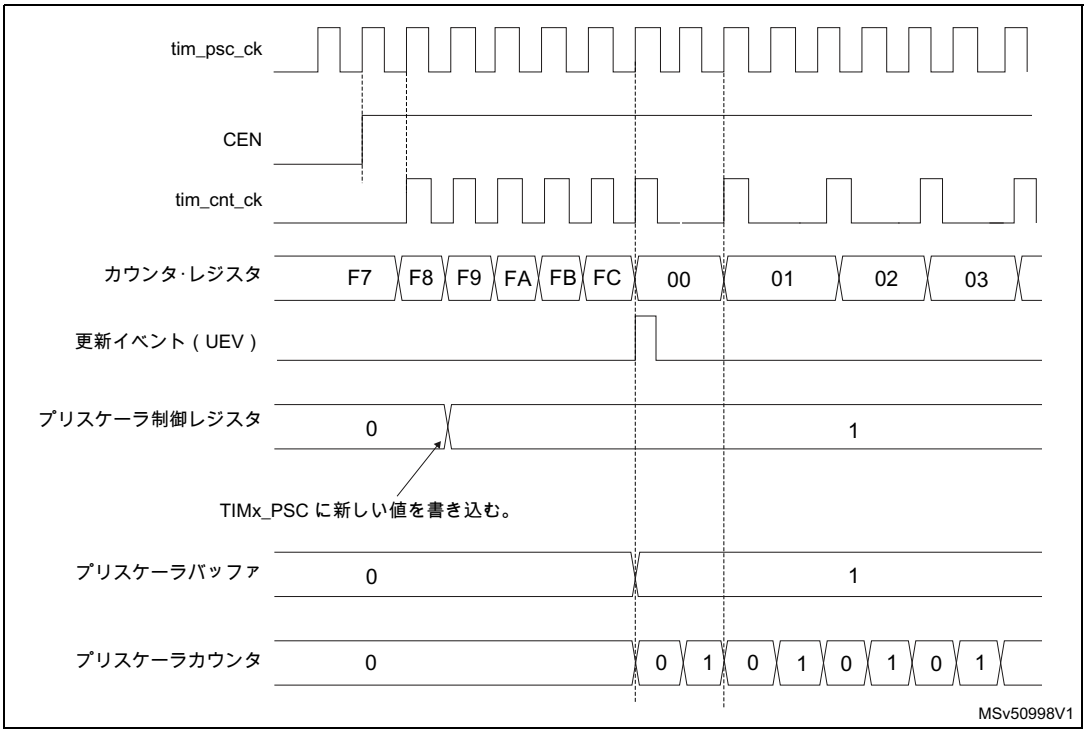
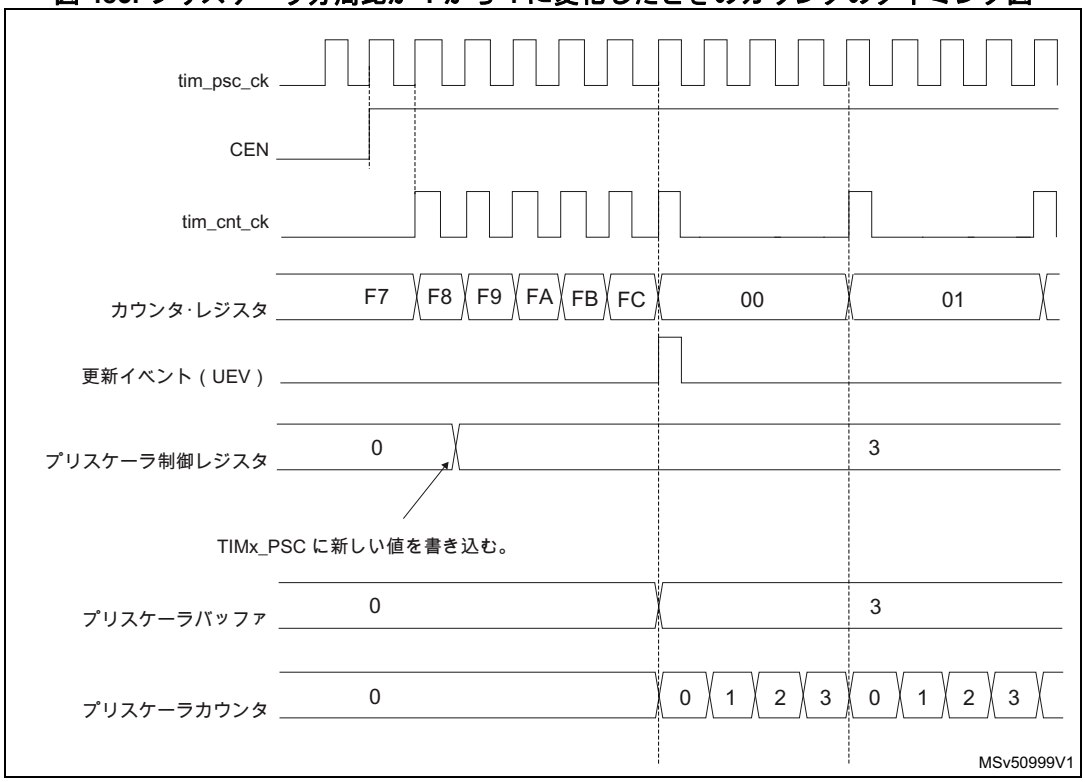


図 483. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



31.3.5 カウントモード

カウンタは、0 から自動再ロード値（TIMx_ARR レジスタの内容）までカウントした後、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGR レジスタの UG ビットをセットすることによって（ソフトウェアで、または、スレーブモードコントローラを使用して）生成できます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDIS ビットに 0 が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケーラカウンタは両方とも 0 からリスタートします（ただし、プリスケーラ比は変化しません）。さらに、TIMx_CR1 レジスタの URS（更新リクエスト選択）ビットがセットされている場合、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません（割込みや DMA リクエストは送信されません）。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットの設定に応じて、更新フラグ（TIMx_SR レジスタの UIF ビット）がセットされます。

- プリスケーラのバッファにはプリロード値（TIMx_PSC レジスタの内容）が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値（TIMx_ARR）で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 484. 内部クロック分周比が 1 の場合のカウンタのタイミング図

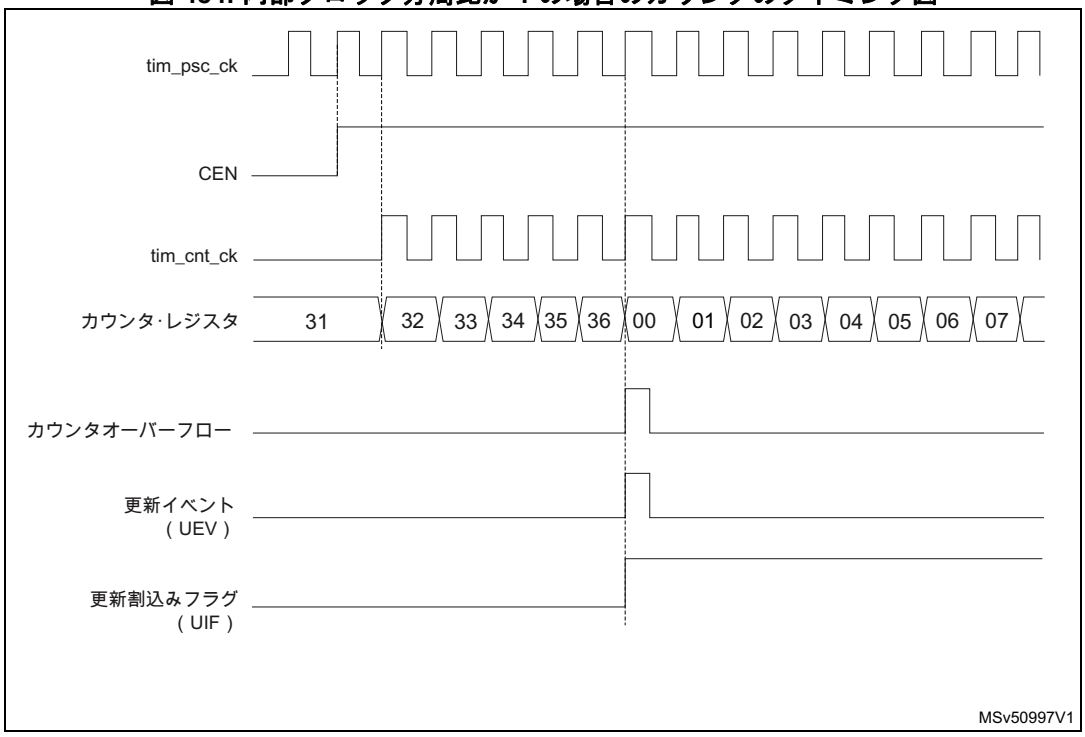


図 485. 内部クロック分周比が 2 の場合のカウンタのタイミング図

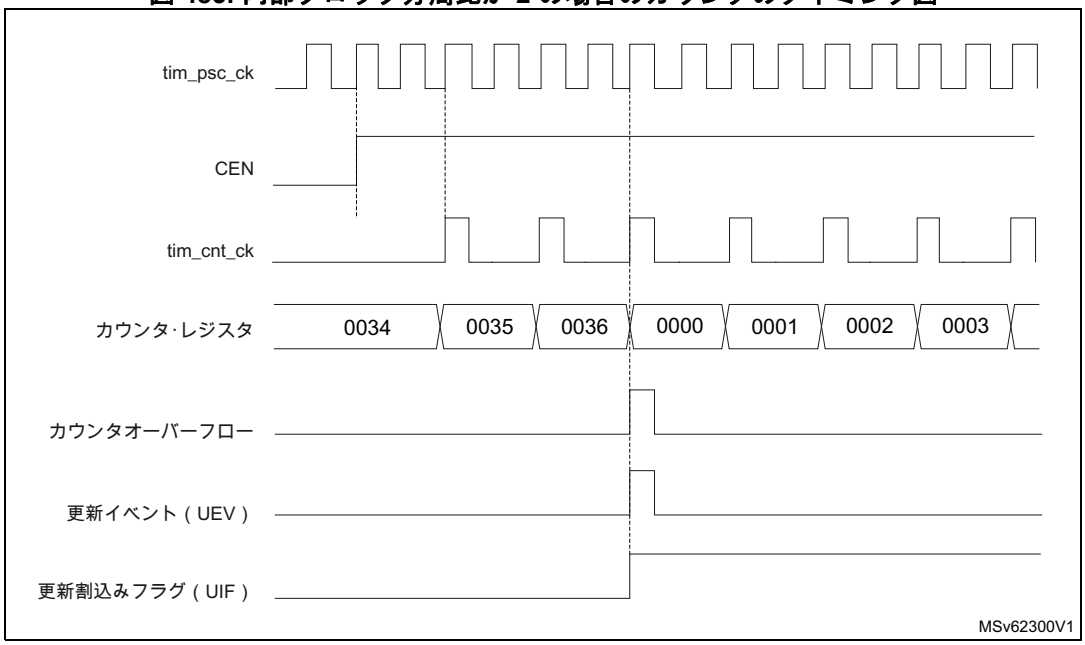


図 486. 内部クロック分周比が 4 の場合のカウンタのタイミング図

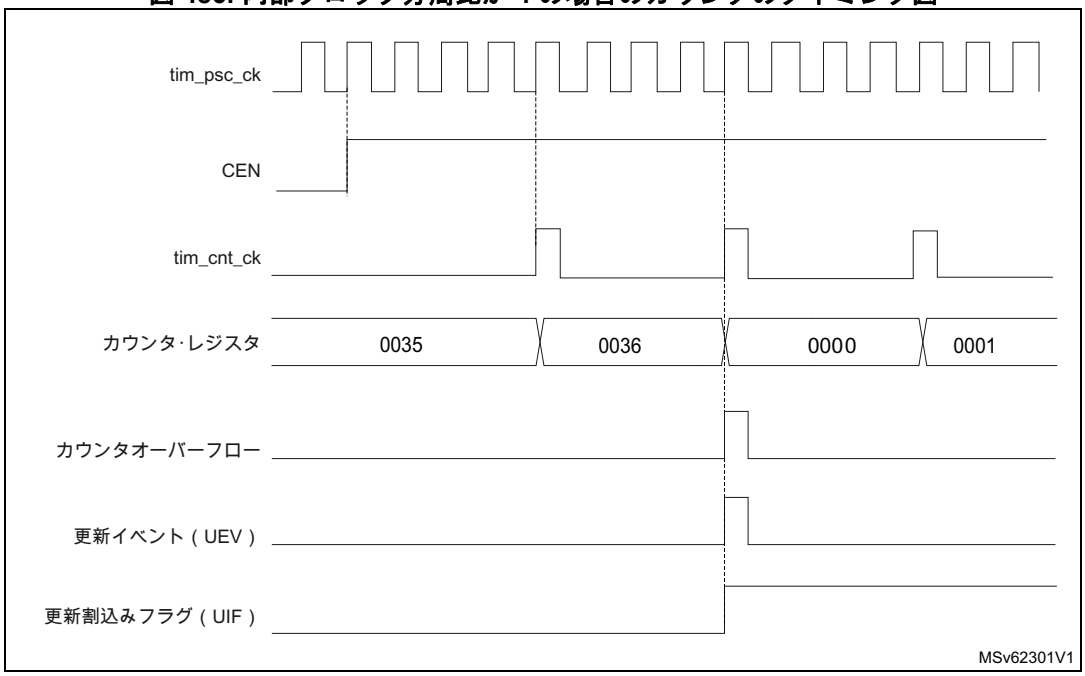


図 487. 内部クロック分周比が N の場合のカウンタのタイミング図

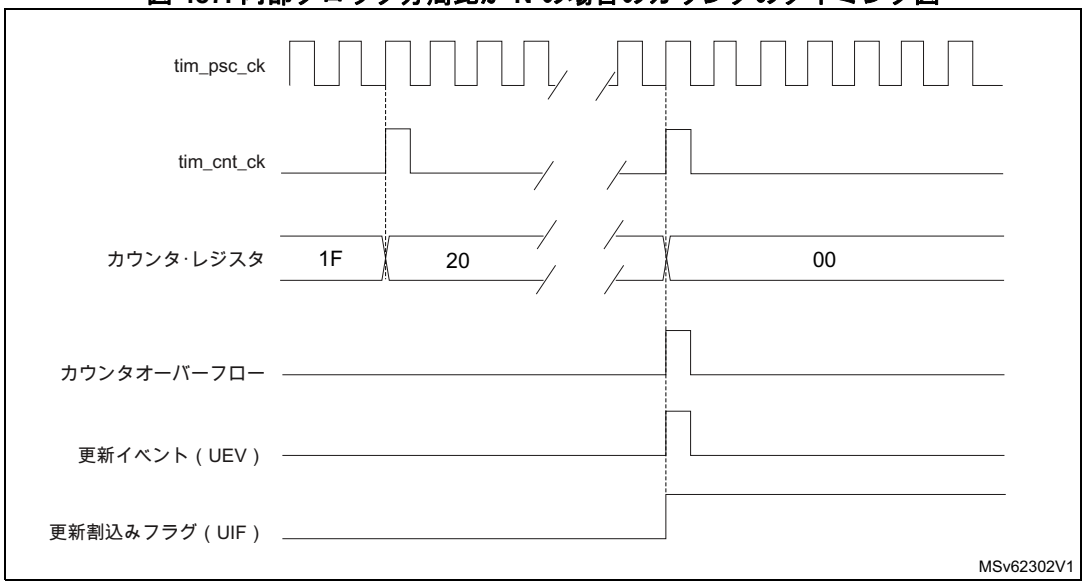


図 488. ARPE=0（TIMx_ARR がプリロードされない）の場合の更新イベント時のカウンタのタイミング図

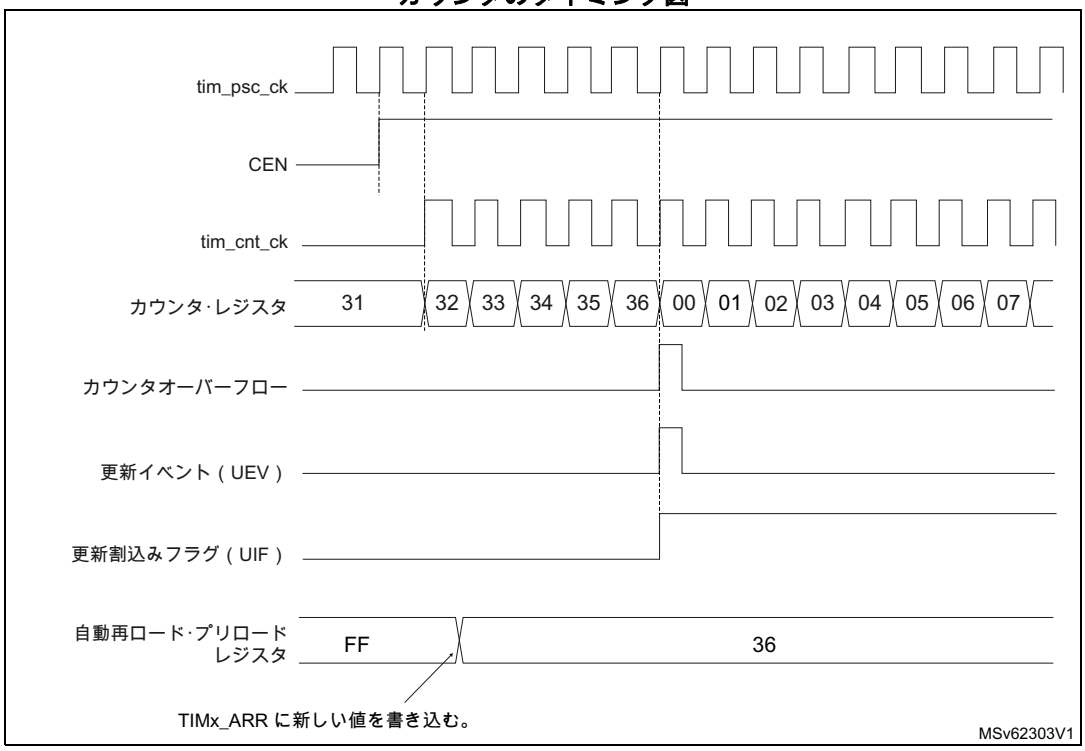
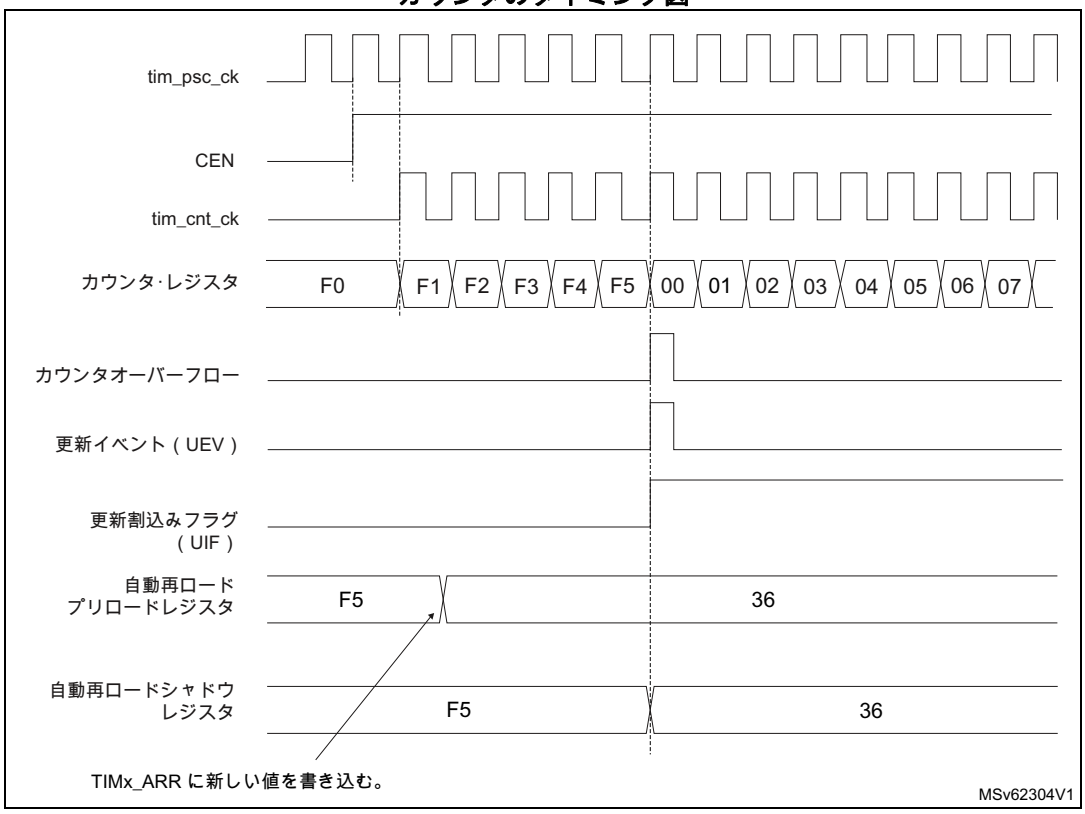


図 489. ARPE=1（TIMx_ARR がプリロードされる）の場合の更新イベント時のカウンタのタイミング図



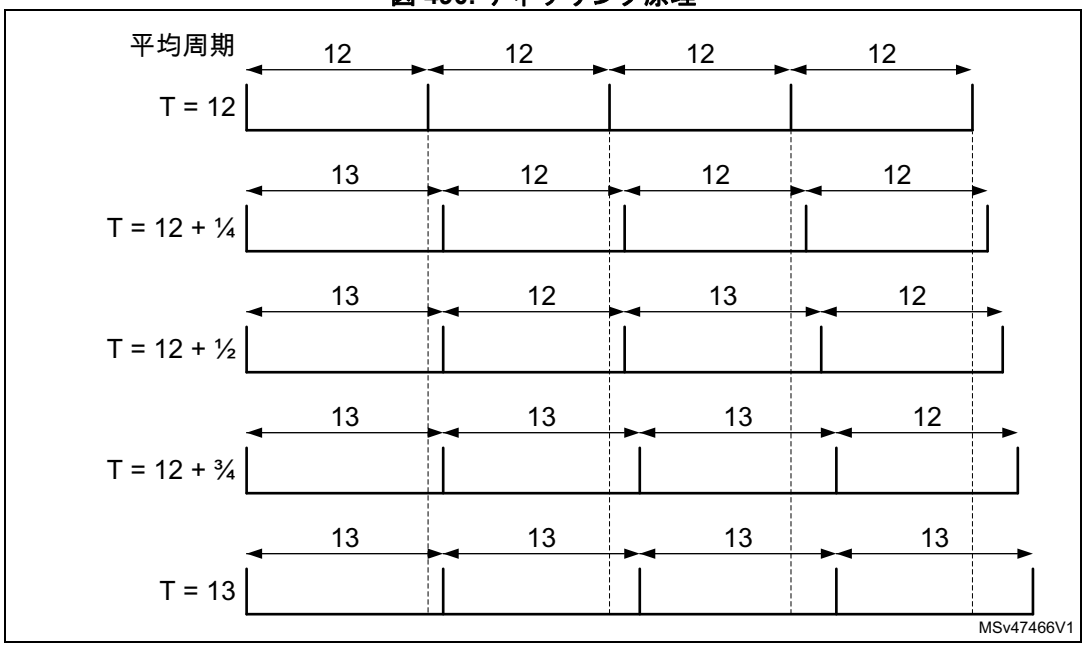
ディザリング·モード

タイムベースの有効分解能は、TIMx_CR1 レジスタの DITHEN ビットを使用して、ディザリングモードを有効にすることで増加できます。これは、TIMx_ARR の動作に影響し、タイマがトリガとして使用される場合（通常は DAC の場合）に平均カウンタ周期を調整するのに役立ちます。

動作原理は、事前に定義されたパターンで、16 個の連続した カウント 周期にわたって実際の ARR 値をわずかに変更する（1 タイマクロック周期を追加する、またはしない）ことです。これにより、平均のカウント周期を考えると、16 倍の分解能の向上が可能になります。

下記の図 490 は、4 つの連続するカウント周期に適用されたディザリング原理を示します。

図 490. ディザリング原理

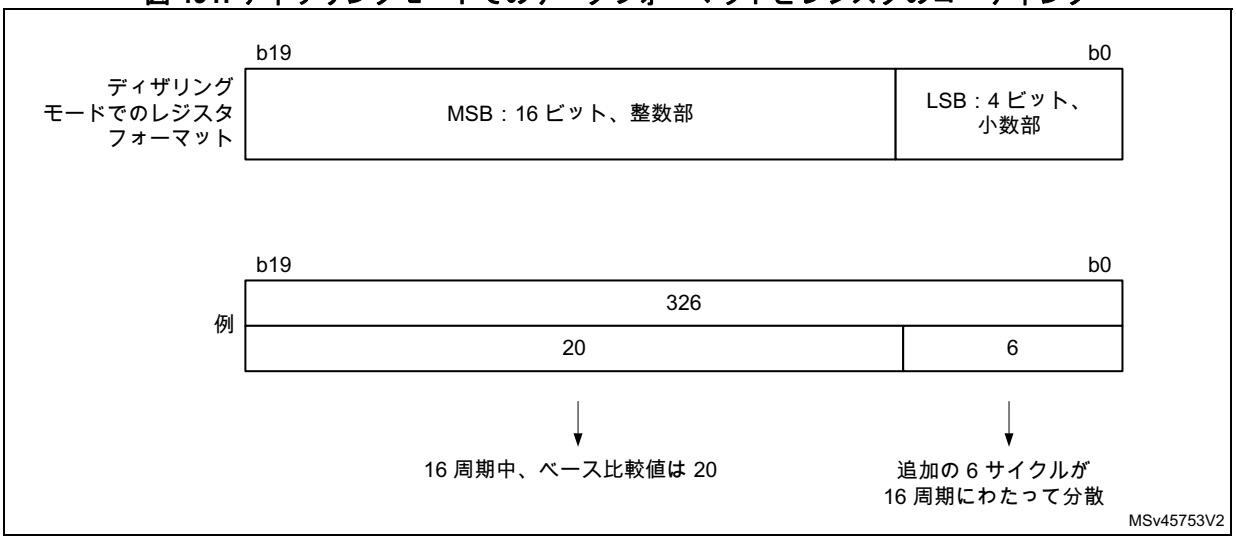


ディザリングモードが有効になっている場合、レジスタのコーディングは以下のように変更されます（たとえば、図 491 を参照）。

- 下位 4 ビットは、拡張分解能部分（分数部分）用です。
- 上位ビットは、ビット 19:4 に左シフトされ、ベース値用のコーディングになります。

注： **DITHEN ビットがセットまたはリセットされると、ARR の値が自動的に更新されます（たとえば、DITHEN = 0 で ARR = 0x05 の場合、DITHEN = 1 で ARR = 0x50 に更新されます）。DITHEN ビットをリセットするときは、次の手順に従う必要があります。**
1. CEN および ARPE ビットをリセットする必要があります。
2. ARR[3:0] ビットをリセットする必要があります。
3. DITHEN ビットをリセットする必要があります。
4. CEN ビットをセットすることができます（最終的に ARPE = 1 とともに）。

図 491. ディザリングモードでのデータフォーマットとレジスタのコーディング



最小周波数は次の計算式で与えられます。

$$\text{分解能} = \frac{F_{Tim}}{F_{pwm}} \Rightarrow F_{pwmMin} = \frac{F_{Tim}}{Max_{Resolution}}$$

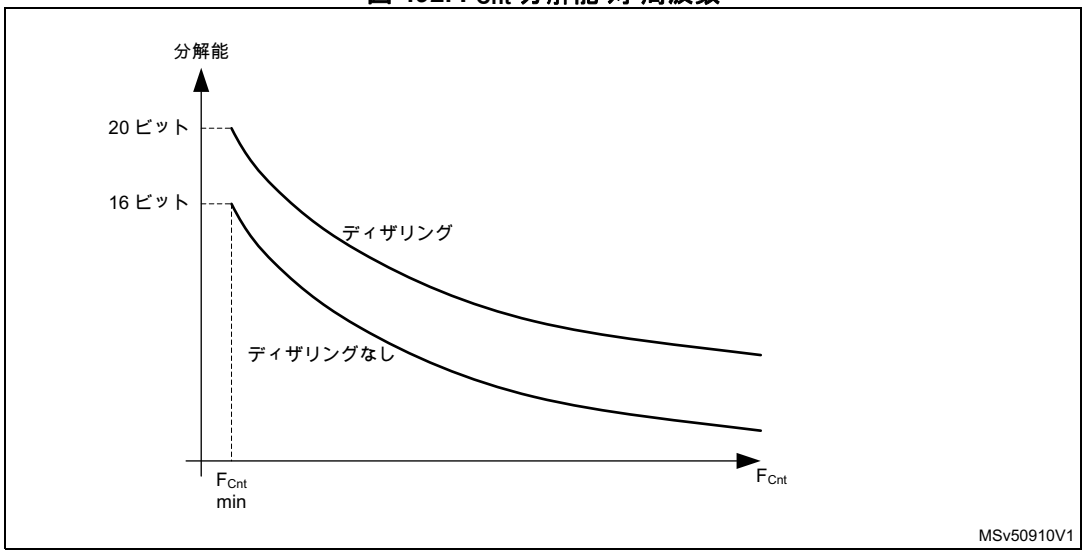
$$\text{ディザリングモード無効時：} F_{pwmMin} = \frac{F_{Tim}}{65536}$$

$$\text{ディザリングモード有効時：} F_{pwmMin} = \frac{F_{Tim}}{65535 + \frac{15}{16}}$$

注： **TIMx_ARR の最大値は、ディザリングモードでは 0xFFFEF に制限されています（整数部では 65534、ディザリング部分では 15 に対応します）。**

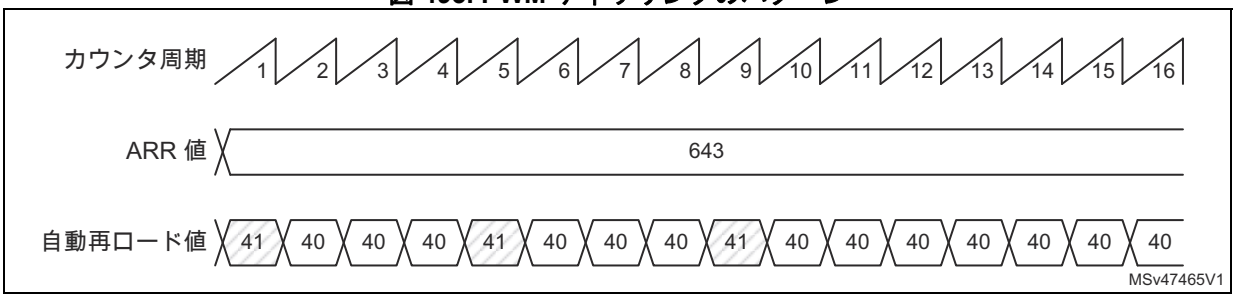
下記の図 492 に示すように、PWM 周波数に関係なく、ディザリングモードによって PWM 分解能を増加することができます。

図 492. F_{Cnt} 分解能 対 周波数



下記の図 493 で説明するように、周期の変更は、連続した 16 周期にわたって広がります。

図 493. PWM ディザリングのパターン



自動再ロード値と比較値の増分は、下記の表 305 に記載された特定のパターンに従って分布します。ディザリングシーケンスは、増分が可能な限り均等に分散され、全体のリップルが最小になるように行われます。

表 305. TIMx_ARR レジスタの変更のディザリングパターン

-	PWM 周期															
LSB 値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0000	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0001	+1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0010	+1	-	-	-	-	-	-	-	+1	-	-	-	-	-	-	-
0011	+1	-	-	-	+1	-	-	-	+1	-	-	-	-	-	-	-
0100	+1	-	-	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0101	+1	-	+1	-	+1	-	-	-	+1	-	-	-	+1	-	-	-
0110	+1	-	+1	-	+1	-	-	-	+1	-	+1	-	+1	-	-	-

表 305. TIMx_ARR レジスタの変更のディザリングパターン （続き）

-	PWM 周期															
LSB 値	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0111	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	-	-
1000	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1001	+1	+1	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-	+1	-
1010	+1	+1	+1	-	+1	-	+1	-	+1	+1	+1	-	+1	-	+1	-
1011	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	-	+1	-
1100	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1101	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	-	+1	+1	+1	-
1110	+1	+1	+1	+1	+1	+1	+1	-	+1	+1	+1	+1	+1	+1	+1	-
1111	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	+1	-

31.3.6 UIF ビットの再配置

TIMx_CR1 レジスタの IUFREMAP ビットでは、タイマカウンタレジスタのビット 31（TIMxCNT[31]）に更新割込みフラグ（UIF）の連続コピーを強制します。これにより、UIFCPY フラグによって示されたカウンタ値と潜在的なロールオーバー条件を分割できないものとして読み取ることができます。特定のケースでは、バックグラウンドタスク（カウンタの読出し）と割込み（更新の割込み）との間で共有されている処理などによって生じる競合状態を避けることで、計算が容易になります。

UIF と UIFCPY フラグのアサートの間には、遅延はありません。

31.3.7 ADC トリガ

タイマでは、リセット･イベント、イネーブル･イベント、比較イベントなどのさまざまな内部信号でADC トリガ･イベントを生成できます。

注： **tim_trgo 信号を受信するスレーブ･ペリフェラル（タイマ、ADC など）のクロックは、マスタ･タイマからイベントを受信する前に有効化する必要があり、マスタ･タイマからトリガを受信している間はクロック周波数（プリスケーラ）を動作中に変更しないでください。**

31.3.8 TIM6/TIM7DMA リクエスト

表 306 に示すように、TIM6/TIM7 は単一の DMA リクエストを生成できます。

表 306. DMA リクエスト

DMA リクエスト信号	DMA 略称	DMA リクエスト	有効化制御ビット
tim_upd_dma	TIM_UP	更新	UDE

31.3.9 デバッグ･モード

マイクロコントローラがデバッグ･モードになると（FPU 搭載 Cortex®-M4 コアは停止状態）、TIMx カウンタは、通常どおりに動作を続けるか、または停止できます。

デバッグモードでの動作は、デバッグサポート（DBG）モジュールでタイマごとに専用の設定ビットを使用してプログラムできます。

詳細については、デバッグサポート（DBG）のセクションを参照してください。

31.3.10 TIM6/TIM7 低消費電力モード

表 307. 低消費電力モードが TIM6/TIM7 に与える影響

モード	説明
SLEEP	影響はなく、ペリフェラルはアクティブです。割込みによって、デバイスは SLEEP モードから復帰できます。
STOP	タイマの動作は停止され、レジスタの内容は保持されます。割込みは生成できません。
STANDBY	タイマはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

31.3.11 TIM6/TIM7 割込み

表 308 に示すように、TIM6/TIM7 は単一の割込みを生成できます。

表 308. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了
TIM6 TIM7	更新	UIF	UIE	UIF に 0 を書き込む	可能	不可

31.4 TIM6/TIM7 レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル・レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

31.4.1 TIMx 制御レジスタ 1（TIMx_CR1）（x = 6 ~ 7）

アドレス・オフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
			rw	rw				rw				rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DITHEN**：ディザリング・イネーブル

0：ディザリングは無効です。

1：ディザリングは有効です。

注： DITHEN ビットは、CEN ビットがリセットされているときのみ変更できます。

ビット 11 **UIFREMAP**：UIF ステータスビットの再配置

0：再配置なし。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされません。

1：再配置は有効です。UIF ステータスビットは TIMx_CNT レジスタのビット 31 にコピーされます。

ビット 10:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **ARPE**：自動再ロード・プリロード・イネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM**：ワンパルス・モード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 **URS**：更新リクエスト・ソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0：次のイベントのいずれかが更新割込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1：カウンタオーバーフロー／アンダーフローのみが更新割込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 **UDIS**：更新ディスエーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0：UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

– カウンタオーバーフロー／アンダーフロー

– UG ビットのセット

– スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1：UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC）はそれぞれの値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケーラは再初期化されます。

ビット 0 **CEN**：カウンタ･イネーブル

0：カウンタは無効です。

1：カウンタは有効です。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

31.4.2 TIMx 制御レジスタ 2（TIMx_CR2）（x = 6 ~ 7）

アドレス･オフセット：0x04

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS[2:0]			Res.	Res.	Res.	Res.
									rw	rw	rw				

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS[2:0]**：マスタ･モード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報（TRGO）を選択するために使用します。組み合わせは、次のとおりです。

000：**リセット**- TIMx_EGR レジスタの UG ビットがトリガ出力（tim_trgo）として使用されます。

001：**イネーブル** - カウンタ･イネーブル信号 tim_cnt_en がトリガ出力（tim_trgo）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、CEN 制御ビットが書き込まれると生成されます。

010：**更新** - 更新イベントがトリガ出力（tim_trgo）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケーラとして使用できます。

注： スレーブタイマまたは tim_trgo を受信するペリフェラルのクロックは、マスタタイマからイベントを受信する前に有効化する必要があり、マスタタイマからトリガを受信している間は動作中に変更しないでください。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

31.4.3 TIMx DMA／割込み有効レジスタ（TIMx_DIER）（x = 6 ~ 7）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
							rw								rw

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UDE**：更新 DMA リクエスト･イネーブル

0：更新 DMA リクエストは無効です。

1：更新 DMA リクエストは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIE**：更新割込みイネーブル

0：更新割込みは無効です。

1：更新割込みは有効です。

31.4.4 TIMx ステータスレジスタ（TIMx_SR）（x = 6 ~ 7）

アドレスオフセット：0x10

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
															rc_w0

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIF**：更新割込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0：更新は発生していません。

1：更新割込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

–TIMx_CR1 レジスタの UDIS = 0 の場合、カウンタのオーバーフローが発生したとき。

–TIMx_CR1 レジスタの URS = 0 かつ UDIS = 0 の場合に、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで CNT が再初期化されたとき。

31.4.5 TIMx イベント生成レジスタ（TIMx_EGR）（x = 6 ~ 7）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
															w

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
0：影響はありません。
1：タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケーラカウンタもクリアされます（プリスケーラ比は変化しません）。

31.4.6 TIMx カウンタ （TIMx_CNT）（x = 6 ~ 7）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIF CPY	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UIFCPY**：UIF コピー

このビットは TIMx_ISR レジスタの UIF ビットの読出し専用コピー。TIMx_CR1 の UIFREMAP ビットがリセットされると、ビット 31 は予約済みで、0 で読み出されます。

ビット 30:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

非ディザリングモード（DITHEN = 0）
このレジスタは、カウンタ値を保持します。
ディザリングモード（DITHEN = 1）
このレジスタは、CNT[15:0] に非ディザリング部分のみを保持します。分数部分は得られません。

31.4.7 TIMx プリスケーラ（TIMx_PSC）（x = 6 ~ 7）

アドレスオフセット：0x28

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **PSC[15:0]**：プリスケーラ値

カウンタクロック周波数 $f_{tim_cnt_ck}$ は $f_{tim_psc_ck}$ /（PSC[15:0] + 1）に等しいです。
PSC には、更新イベントのたびにアクティブプリスケーラレジスタにロードされる値が含まれます。
（更新イベントには、TIMx_EGR レジスタの UG ビットを通じて、カウンタがクリアされる場合も含まれます）。

31.4.8 TIMx 自動再ロードレジスタ （TIMx_ARR）（x = 6 ~ 7）

アドレスオフセット：0x2C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **ARR[19:0]**：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。
ARR の更新と動作の詳細については、セクション 31.3.4 : 1416 ページのタイムベースユニットを参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。
非ディザリングモード（DITHEN = 0）
このレジスタは、ARR[15:0] に自動再ロード値を保持します。ARR[19:16] ビットは予約されています。
ディザリングモード（DITHEN = 1）
このレジスタは、ARR[19:4] に整数部を保持します。ARR[3:0] ビットフィールドにはディザリング部分が含まれます。

31.4.9 TIMx レジスタマップ

TIMx レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 309. TIMx レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**TIMx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DITHEN	UIFREMAP	Res.	Res.	Res.	ARPE	Res.	Res.	Res.	OPM	URS	UDIS	CEN
	リセット値																				0	0				0				0	0	0	0
0x04	**TIMx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MMS [2:0]			Res.	Res.	Res.	Res.
	リセット値																										0	0	0				
0x08	予約済みです。																																
0x0C	**TIMx_DIER**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UDE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIE
	リセット値																								0								0
0x10	**TIMx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UIF
	リセット値																																0
0x14	**TIMx_EGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	UG
	リセット値																																0
0x18-0x20	予約済みです。																																
0x24	**TIMx_CNT**	UIFCPY または Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値	0																0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x28	**TIMx_PSC**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	**TIMx_ARR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[19:0]																			
	リセット値													0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

32 低電力タイマ（LPTIM）

32.1 概要

LPTIM は、消費電力削減の究極的な進展を利用した 16 ビットタイマです。クロックソースの多様性により、LPTIM は STANDBY モードを除くすべての電力モードで実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIM は、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIM はシステムを低電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIM の柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

32.2 LPTIM の主な機能

- 16 ビットアップカウンタ
- 8 つの分周比（1、2、4、8、16、32、64、128）を持つ 3 ビットプリスケーラ
- 選択可能なクロック
  - 内部クロック･ソース：設定可能な内部クロック･ソース（RCC のセクションを参照してください）
  - LPTIM 入力経由の外部クロック･ソース（内蔵オシレータが動作していないときに動作、パルス･カウンタ･アプリケーションで使用）
- 16 ビット ARR 自動再ロード･レジスタ
- 16 ビット比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能な I/O 極性
- エンコーダモード

32.3 LPTIM の実装

表 310 に、STM32G4 シリーズ デバイスでの LPTIM の実装を示します。

表 310. STM32G4 シリーズ LPTIM 機能

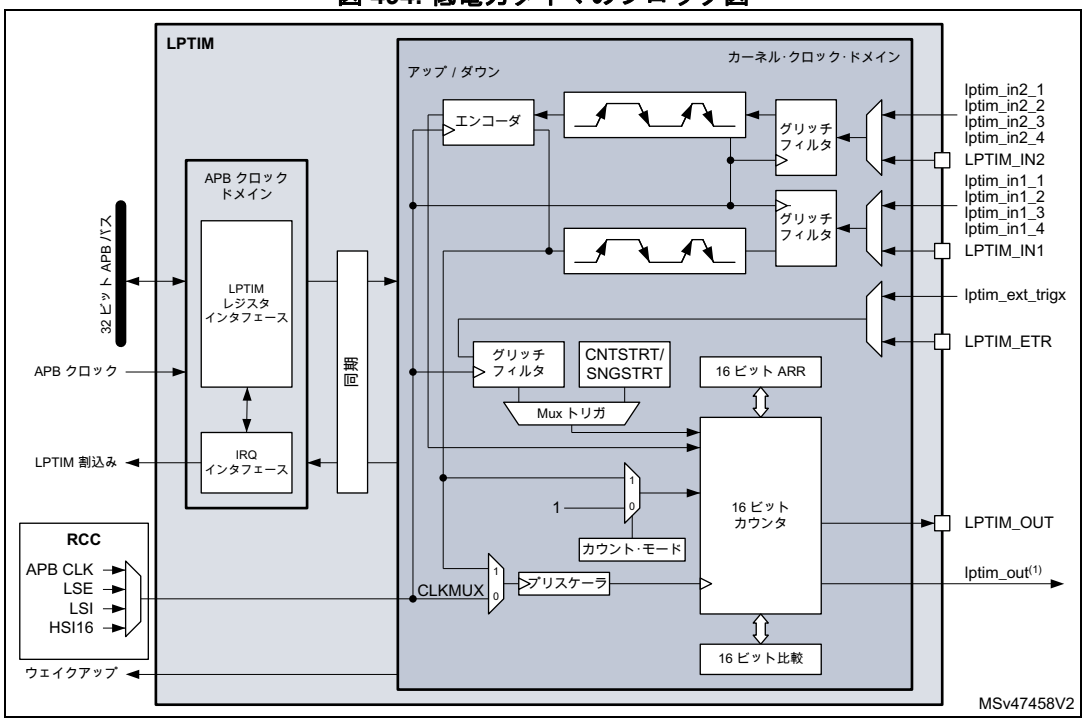
LPTIM のモード／機能(1)	LPTIM1
エンコーダ·モード	X

1. X：サポートされています。

32.4 LPTIM の機能詳細

32.4.1 LPTIM ブロック図

図 494. 低電力タイマのブロック図



1. lptim_out は、内部ペリフェラルに接続できる内部 LPTIM 出力信号です。

32.4.2 LPTIM 入力およびトリガマッピング

LPTIM 外部トリガおよび入力接続について、以下に詳しく説明します。

表 311. LPTIM1 外部トリガ接続

TRIGSEL	外部トリガ
lptim_ext_trig0	GPIO
lptim_ext_trig1	RTC_ALARMA
lptim_ext_trig2	RTC_ALARMB

表 311. LPTIM1 外部トリガ接続 （続き）

TRIGSEL	外部トリガ
lptim_ext_trig3	RTC_TAMP1_OUT
lptim_ext_trig4	RTC_TAMP2_OUT
lptim_ext_trig5	RTC_TAMP3_OUT
lptim_ext_trig6	COMP1_OUT
lptim_ext_trig7	COMP2_OUT
lptim_ext_trig8	COMP3_OUT
lptim_ext_trig9	COMP4_OUT
lptim_ext_trig10	COMP5_OUT
lptim_ext_trig11	COMP6_OUT
lptim_ext_trig12	COMP7_OUT

表 312. LPTIM1 入力 1 接続

lptim_in1_mux	LPTIM1 入力 1 接続先
lptim_in1_0	LPTIM1_IN1 のオルタネート機能としての GPIO ピン
lptim_in1_1	COMP1
lptim_in1_2	COMP3
lptim_in1_3	COMP5
lptim_in1_4	COMP7

表 313. LPTIM1 入力 2 接続

lptim_in2_mux	LPTIM1 入力 2 接続先
lptim_in2_0	LPTIM1_IN2 のオルタネート機能としての GPIO ピン
lptim_in2_1	COMP2
lptim_in2_2	COMP4
lptim_in2_3	COMP6
lptim_in2_4	COMP6

32.4.3 LPTIM のリセットとクロック

LPTIM のクロックには、いくつかのクロックソースを使用できます。設定可能な任意の内部クロック･ソースを RCC で選択し、LPTIM への内部クロック信号とすることができます（詳細は RCC のセクションを参照してください）。また、LPTIM のクロックとして、外部 Input1 に入力した外部クロック信号を使用することもできます。外部クロックソースでクロック供給されるときには、LPTIM は、次の 2 つの構成のいずれかで動作します。

- 第 1 の構成では、LPTIM のクロックを外部信号によって供給しますが、設定可能な内部クロック･ソースによる内部クロック信号も同時に LPTIM に供給します（RCC のセクションを参照してください）。

- 第 2 の構成では、外部 Input1 からの外部クロック･ソースのみで LPTIM を駆動します。この構成は、低電力モードになった後、すべての埋め込みオシレータがオフになるときに、タイムアウト機能またはパルスカウンタ機能を実現するために使用されます。

CKSEL ビットおよび COUNTMODE ビットに書き込むことによって、LPTIM が外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOL ビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給されます（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の 4 倍以上である必要があります。

32.4.4 グリッチフィルタ

LPTIM 入力の外部入力（GPIO にマップされたもの）または内部入力（チップレベルで他の内蔵ペリフェラルにマップされたもの）はデジタルフィルタによって保護され、グリッチとノイズの影響が LPTIM 内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースを LPTIM に供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

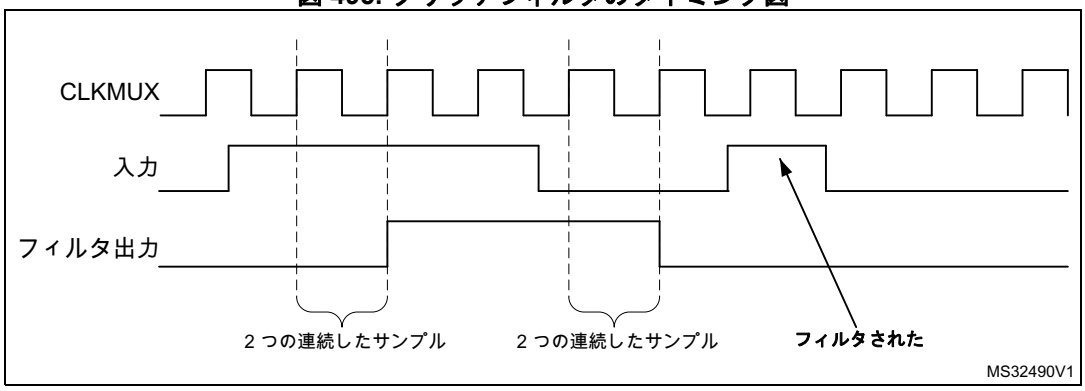
デジタルフィルタは、2 つのグループに分けられます。

- 第 1 のグループのデジタル･フィルタは、LPTIM の外部入力を保護します。デジタルフィルタの感度は CKFLT ビットによって制御されます。
- 2 番目のグループのデジタルフィルタは、LPTIM の内部トリガ入力を保護します。デジタルフィルタの感度は TRGFLT ビットによって制御されます。

注： **デジタルフィルタの感度は 、グループごとに制御されます。同じグループ内の各デジタルフィルタの感度を個別に設定することはできません。**

フィルタの感度は、信号のレベル変化を有効な遷移とみなすために、LPTIM 入力の1 つで検出される連続した等しいサンプルの数に基づいて作用します。図 495 に、2 つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図 495. グリッチフィルタのタイミング図



注： **内部クロック信号が供給されない場合は、CKFLT および TRGFLT ビットを 0 にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIM の外部入力をグリッチから保護できます。**

32.4.5 プリスケーラ

LPTIM 16 ビットカウンタの前には、設定可能な 2 のべき乗プリスケーラがあります。プリスケーラの分周比は PRESC[2:0] 3 ビットフィールドによって制御されます。下の表に、可能な分周比を示します。

表 314. プリスケーラ分周比

プログラミング	分周比
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

32.4.6 トリガマルチプレクサ

LPTIM カウンタは、ソフトウェアによって、または 13 個のトリガ入力の 1 つのアクティブエッジの検出後に開始できます。

LPTIM のトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が 00 の場合、LPTIM カウンタは、CNTSTRT または SNGSTRT ビットがソフトウェアによってセットされるとすぐに開始します。TRIGEN[1:0] の残りの 3 つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIM カウンタは、アクティブエッジが検出されるとすぐに開始します。
- TRIGEN[1:0] が 00 以外のときには、カウンタの開始に使用される 13 個のトリガ入力の 1 つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIM の場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに 2 カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： **SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。**

注： **ソフトウェアによってカウンタを起動する場合（TRIGEN[1:0] = 00）、LPTIM_CR レジスタの更新（SNGSTRT ビットまたは CNTSTRT ビットのセット）から、その更新を反映してカウンタが起動するまで、カーネル·クロック 3 サイクル分の遅延があります。**

32.4.7 動作モード

LPTIM には 2 つの動作モードがあります。

- 連続モード : タイマはフリーランニングし、トリガイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード : タイマはトリガイベントによって開始し、ARR 値に達すると停止します。

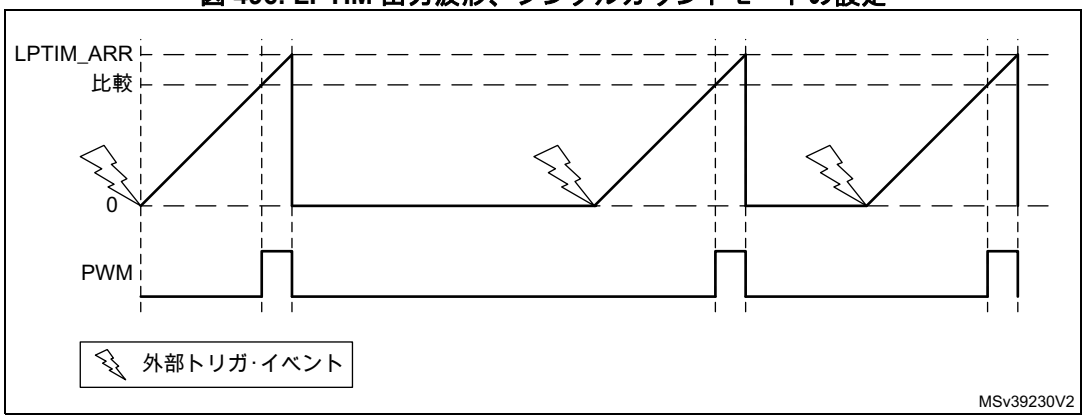
ワンショットモード

ワンショットカウントを有効にするには、SNGSTRT ビットをセットする必要があります。

新しいトリガイベントが発生すると、タイマは再開始します。カウンタの開始後、カウンタが ARR に達する前に発生したトリガイベントは破棄されます。

外部トリガが選択された場合、SNGSTRT ビットがセットされ、かつカウンタレジスタが停止された（ゼロ値を含む）後に各外部トリガイベントが着信すると、カウンタは新たなワンショットカウントサイクルを開始します（図 496 を参照）。

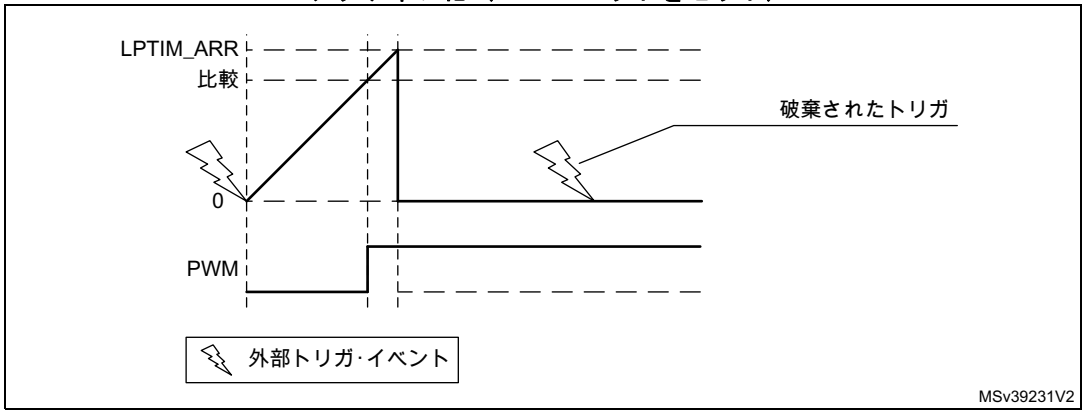
図 496. LPTIM 出力波形、シングルカウントモードの設定



- セットワンスモードのアクティブ化 :

LPTIM_CFGR レジスタの WAVE ビットフィールドがセットされると、セットワンスモードがアクティブになることに注意してください。この場合、最初のトリガに続いてカウンタが一度だけ開始され、その後発生したトリガイベントはすべて破棄されます（図 497 を参照）。

図 497. LPTIM 出力波形、シングルカウントモードの設定およびセットワンスモードのアクティブ化（WAVE ビットをセット）



ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、SNGSTRT をセットすると、カウンタはワンショットカウントを開始します。

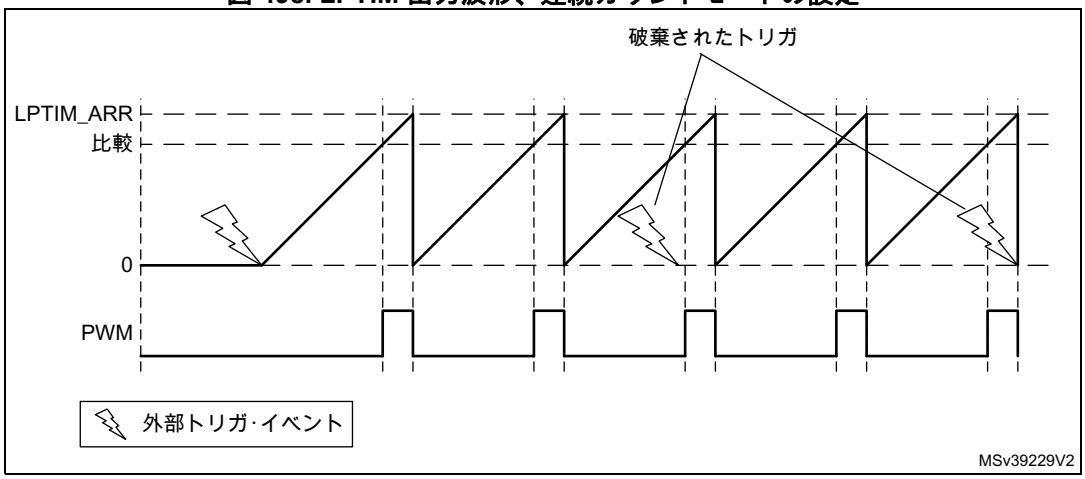
連続モード

連続カウントを有効にするには、CNTSTRT ビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRT がセットされた後に外部トリガイベントが着信すると、カウンタは連続カウントを開始します。その後発生した外部トリガイベントはすべて、図 498 に示すように破棄されます。

ソフトウェアによる開始の場合（TRIGEN[1:0] = 00）、CNTSTRT をセットすると、カウンタは連続カウントを開始します。

図 498. LPTIM 出力波形、連続カウントモードの設定



SNGSTRT および CNTSTRT ビットは、タイマが有効なときのみ（イネーブルビットが 1 にセットされている）、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRT をセットすると、LPTIM はワンショットモードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRT をセットすると、LPTIM は連続モードに切り替わります。カウンタ（アクティブな場合）は、ARR に達するとすぐに再開始します。

32.4.8 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIM カウンタをリセットできます。この機能は、TIMOUT ビットで制御されます。

最初のトリガイベントでタイマが開始し、その後のトリガイベントでカウントがリセットされ、タイマが再開始します。

低電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによって MCU がウェイクアップします。

32.4.9 波形生成

2 つの 16 ビット･レジスタ LPTIM_ARR（自動再ロード･レジスタ）と LPTIM_CMP（比較レジスタ）は、LPTIM 出力にさまざまな波形を生成するために使用します。

このタイマは次の波形を生成できます。

- PWM モード : LPTIM 出力は、LPTIM_CNT のカウンタ値が LPTIM_CMP の比較値を超えるとすぐにセットされます。LPTIM 出力は、LPTIM_ARR レジスタと LPTIM_CNT レジスタの間で一致が発生するとすぐにリセットされます。
- ワンパルスモード : 出力波形は、最初のパルスについては PWM モードの波形と同様であり、その後、出力は永続的にリセットされます。
- セットワンスモード : 出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます（設定された出力極性に応じて）。

上記のモードでは、LPTIM_ARR レジスタの値は LPTIM_CMP レジスタの値より必ず大きい必要があります。

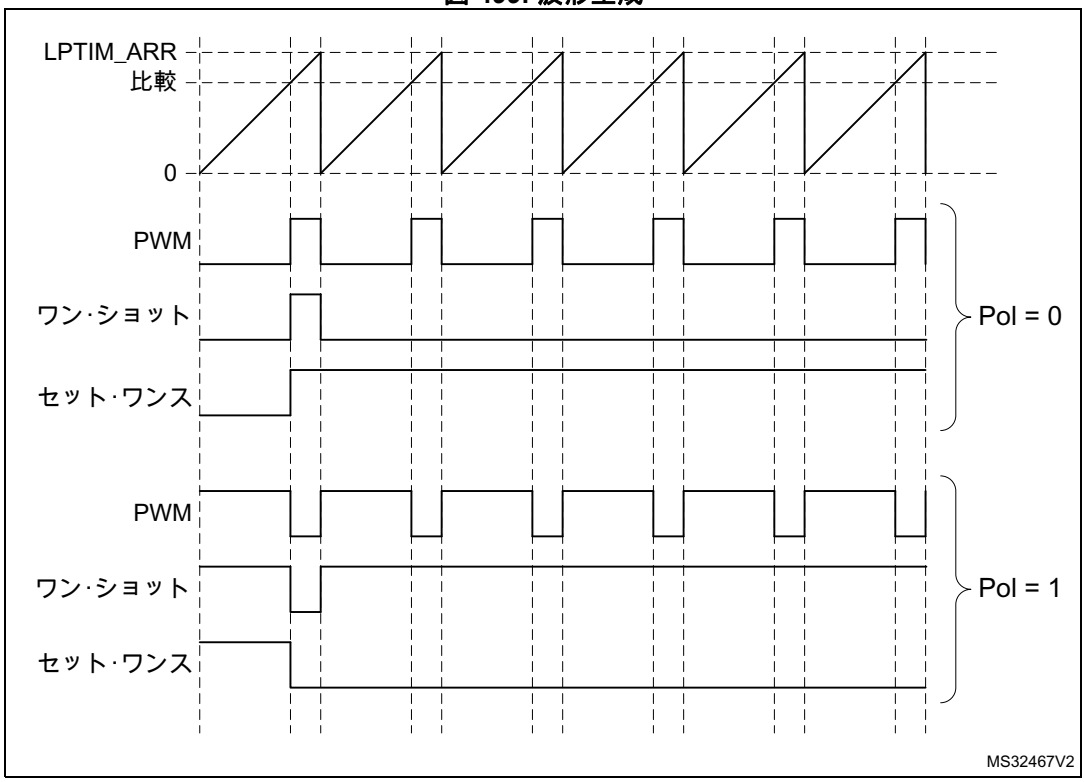
LPTIM 出力波形は、WAVE ビットによって次のように設定できます。

- WAVE ビットを 0 にリセットすると、LPTIM は CNTSTRT または SNGSTRT のセットされているビットに応じて、PWM 波形またはワンパルス波形のいずれかを生成します。
- WAVE ビットを 1 にセットすると、LPTIM はセットワンスモード波形を生成します。

WAVPOL ビットは、LPTIM 出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力はただちに変更されます。

LPTIM クロック周波数を 2 分周した周波数までの信号を生成できます。図 499 に、LPTIM 出力で生成できる 3 つの波形を示します。また、WAVPOL ビットを使用して極性を変更したときの効果も示します。

図 499. 波形生成



32.4.10 レジスタの更新

LPTIM_ARR レジスタと LPTIM_CMP レジスタは、APB バス書込み操作の直後に、またはタイマがすでに開始していた場合は現在の周期の終了時に更新されます。

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

- PRELOAD ビットが“0”にリセットされたときには、LPTIM_ARR レジスタおよび LPTIM_CMP レジスタは書込みアクセスの直後に更新されます。
- PRELOAD ビットが“1”にセットされたときには、LPTIM_ARR および LPTIM_CMP レジスタは、タイマがすでに開始していた場合、現在の周期の終了時に更新されます。

LPTIM APB インタフェースと LPTIM カーネルロジックは異なるクロックを使用するので、APB 書込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延時間内は、これらのレジスタに追加で書き込めません。

LPTIM_ISR レジスタの ARROK フラグと CMPOK フラグは、それぞれ、LPTIM_ARR レジスタと LPTIM_CMP レジスタへの書込み操作が完了したことを示します。

LPTIM_ARR レジスタまたは LPTIM_CMP レジスタへの書込みの後、同じレジスタへの新しい書込み操作は、前の書込み操作が完了してからでなければ実行できません。ARROK フラグまたは CMPOK フラグがセットされる前に連続した書込みが行われると、予測不能な結果になります。

32.4.11 カウンタモード

LPTIM カウンタを使用して、LPTIM Input1 の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。CKSEL ビットおよび COUNTMODE ビットは、カウンタの更新にどのソースを使用するかを決定します。

LPTIM が Input1 の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジで更新できます。

CKSEL および COUNTMODE の値に応じて、以下に示すカウントモードを選択できます。

- CKSEL = 0：LPTIM のクロックは、内部クロックソースによって供給されます。
  - COUNTMODE = 0

    LPTIM が内部クロックソースによってクロック供給されるように設定され、LPTIM カウンタが各内部クロックパルス後に更新されるように設定されます。
  - COUNTMODE = 1

    LPTIM 外部 Input1 は、LPTIM に供給される内部クロックでサンプリングされます。

    結果として、イベントをミスしないためには、外部 Input1 信号の変化の周波数が、LPTIM に供給される内部クロックの周波数を超えない必要があります。また、LPTIM に供給される内部クロックを分周してはなりません（PRESC[2:0] = 000）。
- CKSEL = 1：LPTIM のクロックは、外部クロックソースによって供給されます。

  COUNTMODE の 値は無視されます。

  この構成では、LPTIM は内部クロックソースを必要としません（グリッチフィルタ有効である場合を除く）。LPTIM 外部 Input1 に入力された信号が LPTIM のシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。

  この構成の場合、LPTIM カウンタは、Input1 クロック信号の立ち上がりエッジまたは立ち下がりエッジで更新できますが、立ち上がりと立ち下がりの両方のエッジで更新することはできません。

  LPTIM 外部 Input1 に入力された信号は LPTIM カーネルロジックのクロック動作にも使用されるので、カウンタがインクリメントされる前（LPTIM が有効にされた後）、初期遅延があります。より正確には、LPTIM 外部 Input1 の（LPTIM が有効になった後の）最初の 5 つのアクティブエッジは失われます。

32.4.12 タイマの有効化

LPTIM_CR レジスタのイネーブルビットは、LPTIM カーネルロジックを有効化／無効化するために使用されます。イネーブルビットをセットした後、LPTIM が実際に有効になるまで、2 カウンタクロックの遅延が必要です。

LPTIM_CFGR および LPTIM_IER レジスタの変更は、LPTIM が無効なときにのみ行う必要があります。

32.4.13 タイマカウンタのリセット

LPTIM_CNT レジスタの内容をゼロにリセットするために、以下の 2 つのリセットメカニズムが実装されています。

- 同期リセットメカニズム：同期リセットは、LPTIM_CR レジスタの COUNTRST ビットによって制御されます。COUNTRST ビットフィールドを“1”にセットした後、リセット信号が LPTIM カーネルクロックドメイン内に伝播されます。したがって、リセットが考慮される前に、LPTIM カーネルロジックの若干のクロックパルスが経過することに注意することが重要です。このため、リセットがトリガされてから有効になるまでの間に、LPTIM カウンタは若干の余分なパルスをカウントすることになります。COUNTRST ビットは APB クロックドメインに位置し、LPTIM カウンタは LPTIM カーネルクロックドメインに位置しているので、COUNTRST ビットに“1”を書き込んだ時に APB クロックによって発行されるリセット信号に同期するのに、カーネルクロックの 3 クロックサイクルの遅延が必要となります。
- 非同期リセットメカニズム：非同期リセットは、LPTIM_CR レジスタにある RSTARE ビットによって制御されます。このビットが“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読出しアクセスによって、その内容がゼロにリセットされます。非同期リセットは、LPTIM コアクロックが提供されない時間枠内にトリガされるべきです。たとえば、LPTIM 入力 1 が外部クロックソースとして使用されているとき、非同期リセットは、LPTIM 入力 1 に反転が起きないという十分な備えがあるときにのみ適用すべきです。
  LPTIM_CNT レジスタの内容を信頼できるように読み出すためには、2 回の連続した読出しアクセスを実行して比較する必要があることに注意してください。2 回の読出しアクセスで得られた値が同じであるとき、各読出しアクセスは信頼できると考えられます。しかし残念ながら、非同期リセットが有効になっているとき、LPTIM_CNT レジスタを 2 度読み出すことはできません。

警告： LPTIM 内部には、2 つのリセットメカニズムを同時に使用することを防ぐメカニズムはありません。したがって、開発者はこれらの 2 つのメカニズムを排他的に使用することを確実にするべきです。

32.4.14 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが 0 と LPTIM_ARR レジスタでプログラムされた自動再ロード値の間で（方向に応じて、0 から ARR まで、または ARR から 0 まで）連続的にカウントすることを意味します。したがって、カウンタの開始前に LPTIM_ARR を設定する必要があります。Input1 と Input2 の 2 つの外部入力信号から、LPTIM カウンタを駆動するクロック信号が生成されます。この 2 つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIM が内部クロックソースからクロック供給されるときにのみ使用できます。Input1 と Input2 の両方の信号周波数は、LPTIM 内部クロック周波数を 4 分周したものを超えてはなりません。これは、LPTIM の正しい動作を保証するために必要です。

方向の変更は、LPTIM_ISR レジスタの Down と Up の 2 つのフラグによって通知されます。また、DOWNIE ビットを通じて有効化された場合、両方の方向変更イベントで割込みを生成できます。

エンコーダモードを有効にするには、ENC ビットを 1 にセットする必要があります。LPTIM を、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIM カウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウント方向は、Up および Down フラグによって通知され、エンコーダのロータの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ検出に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組み合わせを示します（Input1 と Input2 は同時に切り替わらないと想定しています）。

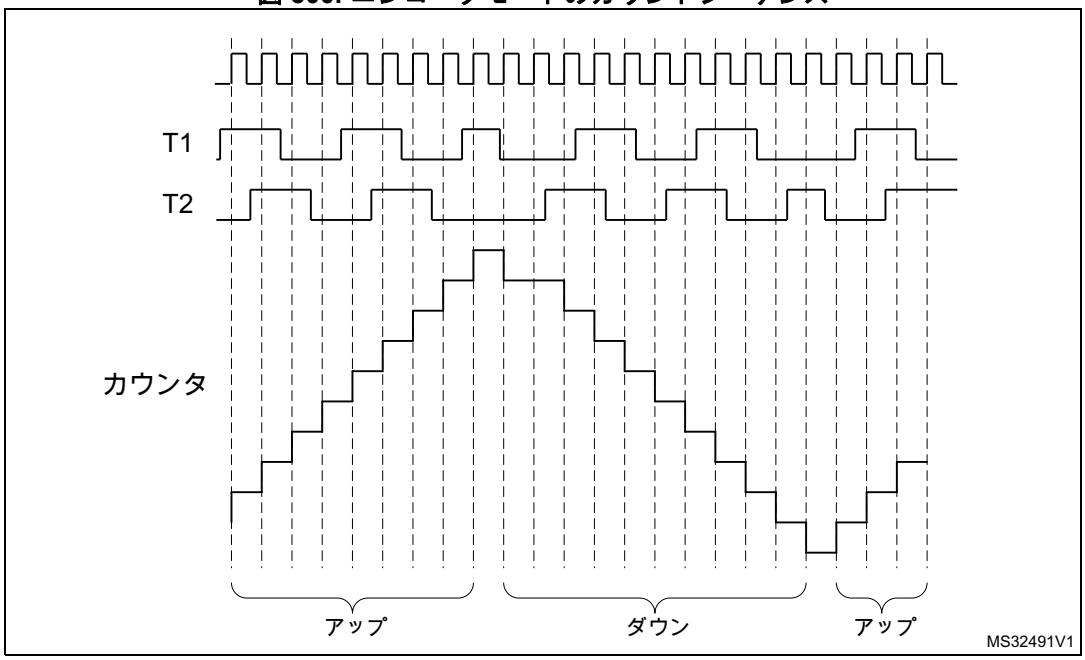
表 315. エンコーダのカウントシナリオ

アクティブエッジ	他方の信号のレベル（Input2 に対する Input1、Input1 に対する Input2）	Input1 信号		Input2 信号	
		立ち上がり	立ち下がり	立ち上がり	立ち下がり
立ち上がりエッジ	ハイ	ダウン	カウントなし	アップ	カウントなし
	ロー	アップ	カウントなし	ダウン	カウントなし
立ち下がりエッジ	ハイ	カウントなし	アップ	カウントなし	ダウン
	ロー	カウントなし	ダウン	カウントなし	アップ
両エッジ	ハイ	ダウン	アップ	アップ	ダウン
	ロー	アップ	ダウン	ダウン	アップ

次の図に、両方のエッジ検出が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： **このモードでは、LPTIM のクロックは内部クロックソースによって供給される必要があるため、CKSEL ビットをリセット値（0）に維持する必要があります。また、プリスケーラの分周比はリセット値である 1 に等しくなければなりません（PRESC[2:0] ビットが 000 である必要があります）。**

図 500. エンコーダモードのカウントシーケンス



32.4.15 デバッグモード

マイクロコントローラがデバッグモードになると（コアは停止状態）、LPTIM カウンタは、DBG モジュールの DBG_LPTIM_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

32.5 LPTIM 低電力モード

表 316. 低電力モードが LPTIM に与える影響

モード	説明
SLEEP	影響はありません。LPTIM 割込みによって、デバイスは SLEEP モードを終了します。
低電力 RUN	影響はありません。
低電力 SLEEP	影響はありません。LPTIM 割込みによって、デバイスは低電力 SLEEP モードから復帰します。
STOP 0 / STOP 1	LPTIM のクロックが LSE または LSI によって供給される場合、影響はありません。LPTIM 割込みによって、デバイスは STOP 0 および STOP 1 から復帰します。
STANDBY	LPTIM ペリフェラルはパワーダウンされ、STANBY または SHUTDOWN モード終了後に再初期化する必要があります。
SHUTDOWN	

32.6 LPTIM 割込み

LPTIM_IER レジスタで有効化されていた場合、次のイベントが発生すると、割込み／ウェイクアップイベントが生成されます。

- 比較一致
- 自動再ロード一致（エンコーダモードの場合は方向にかかわらず）
- 外部トリガイベント
- 自動再ロードレジスタへの書込み完了
- 比較レジスタへの書込み完了
- 方向変更（エンコーダモード）、プログラム可能（アップ／ダウン／両方）

注： **LPTIM_ISR レジスタ（ステータスレジスタ）の対応するフラグがセットされた後で LPTIM_IER レジスタ（割込み有効レジスタ）のビットがセットされた場合、割込みはアサートされません。**

表 317. 割込みイベント

割込みイベント	説明
比較一致	カウンタレジスタ（LPTIM_CNT）の内容が比較レジスタ（LPTIM_CMP）の内容と一致したときに、割込みフラグが立ちます。
自動再ロード一致	カウンタレジスタ（LPTIM_CNT）の内容が自動再ロードレジスタ（LPTIM_ARR）の内容と一致したときに、割込みフラグが立ちます。
外部トリガイベント	外部トリガイベントが検出されたときに、割込みフラグが立ちます。
自動再ロードレジスタ更新 OK	LPTIM_ARR レジスタへの書込み動作が完了したときに、割込みフラグが立ちます。
比較レジスタ更新 OK	LPTIM_CMP レジスタへの書込み動作が完了したときに、割込みフラグが立ちます。
方向の変更	エンコーダモードで使用されます。方向の変更を通知するために次の 2 つの割込みフラグが内蔵されています。 – UP フラグはアップカウント方向への変更を通知します。 – DOWN フラグはダウンカウント方向への変更を通知します。

32.7 LPTIM レジスタ

ペリフェラルレジスタには、ワード（32-bit）単位でのみアクセスできます。

32.7.1 LPTIM 割込みおよびステータス・レジスタ（LPTIM_ISR）

アドレスオフセット：0x000

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN	UP	ARR OK	CMP OK	EXT TRIG	ARRM	CMPM
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWN**：カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWN ビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。DOWN フラグは、LPTIM_ICR レジスタの DOWNCF ビットに 1 を書き込むことでクリアできます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3：LPTIM の実装を参照してください。

ビット 5 **UP**：カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UP ビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。UP フラグは、LPTIM_ICR レジスタの UPCF ビットに 1 を書き込むことでクリアできます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3：LPTIM の実装を参照してください。

ビット 4 **ARROK**：自動再ロード・レジスタ更新 OK

ARROK は、LPTIM_ARR レジスタへの APB バス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARROK フラグは、LPTIM_ICR レジスタの ARROKCF ビットに 1 を書き込むことでクリアできます。

ビット 3 **CMPOK**：比較レジスタ更新 OK

CMPOK は、LPTIM_CMP レジスタへの APB バス書込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CMPOK フラグは、LPTIM_ICR レジスタの CMPOKCF ビットに 1 を書き込むことでクリアできます。

ビット 2 **EXTTRIG**：外部トリガエッジイベント

EXTTRIG は、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。EXTTRIG フラグは、LPTIM_ICR レジスタの EXTTRIGCF ビットに 1 を書き込むことでクリアできます。

ビット 1 **ARRM**：自動再ロード一致

ARRM は、LPTIM_CNT レジスタの値が LPTIM_ARR レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。ARRM フラグは、LPTIM_ICR レジスタの ARRMCF ビットに 1 を書き込むことでクリアできます。

ビット 0 **CMPM**：比較一致

CMPM は、LPTIM_CNT レジスタの値が LPTIM_CMP レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。CMPM フラグは、LPTIM_ICR レジスタの CMPMCF ビットに 1 を書き込むことでクリアできます。

32.7.2 LPTIM 割込みクリア･レジスタ（LPTIM_ICR）

アドレスオフセット：0x004

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF	UPCF	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
									w	w	w	w	w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNCF**：ダウンへの方向変更フラグ･クリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの DOWN フラグがクリアされます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3 : LPTIM の実装を参照してください。

ビット 5 **UPCF**：アップへの方向変更フラグ･クリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの UP フラグがクリアされます。

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3 : LPTIM の実装を参照してください。

ビット 4 **ARROKCF**：自動再ロードレジスタ更新 OK クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの ARROK フラグがクリアされます。

ビット 3 **CMPOKCF**：比較レジスタ更新 OK クリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの CMPOK フラグがクリアされます。

ビット 2 **EXTTRIGCF**：外部トリガ有効エッジクリアフラグ

このビットに 1 を書き込むと、LPTIM_ISR レジスタの EXTTRIG フラグがクリアされます。

ビット 1 **ARRMCF**：自動再ロード一致フラグ･クリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの ARRM フラグがクリアされます。

ビット 0 **CMPMCF**：比較一致フラグ･クリア

このビットに 1 を書き込むと、LPTIM_ISR レジスタの CMPM フラグがクリアされます。

32.7.3 LPTIM 割込み有効レジスタ（LPTIM_IER）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE	UPIE	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DOWNIE**：ダウンへの方向変更割込み有効化

0： DOWN 割込み無効

1： DOWN 割込み有効

注： LPTIM がエンコーダ･モード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3: LPTIM の実装を参照してください。

ビット 5 **UPIE**：アップへの方向変更割込み有効化

0： UP 割込み無効

1： UP 割込み有効

注： LPTIM がエンコーダ･モード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3: LPTIM の実装を参照してください。

ビット 4 **ARROKIE**：自動再ロードレジスタ更新 OK 割込み有効化

0： ARROK 割込み無効

1： ARROK 割込み有効

ビット 3 **CMPOKIE**：比較レジスタ更新 OK 割込みイネーブル

0： CMPOK 割込み無効

1： CMPOK 割込み有効

ビット 2 **EXTTRIGIE**：外部トリガ有効エッジ割込み有効化

0： EXTTRIG 割込み無効

1： EXTTRIG 割込み有効

ビット 1 **ARRMIE**：自動再ロード一致割込み有効化

0： ARRM 割込み無効

1： ARRM 割込み有効

ビット 0 **CMPMIE**：比較一致割込み有効化

0： CMPM 割込み無効

1： CMPM 割込み有効

注意： LPTIM_IER レジスタの変更は、LPTIM が無効（ENABLE ビットが 0 にリセットされている）のときにのみ行う必要があります。

32.7.4 LPTIM 設定レジスタ（LPTIM_CFGR）

アドレスオフセット：0x00C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	TRIGSEL[3]	Res.	Res.	Res.	Res.	ENC	COUNTMODE	PRELOAD	WAVPOL	WAVE	TIMOUT	TRIGEN[1:0]		Res.
		rw					rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIGSEL[2:0]			Res.	PRESC[2:0]			Res.	TRGFLT[1:0]		Res.	CKFLT[1:0]		CKPOL[1:0]		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 28:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **ENC**：エンコーダ･モード有効化

ENC ビットは、エンコーダモードを制御します。

0： エンコーダモード無効

1： エンコーダモード有効

注： LPTIM がエンコーダモード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3：LPTIM の実装を参照してください。

ビット 23 **COUNTMODE**：カウンタモード有効化

COUNTMODE ビットは、LPTIM がカウンタのクロックに使用するクロックソースを選択します。

0：カウンタは各内部クロックのパルスに従ってインクリメントされます。

1：カウンタは LPTIM 外部 Input1 の各有効なクロックパルスに従ってインクリメントされます。

ビット 22 **PRELOAD**：レジスタ更新モード

PRELOAD ビットは、LPTIM_ARR および LPTIM_CMP レジスタの更新方法を制御します。

0： レジスタは、各 APB バス書込みアクセス後に更新されます。

1： レジスタは、現在の LPTIM 周期の終了時に更新されます。

ビット 21 **WAVPOL**：波形極性

WAVEPOL ビットは、出力の極性を制御します。

0： LPTIM 出力は LPTIM_CNT レジスタと LPTIM_CMP レジスタの比較結果を反映します。

1： LPTIM 出力は LPTIM_CNT レジスタと LPTIM_CMP レジスタの比較結果を反転した結果を反映します。

ビット 20 **WAVE**：波形

WAVE ビットは、出力波形を制御します。

0： セットワンス･モードを非アクティブにします。

1： セットワンスモードをアクティブにします。

ビット 19 **TIMOUT**：タイムアウト有効化

TIMOUT ビットは、タイムアウト機能を制御します。

0： タイマがすでに開始しているときに着信したトリガイベントは無視されます。

1： タイマがすでに開始しているときにトリガイベントが着信すると、カウンタがリセットされ、再開始します。

ビット 18:17 **TRIGEN[1:0]**：トリガ有効化および極性

TRIGEN ビットは、LPTIM カウンタが外部トリガによって開始されるかどうかを制御します。外部トリガオプションが選択された場合、トリガのアクティブエッジについて 3 つの構成が可能です。

00：ソフトウェアトリガ（カウントの開始はソフトウェアによって行われます）。

01：立ち上がりエッジがアクティブエッジです。

10：立ち下がりエッジがアクティブエッジです。

11：両方のエッジがアクティブエッジです。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 29、15、14、13 **TRIGSEL[3:0]**：トリガセレクタ

TRIGSEL ビットは、次の 13 つのソースの中のLPTIM のトリガイベントの役目を果たすトリガソースを選択します。

0000：lptim_ext_trig0

0001：lptim_ext_trig1

0010：lptim_ext_trig2

0011：lptim_ext_trig3

0100：lptim_ext_trig4

0101：lptim_ext_trig5

0110：lptim_ext_trig6

0111：lptim_ext_trig7

1000：lptim_ext_trig8

1001：lptim_ext_trig9

1010：lptim_ext_trig10

1011：lptim_ext_trig11

1100：lptim_ext_trig12

その他：予約済みです。

詳細については、セクション 32.4.2：LPTIM 入力およびトリガマッピングを参照してください。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:9 **PRESC[2:0]**：クロックプリスケーラ

PRESC ビットは、プリスケーラ分周比を設定します。次の分周比から選択できます。

000：/1

001：/2

010：/4

011：/8

100：/16

101：/32

110：/64

111：/128

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7:6 **TRGFLT[1:0]**：設定可能なトリガ用デジタルフィルタ

TRGFLT 値は、有効なレベル遷移とみなされる前に、内部トリガでレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：トリガのアクティブレベル変更は、有効なトリガとみなされます。

01：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:3 **CKFLT[1:0]**：設定可能な外部クロック用デジタルフィルタ

CKFLT 値は、有効なレベル遷移とみなされる前に、外部クロック信号でレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

00：外部クロック信号のレベル変更は、有効な遷移とみなされます。

01：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。

10：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。

11：外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 2:1 **CKPOL[1:0]**：クロック極性

LPTIM のクロックが、外部クロックソースによって供給される場合：

LPTIM のクロックが外部クロックソースによって供給されるとき、CKPOL ビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。

00：立ち上がりエッジが、カウントに使用されるアクティブエッジです。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 1 がアクティブです。

01：立ち下がりエッジが、カウントに使用されるアクティブエッジです。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 2 がアクティブです。

10：両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIM のクロックは内部クロックソースからも供給される必要があり、その周波数は外部クロック周波数の 4 倍以上である必要があります。
LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合、エンコーダサブモード 3 がアクティブです。

11：使用できません。

エンコーダモードのサブモードの詳細については、セクション 32.4.14：エンコーダモードを参照してください。

ビット 0 **CKSEL**：クロック･セレクタ

CKSEL ビットは、LPTIM が使用するクロックソースを選択します：

0： LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。

1： LPTIM のクロックは、LPTIM 外部 Input1 を通じて外部クロックソースによって供給されます。

注意： LPTIM_CFGR レジスタの変更は、LPTIM が無効（イネーブルビットが 0 にリセットされている）のときにのみ行う必要があります。

32.7.5 LPTIM 制御レジスタ（LPTIM_CR）

アドレスオフセット：0x010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
											rw	rs	rw	rw	rw

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **RSTARE**：読出し後リセット有効化

このビットは、ソフトウェアによってセット／クリアされます。RSTARE が“1”にセットされると、LPTIM_CNT レジスタに対するあらゆる読出しアクセスによって、LPTIM_CNT レジスタの内容が非同期的にリセットされます。

このビットをセットできるのは、LPTIM が有効なときだけです。

ビット 3 **COUNTRST**：カウンタリセット

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。“1”にセットされると、このビットは LPTIM_CNT カウンタレジスタの同期リセットをトリガします。このリセットの同期的な性質のため、3 LP タイマコアクロックサイクルの同期的遅延の後にのみ、この状況が発生します（LP タイマコアクロック は APB クロックと異なる場合があります）。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にクリアされます。

注意： COUNTRST をハードウェアで“0”にクリアしていない段階で、ソフトウェアで“1”にセットしないでください。したがって、ソフトウェアで COUNTRST ビットを“1”にセットしようとする前に、それがすでに“0”にクリアされていることを確認するべきです。

ビット 2 **CNTSTRT**：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM は連続モードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。

シングルパルスモードでのカウント中にこのビットがセットされた場合、LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときにタイマは停止せず、LPTIM カウンタは連続モードでのカウントを続行します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 1 **SNGSTRT**：シングルモードで LPTIM 開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM はシングルパルスモードで開始します。

ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIMはシングルパルスモードで開始します。

LPTIM が連続カウントモードのときにこのビットがセットされた場合、LPTIM は LPTIM_ARR レジスタと LPTIM_CNT レジスタが次に一致したときに停止します。

このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 0 **ENABLE**：LPTIM 有効化

ENABLE ビットは、ソフトウェアによってセットされ、またクリアされます。

0 :LPTIM は無効です。

1 :LPTIM は有効です。

32.7.6 LPTIM 比較レジスタ（LPTIM_CMP）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CMP[15:0]**：比較値

CMP は、LPTIM によって使用される比較値です。

注意： **LPTIM_CMP レジスタの変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。**

32.7.7 LPTIM 自動再ロードレジスタ（LPTIM_ARR）

アドレスオフセット：0x018

リセット値：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ARR[15:0]**：自動再ロード値

ARR は、LPTIM の自動再ロード値です。

この値は、CMP[15:0] 値より大きくなければなりません。

注意： **LPTIM_ARR レジスタの変更は、LPTIM が有効（イネーブルビットが 1 にセットされている）のときにのみ行う必要があります。**

32.7.8 LPTIM カウンタレジスタ（LPTIM_CNT）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **CNT[15:0]**：カウンタ値

LPTIM が非同期クロックで実行しているとき、LPTIM_CNT レジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2 つの連続した読出しアクセスを実行して、返された 2 つの値が同じかどうかを確認する必要があります。

信頼できる LPTIM_CNT レジスタの読出しアクセスでは、2 回の連続した読出しアクセスを実行して比較する必要があることに注意してください。2 回の連続読出しアクセスで得られた値が同じであるとき、各読出しアクセスは信頼できると考えられます。

32.7.9 LPTIM オプション･レジスタ（LPTIM_OR）

アドレス･オフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	IN2 [2:1]		IN1 [2:1]		IN2[0]	IN1[0]
										rw	rw	rw	rw	rw	rw

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **IN2[2:1]**：LP タイマ入力 2 再配置拡張
ソフトウェアでセット／クリアされます。
00：COMP2 に接続されます。
01：COMP4 に接続されます。
10：COMP6 に接続されます。
11：COMP6 に接続されます。

ビット 3:2 **IN1[2:1]**：LP タイマ入力 1 再配置拡張
ソフトウェアでセット／クリアされます。
00：COMP1 に接続されます。
01：COMP3 に接続されます。
10：COMP5 に接続されます。
11：COMP7 に接続されます。

ビット 1 **IN2[0]**：LP タイマ入力 2 再配置
ソフトウェアでセット／クリアされます。
1：IN2[2:1] の値に従って COMP 出力に接続されます。
0：GPIO に接続されます。

ビット 0 **IN1[0]**：LP タイマ入力 1 再配置
ソフトウェアでセット／クリアされます。
1：IN1[2:1] の値に従って COMP 出力に接続されます。
0：GPIO に接続されます。

32.7.10 LPTIM レジスタマップ

次の表に LPTIM レジスタの一覧を示します。

表 318. LPTIM レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	LPTIM_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWN(1)	UP(1)	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
	リセット値																										0	0	0	0	0	0	0
0x004	LPTIM_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNCF(1)	UPCF(1)	ARROKCF	CMPOKCF	EXTTRIGCF	ARRMCF	CMPMCF
	リセット値																										0	0	0	0	0	0	0
0x008	LPTIM_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DOWNIE(1)	UPIE(1)	ARROKIE	CMPOKIE	EXTTRIGIE	ARRMIE	CMPMIE
	リセット値																										0	0	0	0	0	0	0
0x00C	LPTIM_CFGR	Res.	Res.	TRIGSEL[3]	Res.	Res.	Res.	Res.	ENC(1)	COUNTMODE	PRELOAD	WAVEPOL	WAVE	TIMOUT	TRIGEN		Res.	TRIGSEL[2:0]			Res.	PRESC			Res.	TRGFLT		Res.	CKFLT		CKPOL		CKSEL	
	リセット値			0					0	0	0	0	0	0	0	0		0	0	0		0	0	0		0	0		0	0	0	0	0	0
0x010	LPTIM_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RSTARE	COUNTRST	CNTSTRT	SNGSTRT	ENABLE
	リセット値																												0	0	0	0	0
0x014	LPTIM_CMP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CMP[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	LPTIM_ARR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x01C	LPTIM_CNT	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	LPTIM_OR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.		IN2 [2:1]		IN1 [2:1]		IN2[0]	IN1[0]
	リセット値																											0	0	0	0	0	0

1. LPTIM がエンコーダ·モード機能をサポートしない場合、このビットは予約済みとなります。セクション 32.3 : LPTIM の実装を参照してください。

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

33 赤外線インタフェース（IRTIM）

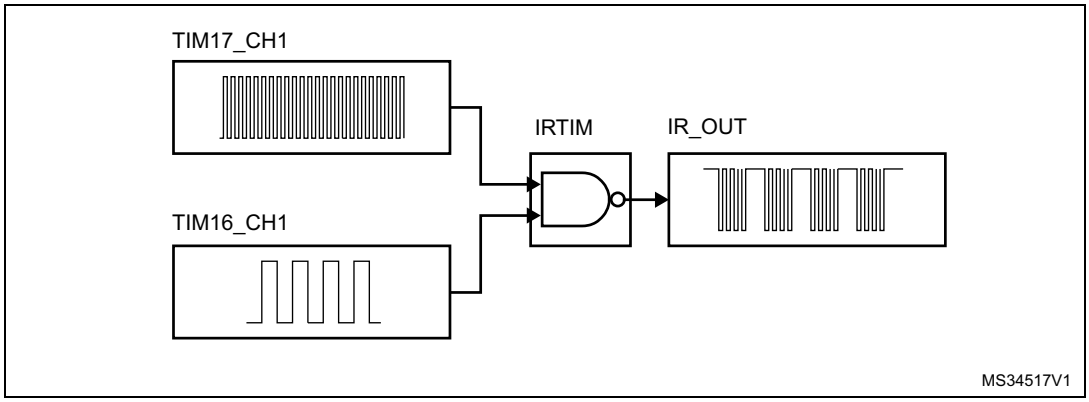
このデバイスではリモートコントロールで赤外線インタフェース（IRTIM）を使用できます。リモートコントロール機能を実行するために、赤外線 LED と併せて使用できます。

図 501 に示すように、TIM16 および TIM17 との間で内部接続を使用します。

赤外線リモート･コントロール信号を生成するには、正しい波形を生成するために IR インタフェースを有効にし、TIM16 チャネル 1 （TIM16_OC1）と TIM17 チャネル 1 （TIM17_OC1）を適切に設定する必要があります。

赤外線レシーバは、基本的な入力キャプチャモードで簡単に実装できます。

図 501. TIM16 および TIM17 との IRTIM 内部ハードウェア接続



標準の IR パルス変調モードはすべて 2 つのタイマ出力比較チャネルをプログラミングすることで取得できます。

TIM16 は変調包絡線を生成し、TIM17 は高周波キャリア信号を生成するために使用されます。

赤外線機能は IR_OUT ピンから出力されます。この機能の有効化は、GPIOx_AFRx レジスタを通じ、関連するオルタネート機能ビットを有効にすることで行われます。

大電流シンク LED ドライバ機能（PB9 ピンと PA13 ピンでのみ使用可能）の有効化は SYSCFG_CFGR1 レジスタの I2C_PB9_FMP ビットを通じて行われ、赤外線 LED を直接制御するために必要な大電流を流すために使用されます。

34 AES ハードウェア･アクセラレータ（AES）

34.1 概要

AES ハードウェア･アクセラレータ（AES）は、連邦情報処理規格公報（FIPS）197 によって規定されている高度暗号化標準（AES：Advanced Encryption Standard）に完全に準拠したアルゴリズムと実装を使用してデータを暗号化および復号します。

このペリフェラルは、128 ビットまたは 256 ビットのキー･サイズを使用する CTR、GCM、GMAC、CCM、ECB、CBC の各連鎖モードをサポートします。

AES は、32 ビットのシングル･アクセスでのみアクセス可能な AMBA AHB スレーブ･ペリフェラルです。他の方法でアクセスすると AHB エラーが発生し、32 ビット書込み以外では、レジスタの内容が破損する可能性があります。

ペリフェラルは、入力および出力データの DMA シングル転送をサポートします（2 つの DMA チャネルが必要です）。

34.2 AES の主な特徴

- 2001 年 11 月発効の NIST『高度暗号化標準（AES）、FIPS 公報 197』に準拠
- 128 ビットデータブロックの処理
- 128 ビットおよび 256 ビットの暗号キー長に対応
- 次の複数の連鎖モードで暗号化と復号化：
  - 電子コードブック（ECB）モード
  - 暗号ブロック連鎖（CBC）モード
  - カウンタ（CTR）モード
  - ガロアカウンタモード（GCM）
  - ガロアメッセージ認証コード（GMAC）モード
  - CBC-MAC 付きカウンタ（CCM）モード
- 128 ビットまたは 256 ビットのキーを使用して 1 個の 128 ビット･ブロックのデータを処理する場合は ECB モードでそれぞれ 51 または 75 クロック･サイクルの遅延
- ECB/CBC 復号の最終ラウンドのキーの計算に使用する内蔵ラウンド･キー･スケジューラ
- 32 ビット･ワードのシングル･アクセスでのみアクセス可能な AMBA AHB スレーブ･ペリフェラル
- 暗号化キーを保存するための 256 ビット･レジスタ（8 個の 32 ビット･レジスタ）
- 初期化ベクタを保存するための 128 ビット･レジスタ（4 個の 32 ビット･レジスタ）
- データ入力および出力用の 32 ビット･バッファ
- 2 つのチャネル（1 チャネルは入力データ用、1 チャネルは処理済みデータ用）を使用したシングル転送ダイレクトメモリアクセス（DMA）のサポートによる自動データフロー制御
- 1、8、16、32 ビットデータをサポートするデータスワッピングロジック
- AES がより優先順位の高い別メッセージを処理する必要がある場合は、ソフトウェアでメッセージをサスペンドしてから元のメッセージのレジュームが可能

34.3 AES の実装

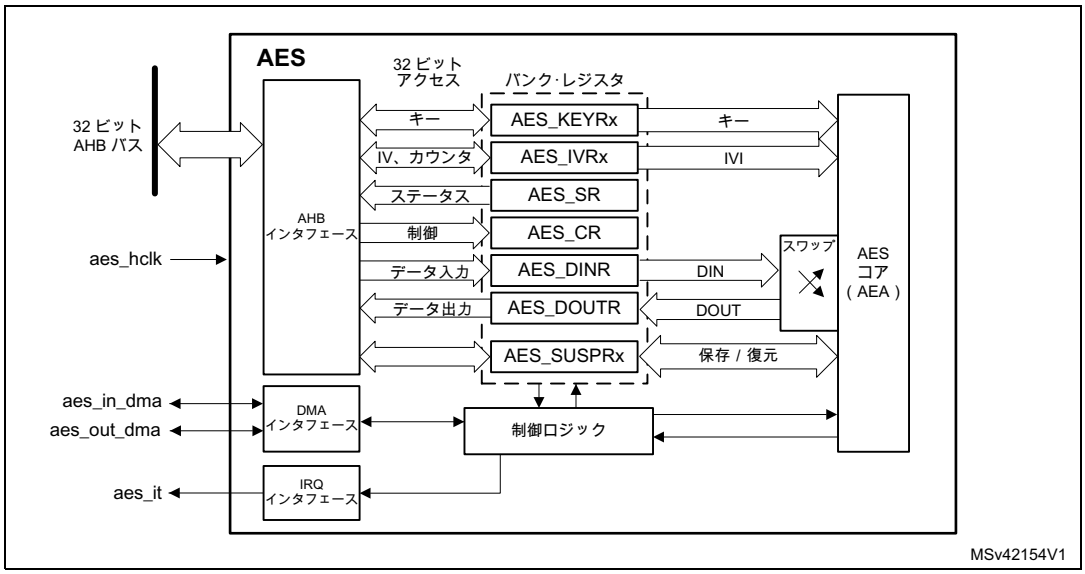
このデバイスは AES ペリフェラルを 1 つ搭載しています。

34.4 AES 機能詳細

34.4.1 AES ブロック図

図 502 に AES のブロック図を示します。

図 502. AES ブロック図



34.4.2 AES 内部信号

表 319 には、AES ペリフェラルにインタフェース接続するユーザ関連内部信号を示しています。

表 319. AES 内部入力／出力信号

信号名	信号タイプ	説明
aes_hclk	入力	AHBバス･クロック
aes_it	出力	AES 割込みリクエスト
aes_in_dma	入力／出力	入力 DMA シングル･リクエスト／確認応答
aes_out_dma	入力／出力	出力 DMA シングル･リクエスト／確認応答

34.4.3 AES 暗号コア

概要

AES 暗号コアは、次のコンポーネントで構成されています。

- AES コア･アルゴリズム（AEA）
- バイナリ･ガロア･フィールド（GF2mul）の乗算器
- キー入力
- 初期化ベクタ（IV）入力
- 連鎖アルゴリズム･ロジック（XOR、フィードバック／カウンタ、マスク）

AES コアは 128 ビットまたは 256 ビットのキー長を使用して 128 ビット･データ･ブロック（4 ワード）で動作します。連鎖モードに応じて、AESは 0 個または 1 個の 128 ビット初期化ベクタ（IV）を必要とします。

AES には、次の動作モードがあります。

- **モード 1 :**

  AES_KEYRx レジスタに格納されたキーを使用して平文を暗号化します。
- **モード 2 :**

  ECB または CBC の復号化キーを準備します。ECB または CBC の連鎖モードでモード 3 を選択する前に使用する必要があります。復号化のために準備されたキーは、AES_KEYRx レジスタに自動的に格納されます。これで、AES ペリフェラルをモード 3 に切り替えて、データ復号化を実行する準備ができます。
- **モード 3 :**

  AES_KEYRx レジスタに格納されたキーを使用して暗号文を復号化します。ECB および CBC の連鎖モードを選択した場合、モード 2 で事前にキーを準備する必要があります。
- **モード 4 :**

  AES_KEYRx レジスタに保存されたキーを使った、ECB または CBC 暗号文の単一キー復号化（初期化キーは自動的に派生します）

注 : **モード 2 およびモード 4 は ECB および CBC 復号化を行うときのみ使用されます。**

モード 4 の選択では 復号化できるのは 1 件だけなので、代わりにモード 2 か モード 3 の使用が推奨されます。

動作モードは、AES_CR レジスタの MODE[1:0] ビットフィールドをプログラムして選択します。この操作は、AES ペリフェラルが無効である場合のみ実行できます。

標準的データ処理

AES の標準的な使用方法については1466 ページのセクション 34.4.4 :暗号操作を行うための AES の手順に説明があります。

注 : **IVI ビットフィールドを除き、中間 AEA ステージの出力が暗号境界の外に漏れることは決してありません。**

連鎖モード

次の連鎖モードが AES でサポートされており、AES_CR レジスタの CHMOD[2:0] ビットフィールドで選択します。

- 電子コードブック（ECB）
- 暗号ブロック連鎖（CBC）
- カウンタ（CTR）
- ガロアカウンタモード（GCM）
- ガロアメッセージ認証コード（GMAC）
- CBC-MAC付きカウンタ（CCM）

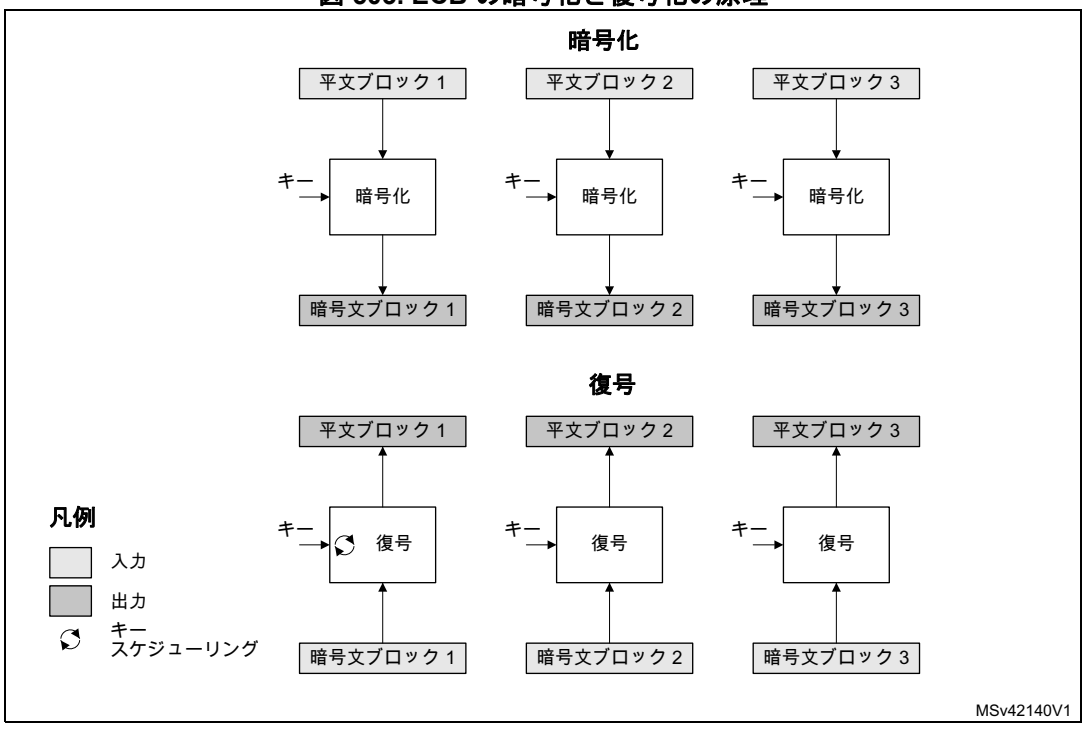
注： **連鎖モードは、AES が無効である（AES_CR レジスタの EN ビットがクリアされている）場合のみ変更できます。**

各 AES 連鎖モードの原理については、以降の項目に記載されています。

詳細については、セクション 34.4.8 : AES 基本連鎖モード（ECB、CBC）以降の専用セクションに示されています。

電子コードブック（ECB）モード

図 503. ECB の暗号化と復号化の原理

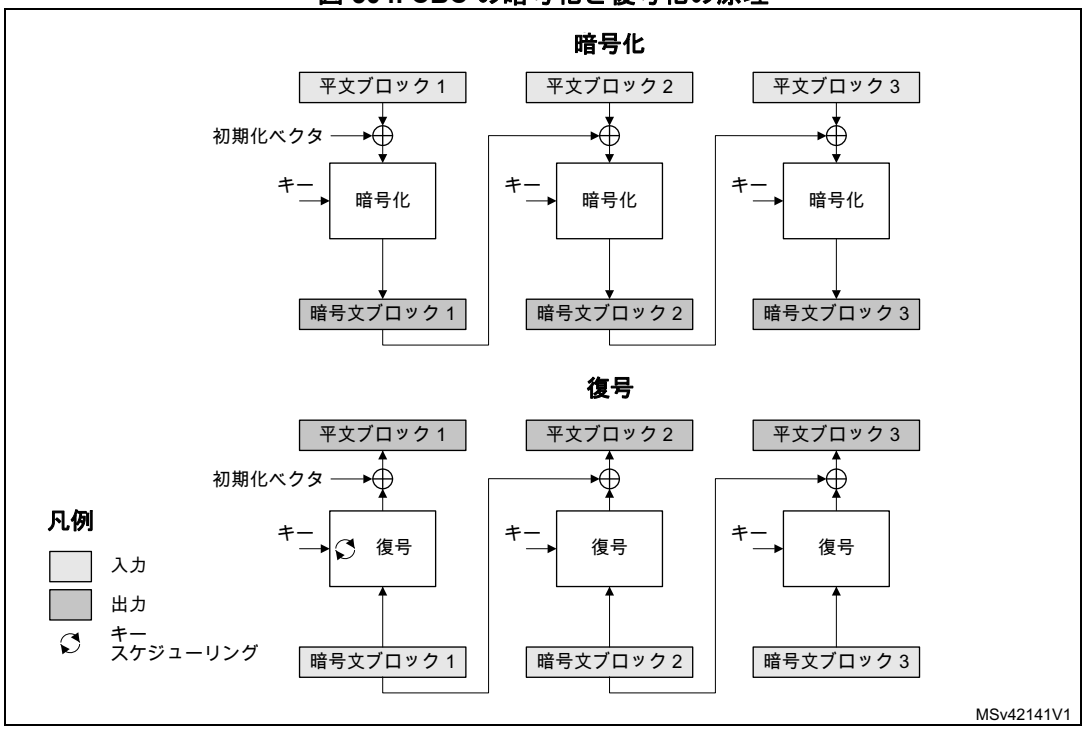


ECB は最も単純な動作モードです。連鎖操作も、特別な初期化ステージもありません。メッセージはブロックに分割され、各ブロックが個別に暗号化または復号化されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

暗号ブロック連鎖（CBC）モード

図 504. CBC の暗号化と復号化の原理

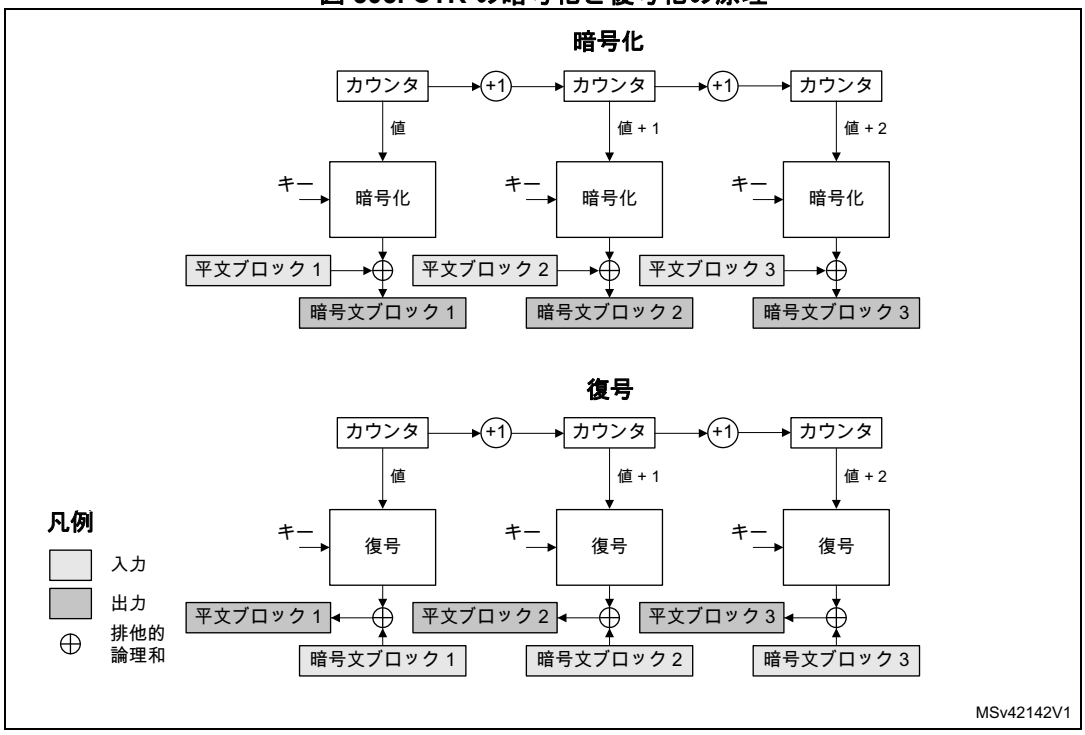


CBC モードでは、各ブロックの出力が次のブロックの入力で連鎖されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクタが使用されます。

注： **復号化では、最初のブロックを処理する前に特別なキースケジューリングが必要となります。**

カウンタ（CTR）モード

図 505. CTR の暗号化と復号化の原理

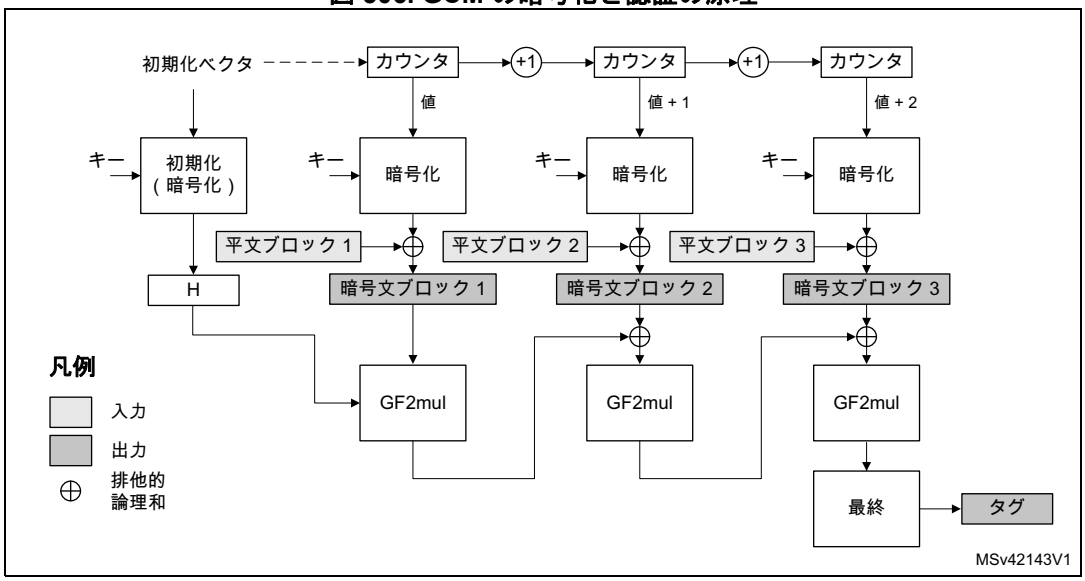


CTR モードでは、AES コアを使用してキーストリームを生成します。次に、このキーと平文との排他論理和演算によって、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード』で指定されている暗号文が得られます。

注： **この連鎖スキームでは、キー･ストリームまたはカウンタ･ブロックの生成に AES コアが暗号化モードで必ず使用されるため、ECB モードや CBC モードとは異なり、CTR の復号にキー･スケジューリングは必要ありません。**

ガロア／カウンタモード（GCM）

図 506. GCM の暗号化と認証の原理

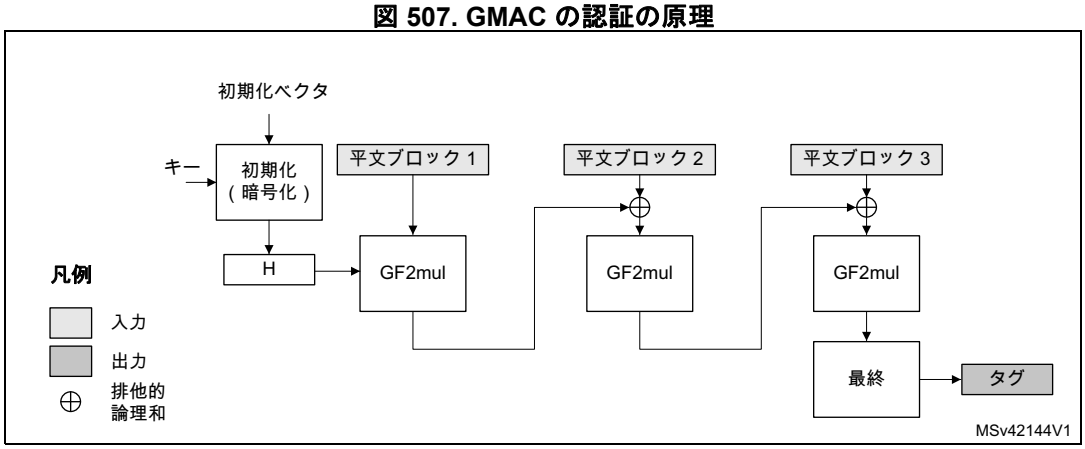


ガロア／カウンタモード（GCM）では、平文メッセージが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文とその MAC（認証タグとも言います）が生成されます。これは、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』の中に定義されています。

GCM モードは、機密性のためにカウンタモードでの AES に基づいています。固定の有限体に乗算器を使用して、メッセージ認証コードを計算します。初期値ならびにメッセージの最後に特殊な 128 ビットブロックを必要とします。

ガロアメッセージ認証コード（GMAC）の原理

図 507. GMAC の認証の原理

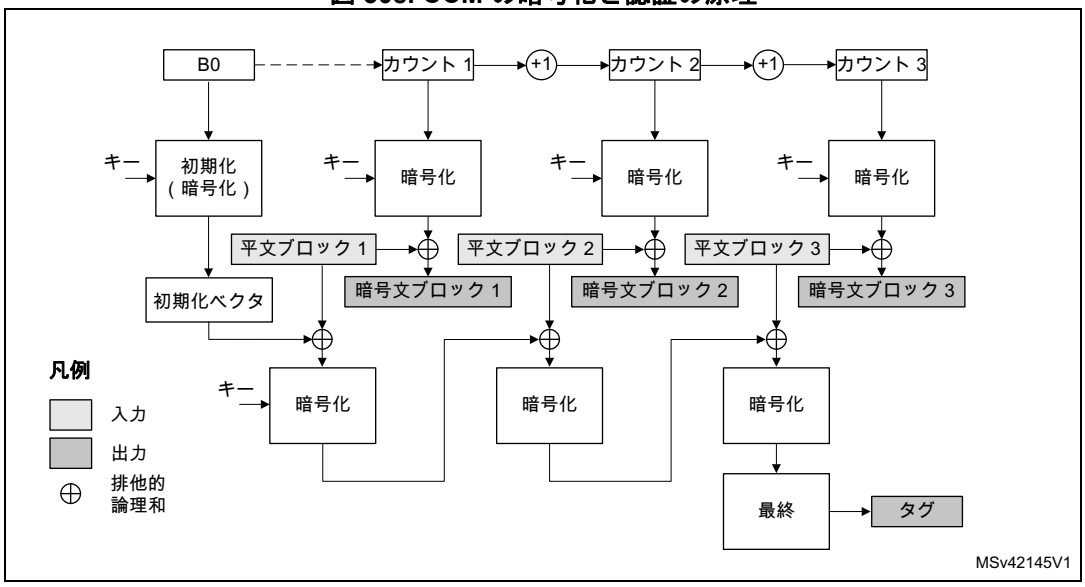


ガロアメッセージ認証コード（GMAC）を使用すると、メッセージの認証と、対応するメッセージ認証コード（MAC）の生成が可能となります。これは、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタ･モード（GCM）および GMAC』の中に定義されています。

平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみでペイロードなし）に適用されることを除けば、GMAC は GCM と似ています。

CBC-MAC 付きカウンタ（CCM）の原理

図 508. CCM の暗号化と認証の原理



暗号ブロック連鎖-メッセージ認証コード付きカウンタ（CCM）モードでは、平文メッセージが暗号化されている間に、並列でメッセージ認証コード（MAC）が計算され、対応する暗号文と対応するMAC（タグとも言います）が生成されます。これは、NIST 特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のための CCM モード』の中に説明されています。

CCM モードは、機密性のためにカウンタモードでの AES に基づいており、CBC を使用してメッセージ認証コードを計算します。初期値が必要です。

CCM 連鎖モードは、GCM のように平文の認証済みデータのみで構成されたメッセージ（すなわち、ヘッダのみでペイロードなし）に適用できます。CCM をこのように使用することは CMAC とは呼ばれず（GCM/GMACとは異なります）、その使用は NIST による推奨ではないことに注意してください。

34.4.4 暗号操作を行うための AES の手順

概要

標準的な暗号操作を以下に説明します。詳細については、セクション 34.4.8：AES 基本連鎖モード（ECB、CBC）以降のセクションに示されています。

AES の初期化

AES を初期化するには、まず AES_CR レジスタの EN ビットをクリアして無効化します。次に、任意の順番で次の手順を実行します。

- AES_CR レジスタの MODE[1:0] ビットフィールドをプログラムして、AES モードを設定します。
  - 暗号化の場合は、モード 1（MODE[1:0] = 00）を選択します。
  - 復号の場合は、ECB または CBC の連鎖モードを使用していないかぎり、モード 3（MODE[1:0] = 10）を選択します。ECB または CBC の連鎖モードの場合は、セクション 34.4.5：AES 復号ラウンド･キーの準備 に記載されているとおり、暗号化キーの初期キー導出を実行します。
- AES_CR レジスタの CHMOD[2:0] ビットフィールドをプログラムして、連鎖モードを選択します。

- AES_CR レジスタの DATATYPE[1:0] ビットフィールドでデータ型（1、8、16、32 ビット）を設定します。
- 必要であれば（CBC または CTR の連鎖モードの場合など）、初期化ベクタを AES_IVRx レジスタに書き込みます。
- AES_CR レジスタの KEYSIZE ビットフィールドでキー･サイズ（128 ビットまたは 256 ビット）を設定します。
- 対称キーを AES_KEYRx レジスタ（キー･サイズに応じて 4 個または 8 個のレジスタ）に書き込みます。

データ追加

このセクションでは、処理するデータのサイズが 128 ビットの倍数ではない場合に、処理するためにデータを追加するさまざまな方法について説明します。

ECB または CBC モードについては、セクション 34.4.6 : AES 暗号文借用（CTS）およびデータ･パディングを参照してください。これらの場合の最終ブロックの管理は、このセクションで説明しているシーケンスより複雑になります。

ポーリングでのデータ追加

この方法では、フラグポーリングを使用して次のシーケンスでデータ追加を制御します。

1. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
2. ペイロードが完全に処理されるまで次のサブシーケンスを繰り返します。
   a) AES_DINR レジスタに 4 つの入力データワードを書き込みます。
   b) ステータスフラグ CCF が AES_SR でセットされるまで待って、AES_DOUTR レジスタから 4 つのデータワードを読み出します。
   c) AES_CR レジスタの CCFC ビットをセットして CCF フラグをクリアします。
   d) 処理したばかりのデータ･ブロックがメッセージの最後から 2 番目のブロックであり、処理する最終ブロックの有効データが 128 ビットを下回る場合、最終ブロックの残り部分を 0 でパディングし、GCM のペイロード暗号化または CCM のペイロード復号の場合は、AES_CR レジスタの NPBLB ビットフィールドで、AES が正しいタグを計算できるように有効ではないバイト数を指定します。
3. これが最後のブロックであるため、データの一部ではないデータを破棄してから、AES_CR レジスタの EN ビットをクリアして AES ペリフェラルを無効にします。

注： **AES プロセッサにキーを送信できるように、AES_DINR レジスタに対する 2 つの連続する書込みの間に、ウェイト･サイクルが最大 3 つまで自動的に挿入されます。**

NPBLB ビットは、GCM、GMAC、CCM の連鎖モードのヘッダ･フェーズで使用されません。

割込みを使用したデータ追加

この方法は、AES ペリフェラルからの割込みを使用して次のシーケンスでデータ追加を制御します。

1. AES_CR レジスタの CCFIE ビットをセットして、AES からの割込みを有効にします。
2. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
3. まず、AES_DINR レジスタに 4 つの入力データワードを書き込みます。
4. AES 割込みサービスルーチンでデータを処理します。割込み時には次の手順に従います。
   a) AES_DOUTR レジスタから 4 つの出力データワードを読み出します。
   b) AES_CR レジスタの CCFC ビットをセットして、CCF フラグ、つまり保留中の割込みをクリアします。
   c) 処理したばかりのデータ･ブロックがメッセージの最後から 2 番目のブロックであり、処理する最終ブロックの有効データが 128 ビットを下回る場合、最終ブロックの残り部分を

0 でパディングし、GCM のペイロード暗号化または CCM のペイロード復号の場合は、AES_CR レジスタの NPBLB ビットフィールドで、AES が正しいタグを計算できるように有効ではないバイト数を指定します。次に、4e) に進みます。

d) 処理したばかりのデータ･ブロックがメッセージの最終ブロックだった場合、データの一部ではないデータを破棄してから、AES_CR レジスタの EN ビットをクリアして AES ペリフェラルを無効にし、割込みサービス･ルーチンを中止します。

e) AES_DINR レジスタに次の 4 つの入力データ･ワードを書き込み、割込みサービス･ルーチンを中止します。

注： **AES は連続した読出し操作または書込み操作の間の遅延を許容するため、2 つの AES 計算の間に別のペリフェラルからの割込みがある場合などに受け入れることができます。**

NPBLB ビットは、GCM、GMAC、CCM の連鎖モードのヘッダフェーズで使用されません。

DMA を使用したデータ追加

この方法では、すべての転送と処理が DMA および AES によって管理されます。この方法を使用するには、次の手順に従います。

1. ブロックの残り部分を 0 でパディングして、最後の 4 ワードのデータ･ブロックを準備します（処理するデータが完全に埋まっていない場合）。
2. 処理するデータをメモリから AES ペリフェラル入力へ、処理したデータを AES ペリフェラル出力からメモリに転送するよう DMA コントローラを設定します（セクション 34.4.16 : AES DMA インタフェース に記載）。転送完了時に割込みを生成するよう DMA コントローラを設定します。GCM のペイロード暗号化または CCM のペイロード復号で、最後の 4 ワード･ブロックが 0 でパディングされている場合は、それらを DMA 転送から**除外します**。代わりに、ポーリングでのデータ追加 に示したシーケンスをこの最終ブロックに使用する必要があります。これは AES が正しいタグを計算できるように、NPBLB ビットを設定してからブロックを処理する必要があるためです。
3. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
4. AES_CR レジスタの DMAINEN ビットと DMAOUTEN ビットをセットして、DMA リクエストを有効にします。
5. 転送完了を示す DMA 割込みが発生したら、メモリから AES 処理済みデータを取得します。

注： **AES_DOUTR レジスタの読出しは計算フェーズの最後にソフトウェア操作なしで自動的に DMA によって管理されるため、この方法では CCF フラグを使用しません。**

NPBLB ビットは、GCM、GMAC、CCM の連鎖モードのヘッダ･フェーズで使用されません。

34.4.5 AES 復号ラウンド･キーの準備

AES ラウンド･キーの生成には、内部キー･スケジュールを使用します。AES 暗号化のラウンド 0 のキーは、キー･レジスタに格納されているキーです。AES 復号は、最後のラウンド･キーを使用して開始する必要があります。暗号化キーはメモリに保存されているので、復号キーを得るには特別なキー･スケジューリングを実行する必要があります。このキー･スケジューリングは、ECB および CBC モードの AES 復号でのみ必要です。

AES_CR の MODE[1:0] ビットフィールドを 01 に設定してモード 2（キー処理のみ）を選択してから、MODE[1:0] を 10（モード 3、復号のみ）に設定して復号を進める方法が推奨されます。モード 2 の使用方法は、以下のとおりです。

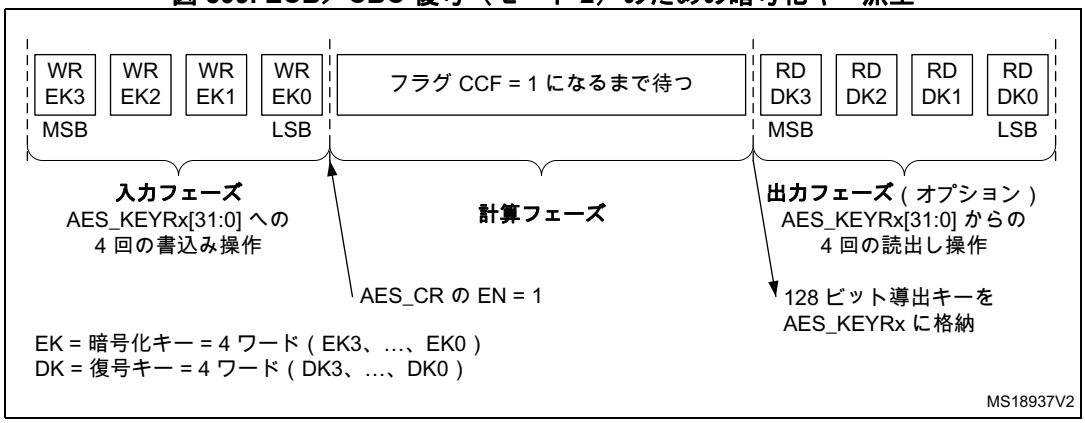
1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR の MODE[1:0] ビットフィールドを 01 にセットして、モード 2 を選択します。このキー導出モードは選択された連鎖アルゴリズムとは無関係なので、この場合、CHMOD[2:0] ビットフィールドは重要ではありません。
3. AES_CR レジスタの KEYSIZE ビットでキー長を 128 ビットまたは 256 ビットに設定します。

4. 図 509 に示すように、AES_KEYRx レジスタに暗号化キーを書き込みます（128 または 256 ビット）。AES_IVRx レジスタへの書込みは無効です。
5. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
6. AES_SR レジスタで CCF フラグがセットされるまで待ちます。
7. CCF フラグをクリアします。導出されたキーは AES コアで使用でき、復号化に使用する準備ができます。アプリケーションは、図 509 に示すように、必要に応じて AES_KEYRx レジスタもまた読み出して派生キーを取得できます（処理済みのキーは自動的に AES_KEYRx レジスタにロードされます）。

注： **導出キーが使用できるようになると、AES はハードウェアによって無効にされます。**

導出キーの計算を再開するには、手順 4、5、6、7 を繰り返します。

図 509. ECB／CBC 復号（モード 2）のための暗号化キー派生



ソフトウェアが復号用の最初のキーを保存しているとすれば、所定の暗号キーで復号化されるすべてのデータに対して、キースケジューリング操作を 1 回だけ行えば良いということになります。

注： **キー準備の操作は、キー・サイズ（128 ビットまたは 256 ビット）に応じて 59 または 82 クロック・サイクル続きます。**

34.4.6 AES 暗号文借用（CTS）およびデータ・パディング

サイズがブロックサイズ（128 ビット）の倍数ではないメッセージを管理するために ECB または CBC モードで AES を使用するときには、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード：CBC モード用暗号文借用の 3 つの変種』に記載されているもののような、暗号文借用技術を使用します。AES ペリフェラルはこのような技術をサポートしていないため、アプリケーションは最後から 2 番目のブロックのデータを使用して、入力データの最終ブロックを完成させる必要があります。

注： **暗号文借用技術は、この参考マニュアルには記載されていません。**

同様に、AES が ECB と CBC 以外のモードで使用される場合には、不完全な入力データブロック（128 ビットより短い入力データを持つブロック）を、暗号化の前に 0 でパディングする必要があります（データストリングの後端に余分なビットを付加する必要があります）。復号後は、余分なビットを破棄する必要があります。AES では、**最終ブロック**に対する自動データパディング操作が行われるため、アプリケーションが1466 ページのセクション 34.4.4：暗号操作を行うための AES の手順に示された推奨に従って、128 ビットの倍数ではないメッセージを管理する必要があります。

注： **パディングデータは AES_CR レジスタの DATATYPE[1:0] フィールドに従って、通常のデータと同様の方法で入れ替えられます（詳細は、セクション 34.4.13：AES データ・レジスタおよびデータ・スワッピングを参照してください）。**

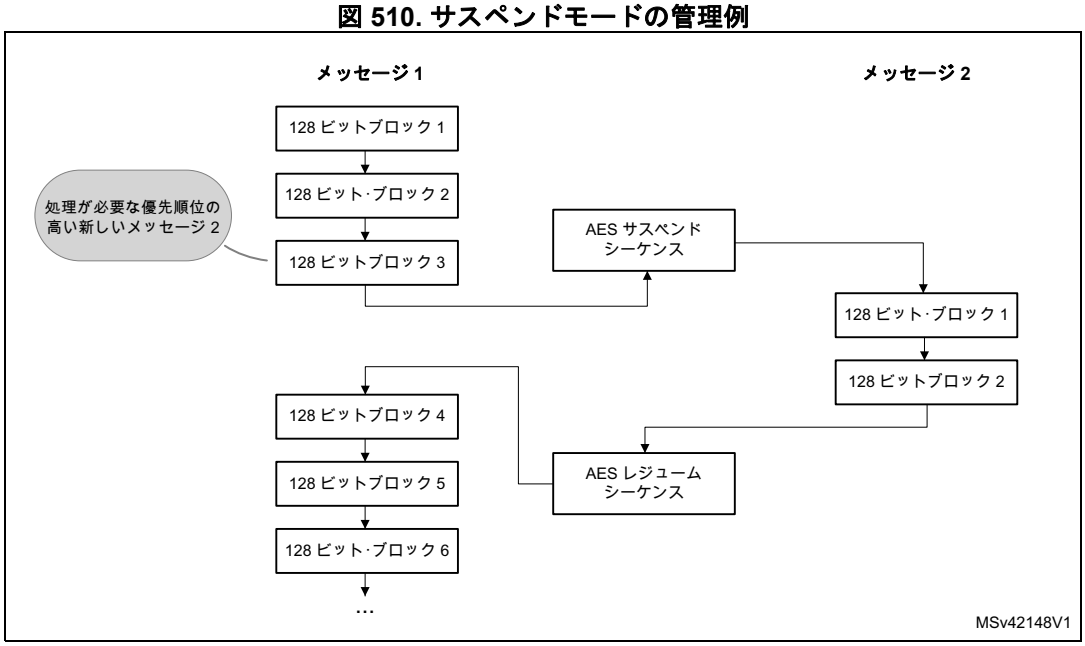
34.4.7 AES タスクのサスペンドとレジューム

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信後、サスペンドされたメッセージは暗号化モードでも復号化モードでも再開できます。

サスペンド／レジューム操作によって連鎖操作が途切れることはなく、AES が再び有効になるとすぐにメッセージ処理を再開して、次のデータブロックを受信できます。

図 510 に、サスペンド／レジューム操作の例を示します。メッセージ 1 は、より短く優先順位の高いメッセージ 2 を送信するためにサスペンドされています。

図 510. サスペンドモードの管理例



サスペンド／レジューム操作の詳細は、各 AES モード専用のセクションに説明があります。

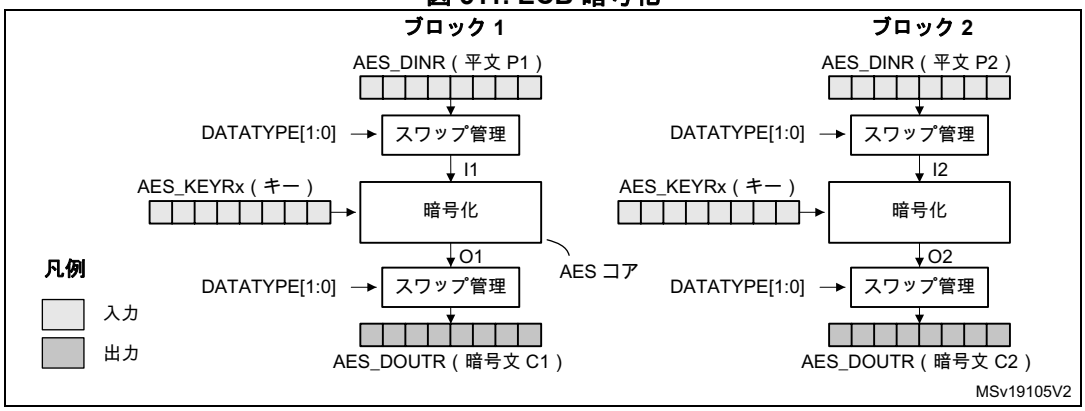
34.4.8 AES 基本連鎖モード（ECB、CBC）

概要

このセクションでは、AES コアが提供する ECB 暗号化、ECB 復号、CBC 暗号化、CBC 復号の 4 つの基本動作モードについて簡単に説明します。詳細については、2001 年 11 月 26 日発行の『FIPS 公報 197』を参照してください。

図 511 に、電子コードブック（ECB）暗号化の図を示します。

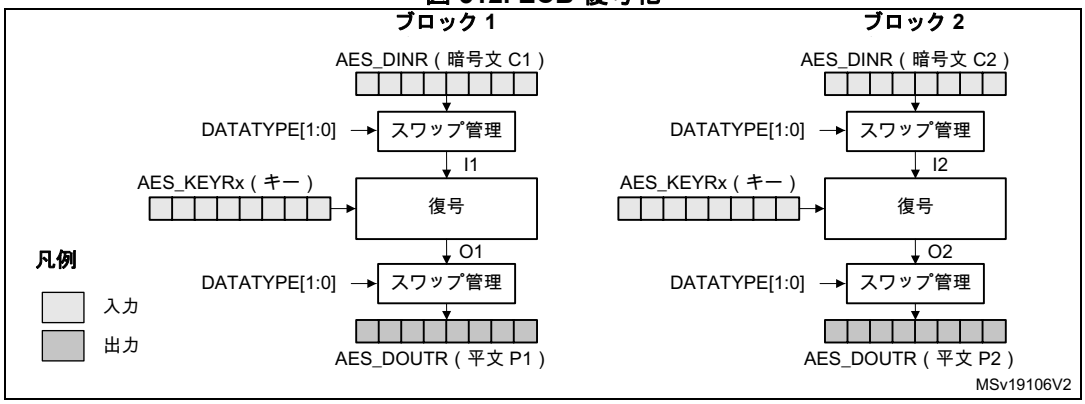
図 511. ECB 暗号化



ECB 暗号化モードでは、まず AES_DINR レジスタの 128 ビットの平文入力データブロック Px のビット／バイト／ハーフワードのスワッピングが行われます。スワップの結果 Ix が、128 ビットまたは 256 ビットキーを使って暗号化モードでセットされた AES コアで処理されます。暗号化の結果 Ox のビット／バイト／ハーフワードのスワッピングが行われ、128 ビットの暗号文出力データブロック Cx として AES_DOUTR レジスタに格納されます。ECB 暗号化は、最後の完全な平文ブロックが暗号化されるまで同様に続きます。

図 512 に、電子コードブック（ECB）復号化の図を示します。

図 512. ECB 復号化

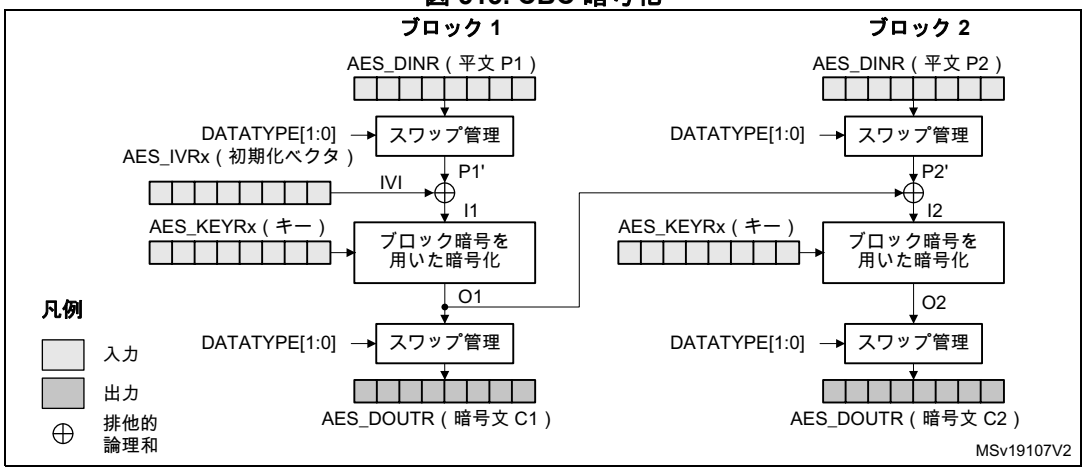


ECB モードで AES 復号化を実施するには、最終ラウンドの暗号化キーを回収し、それを暗号文を復号化するための最初のラウンドのキーとして使用して、秘密鍵を準備する必要があります（まず暗号化の完全なキースケジュールを実行する必要があります）。この準備は、AES コアによってサポートされます。

ECB 復号化モードでは、まず AES_DINR レジスタの 128 ビットの暗号文入力データブロック C1 のビット／バイト／ハーフワードのスワッピングが行われます。このキーイングシーケンスは、ECB 暗号化のものに対し、逆になっています。スワップの結果 I1 は、事前に準備した復号化キーを使って復号化モードでセットされた AES コアで処理されます。復号化結果のビット／バイト／ハーフワードのスワッピングが行われ、128 ビットの平文出力データブロック P1 として AES_DOUTR レジスタに格納されます。ECB 復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

図 513 に、暗号ブロック連鎖（CBC）復号の図を示します。

図 513. CBC 暗号化

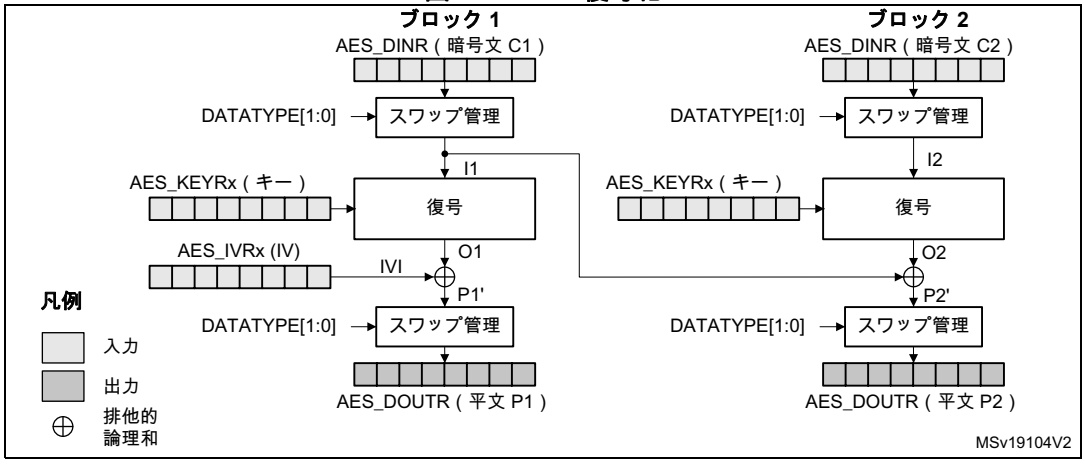


CBC 暗号化モードでは、ビット／バイト／ハーフワードのスワッピング（P1’）後、最初の平文入力ブロックと 128 ビットの IVI ビットフィールド（初期化ベクタおよびカウンタ）の排他的論理和がとられ、128 ビットまたは 256 ビットキーを使って AES コアで暗号化するための I1 入力データを生成します。スワッピング操作後、その結果の 128 ビットの出力ブロック O1 が暗号文 C1 として使用されます。その後、O1 データと 2 番目の平文データ P2’ の排他的論理和がとられ、AES コアが暗号文データの 2 番目を生成するための I2 入力データを生成します。データブロックの連鎖は、メッセージ内の最後の平文ブロックが暗号化されるまで同様に続きます。

メッセージサイズが 128 ビットの倍数ではない場合、最後の不完全なデータブロックは、セクション 34.4.6:AES 暗号文借用(CTS)およびデータ･パディングに説明されている方法で暗号化されます。

図 514 に、暗号ブロック連鎖（CBC）復号化の図を示します。

図 514. CBC 復号化



ECB 復号化モードと同様に、CBC 復号化モードで AES 復号化を実施するには、秘密鍵を準備する必要があります。

キーの準備処理後、最初の 128 ビットの暗号文ブロック（スワップ操作後）が 128 ビットまたは 256 ビットキーを使った復号化操作用の AES コア入力ブロック I1 として直接使用され、復号化が進みます。その出力 O1 と、128 ビットの IVI フィールド（暗号化で使用したものと同じである必要があります）の排他的論理和をとって、最初の平文ブロック P1 を生成します。

2 番目の暗号文ブロックは、最初のブロックの I1 データが初期化ベクタの代わりに使用されることを除けば、最初のブロックと同じ方法で処理されます。

復号化は、最後の完全な暗号文ブロックが復号化されるまで同様に続きます。

メッセージサイズが 128 ビットの倍数ではない場合、最後の不完全なデータブロックは、セクション 34.4.6：AES 暗号文借用（CTS）およびデータ・パディングに説明されている方法で復号化されます。

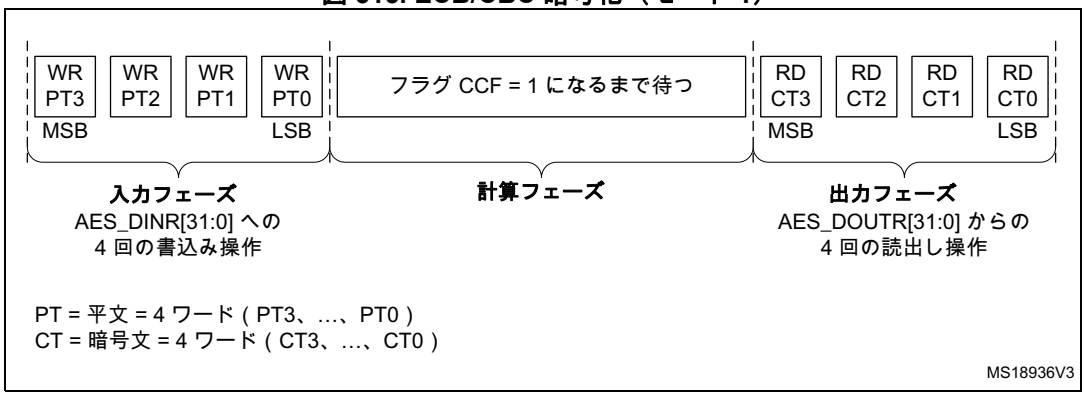
データスワッピングの詳細については、セクション 34.4.13 : AES データ・レジスタおよびデータ・スワッピングを参照してください。

ECB/CBC 暗号化シーケンス

ECB/CBC 暗号化を行うためのイベントのシーケンス（詳細はセクション 34.4.4 : を参照）:

1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR レジスタの MODE[1:0] ビットフィールドを 00 にセットしてモード 1 を選択し、AES_CR レジスタの CHMOD[2:0] ビットフィールドをそれぞれ 000 または 001 にセットして、ECB または CBC の連鎖モードを選択します。データの種類も DATATYPE[1:0] ビットフィールドで定義できます。
3. AES_CR レジスタの KEYSIZE ビットで 128 ビットまたは 256 ビットのキー長を選択します。
4. AES_KEYRx レジスタ（128 ビットまたは 256 ビット）に暗号化キーを書き込みます。CBC モードを選択した場合、AES_IVRx レジスタに初期化ベクタ・データを書き込みます。
5. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
6. AES_DINR レジスタに図 515 に示すとおり 4 回書き込んで平文（MSB ファースト）を入力します。
7. AES_SR レジスタで CCF フラグがセットされるまで待ちます。
8. AES_DOUTR レジスタを図 515 に示すとおり 4 回読み出して、暗号文（MSB ファースト）を取得します。AES_CR レジスタの CCFC ビットをセットして CCF フラグをクリアします。
9. 手順 6、7、8 を繰り返して、同じ暗号化キーですべてのブロックを処理します。

図 515. ECB/CBC 暗号化（モード 1）



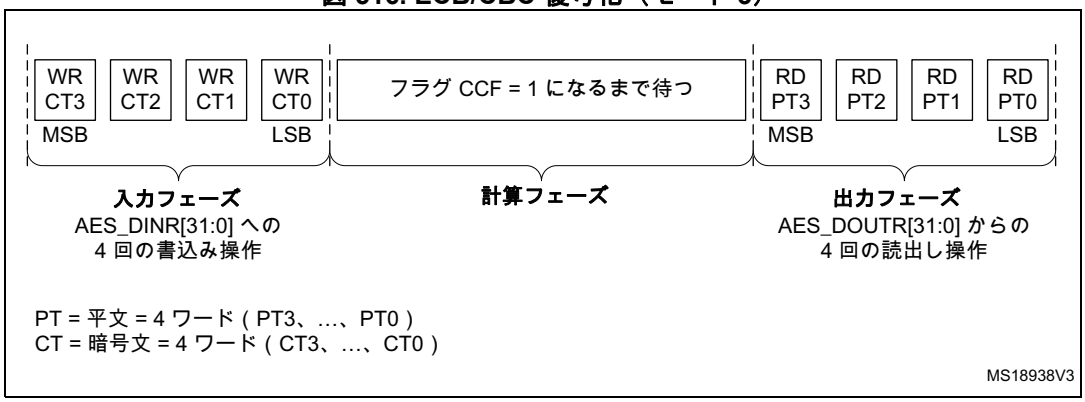
ECB/CBC 復号化シーケンス

AES ECB/CBC 復号を行うためのイベントのシーケンス（詳細はセクション 34.4.4 : を参照）は次のとおりです。

1. セクション 34.4.5 : AES 復号ラウンド・キーの準備に示された手順に従って、AES コアで復号キーを準備します。

2. AES_CR レジスタの MODE[1:0] ビットフィールドを 10 にセットしてモード 3 を選択し、AES_CR レジスタの CHMOD[2:0] ビットフィールドをそれぞれ 000 または 001 にセットして、ECB または CBC の連鎖モードを選択します。データ型も DATATYPE[1:0] ビットフィールドで定義できます。KEYSIZE ビットフィールドでは、現在の値を保持する必要があります。
3. AES_IVRx レジスタに初期化ベクタを書き込みます（CBC モードのみ必要）。
4. AES_CR レジスタの EN ビットをセットして、AES を有効にします。
5. AES_DINR レジスタに図 516 に示すとおり 4 回書き込んで暗号文（MSB ファースト）を入力します。
6. AES_SR レジスタで CCF フラグがセットされるまで待ちます。
7. AES_DOUTR レジスタを図 516 に示すとおり 4 回読み出して、平文（MSB ファースト）を取得します。AES_CR レジスタの CCFC ビットをセットして CCF フラグをクリアします。
8. 手順 5、6、7 を繰り返して、同じキーで暗号化されたすべてのブロックを処理します。

図 516. ECB/CBC 復号化（モード 3）



ECB/CBC モードでのサスペンド／レジューム動作

メッセージの処理をサスペンドするには、次の手順に従います。

1. DMA が使用されている場合には、AES_CR レジスタの DMAINEN ビットをクリアして、IN FIFO への AES DMA 転送を停止します。
2. DMA が使用されていない場合は、AES_DOUTR レジスタを 4 回読み出して、最後に処理されたブロックを保存します。DMA が使用されている場合には、AES_SR レジスタで CCF フラグがセットされるまで待ち、AES_CR レジスタの DMAOUTEN ビットをクリアして、OUT FIFO からの DMA 転送を停止します。
3. DMA が使用されていない場合は、AES_SR レジスタの CCF フラグが 1（計算完了）になるまでポーリングします。
4. AES_CR レジスタの CCFC ビットをセットして CCF フラグをクリアします。
5. 初期化ベクタ･レジスタを保存します（データ処理中に AES_IVRx レジスタが変更される CBC モードでのみ必要です）。
6. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
7. AES_CR レジスタを保存し、キー･レジスタが必要でなければ、より優先順位の高いメッセージを処理するためにクリアします。
8. DMA が使用されている場合には、DMA コントローラのステータス（IN および OUT データ転送のポインタ、残りバイト数など）を保存します。

注： **割込みが発生したプロセスが復号化の場合、ポイント 7 では、AES_KEYRx に保存された派生キーの情報を任意でメモリに保存できます。アプリケーションが元のキー値を知っているので、それらのレジスタを保存する必要は特にありません。**

メッセージの処理をレジュームするには、次の手順に従います。

1. DMA が使用されている場合には、DMA コントローラを設定して残りの FIFO IN と FIFO OUT 転送を完了させます。
2. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
3. AES_CR レジスタを（正しい KEYSIZEで）復元したうえで、AES_KEYRx レジスタを復元します。復号では、元のキー値ではなく導出されたキーの情報を AES_KEYRx レジスタに書き込めます。
4. セクション 34.4.5 : AES 復号ラウンド･キーの準備に示したように復号化キーを準備します（ECB または CBC 復号化のみ必要）。導出されたキーの情報が AES_KEYRx レジスタにロードされていれば、この手順は必要ありません。
5. 保存した設定を使用して AES_IVRx レジスタを復元します（CBC モードでのみ必要）。
6. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
7. DMA が使用されている場合には、AES_CR レジスタの DMAINEN ビットと DMAOUTEN ビットをセットして、AES DMA 転送を有効にします。

モード 4 を使った代替シングル ECB／CBC 復号化

モード 4 を使った ECB／CBC 復号化のシングルラウンドを実行するイベントのシーケンスは以下の通り：

1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR レジスタの MODE[1:0] ビットフィールドを 11 に設定してモード 4 を選択し、AES_CR レジスタの CHMOD[2:0] ビットフィールドを 0x0 または 0x1 に設定して、それぞれ ECB または CBC の連鎖モードを選択します。
3. AES_CR レジスタの KEYSIZE ビットフィールドで 128 ビットまたは 256 ビットのキー長を選択します。
4. AES_KEYRx レジスタに暗号化キーを書き込みます。CBC モードを選択した場合は、AES_IVRx レジスタに書き込みます。
5. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
6. AES_DINR レジスタへの 4 回の書込みにより、MSB を先頭にして順番に暗号文を入力します）。
7. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
8. AES_DOUTR レジスタを 4 回読み出して、平文を取得します（MSB から順に）。AES_CR レジスタの CCFC ビットをセットして CCF フラグをクリアします。

注： **モード 4 が選択されている場合、モード 3 は使用できません。**

モード 4 では、AES_KEYRx レジスタは処理のすべてのフェーズで暗号化キーを含みます。派生キーは、これらのレジスタには保存されません。AES に内部的に保存されます。

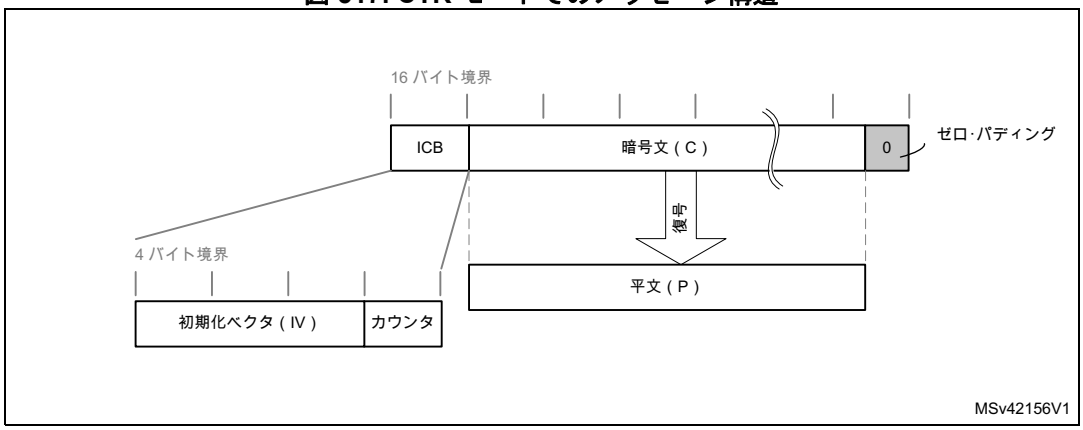
34.4.9 AES カウンタ（CTR）モード

概要

カウンタモード（CTR）では、AES をキーストリームジェネレータとして使用します。生成されたキーは、その後平文との排他的論理和をとって暗号化されます。

CTR 連鎖は、NIST 特別公報 800-38A『ブロック暗号の推奨動作モード』の中に定義されています。CTR モードでの典型的なメッセージ構造を図 517 に示します。

図 517. CTR モードでのメッセージ構造



このメッセージの構造は次のとおりです。

- 16 バイトの初期カウンタブロック（ICB）は、次の 2 つの異なるフィールドで構成されています。
  - **初期化ベクタ**（IV）: ある特定のキーでの暗号化サイクルごとに一意でなければならない 96 ビットの値です。
  - **カウンタ**：ブロック処理が完了するたびにインクリメントされる 32 ビットのビッグエンディアンの整数です。カウンタの初期値は 1 に設定する必要があります。
- 平文 P は、既知の長さの暗号文 C として暗号化されます。この長さは 16 バイトの倍数でなくとも構いませんが、その場合には平文のパディングが必要です。

CTR の暗号化と復号化

図 518 および図 519 では、AES ペリフェラルで実行される CTR の暗号化および復号化の処理についてそれぞれ示しています。CTR モードは、AES_CR レジスタの CHMOD[2:0] ビットフィールドに 010 を書き込んで選択します。

図 518. CTR 暗号化

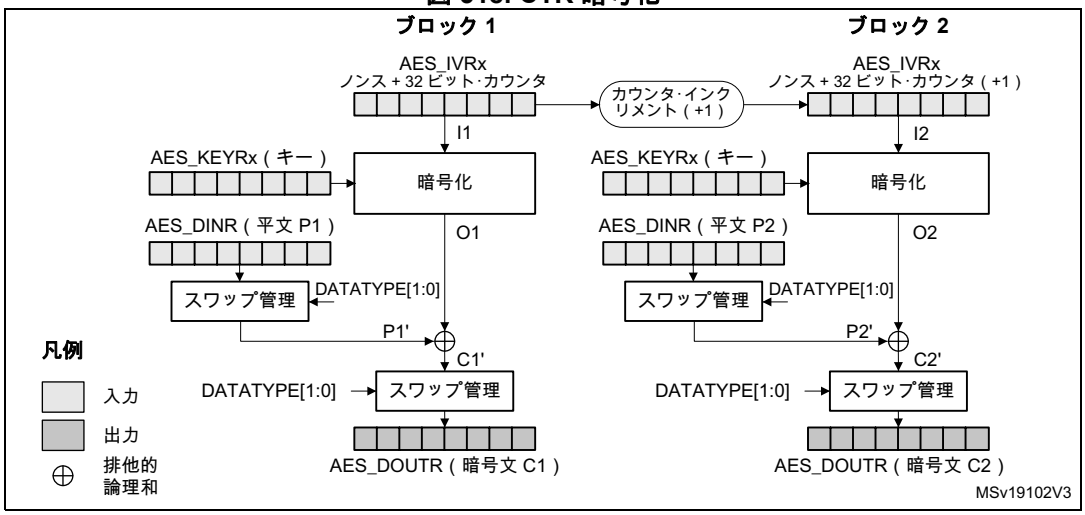
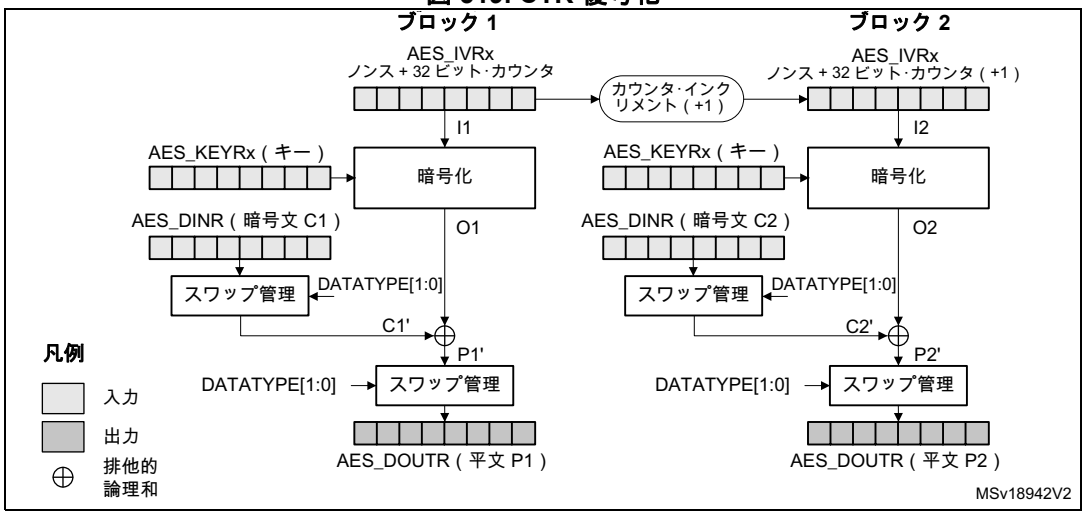


図 519. CTR 復号化



CTR モードでは、暗号コア出力（キーストリームとも呼ばれます）Ox と関連入力ブロック（暗号化では Px'、復号化では Cx'）の排他的論理和がとられ、正しい出力ブロック（暗号化では Cx'、復号化では Px'）を生成します。AES の初期化ベクタは、表 320 に示すように初期化する必要があります。

表 320. CTR モード初期化ベクタの定義

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
IVI[127:96]	IVI[95:64]	IVI[63:32]	IVI[31:0] 32 ビット･カウンタ = 0x0001

最初のデータ･ブロックを処理するときに一度だけ AES_IVRx レジスタを使用する CBC モードとは違って、CTR モードでは、各データ･ブロックを処理するたびに AES_IVRx レジスタが使用され、AES ペリフェラルによって初期化ベクタのカウンタ･ビットがインクリメントされます（ノンス･ビットは変更されません）。

コアが常に現在のカウンタブロックを暗号化して、平文（CTR 暗号化）や暗号文（CTR 復号化）の入力と排他的論理和がとられるキーストリームを生成するので、CTR 復号化は CTR 暗号化と変わりません。CTR モードでは、MODE[1:0] ビットフィールドを 01（キー導出）に設定することが禁止されており、他のすべての設定はデフォルトで暗号化モードになります。

CTR 連鎖モードで暗号化または復号を行うためのイベントのシーケンス：

1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR レジスタの CHMOD[2:0] ビットフィールドを 010 に設定して、CTR 連鎖モードを選択します。MODE[1:0] ビットフィールドを 01 以外の値にセットします。
3. AES_KEYRx レジスタを初期化して、AES_IVRx レジスタを表 320 に示されたとおりロードします。
4. AES_CR レジスタの EN ビットをセットして、現在のカウンタの暗号化を開始します（EN は計算が終了すると自動的にリセットされます）。
5. 最終ブロックの場合には、必要に応じてデータを 0 でパディングして完全なブロックとします。
6. AES でデータを追加して結果を読み出します。可能性のある 3 通りのケースはセクション 34.4.4: 暗号操作を行うための AES の手順に記載されています。
7. 最後から 2 番目のブロックが処理されるまで前の手順を繰り返します。最終ブロックでは、前の 2 つの手順を適用して、ペイロードの一部ではないビットを破棄します（最終入力ブロックの有効なデータのサイズが 16 バイト未満の場合）。

CTR モードでのサスペンド／レジューム動作

CBC モードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。CBC サスペンド／レジュームシーケンスの詳細については、セクション 34.4.8：AES 基本連鎖モード（ECB、CBC）を参照してください。

注： **CBC モードと同様に、レジューム動作中に AES_IVRx レジスタを再ロードする必要があります。**

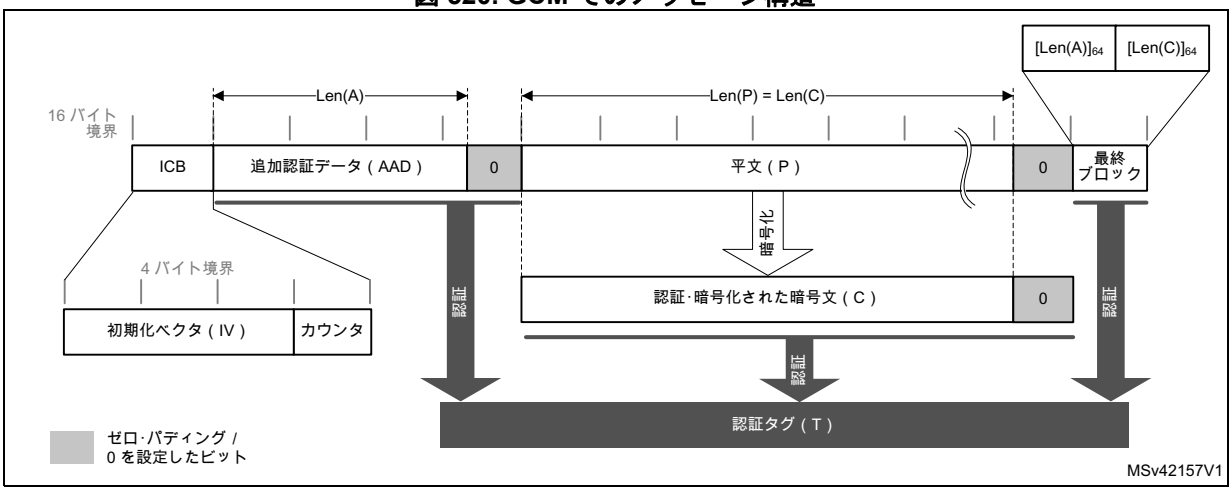
34.4.10 AES ガロア／カウンタモード（GCM）

概要

AES ガロア／カウンタモード（GCM）では、平文メッセージを対応する暗号文およびタグ（メッセージ認証コードとも言います）に暗号化して認証できます。機密性の確保のため、GCM アルゴリズムは AES カウンタモードに基づいています。固定の有限フィールドを超えて乗算器を使用しタグを生成します。

GCM 連鎖は、NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』の中に定義されています。GCM モードでの典型的なメッセージ構造を図 520 に示します。

図 520. GCM でのメッセージ構造



メッセージには、次の構造があります。

- **16 バイトの初期カウンタブロック（ICB）**は、次の 2 つの異なるフィールドで構成されています。
  - **初期化ベクタ（IV）**: ある特定のキーでの暗号化サイクルごとに一意でなければならない 96 ビットの値です。GCM 規格では 96 ビット未満の IV に対応していますが、この場合には規則が厳格に適用されることに注意してください。
  - **カウンタ**：ブロック処理が完了するたびにインクリメントされる 32 ビットのビッグエンディアンの整数です。NIST 仕様によると、ペイロードの最初のブロックを処理するときのカウンタ値は 0x2 です。
- **認証済みヘッダ AAD**（追加認証データとも言います）は、既知の長さである Len(A) を持ちます。この長さは 16 バイトの倍数でなくても構いませんが、$2^{64}1$ ビットを超えてはなりません。メッセージのこの部分には認証だけ行われて、暗号化はされません。
- **平文メッセージ P** には暗号文 C として認証と暗号化の両方が行われ、既知の長さである Len(P) を持ちます。この長さは 16 バイトの倍数でなくても構いませんが、128 ビットの 2^{32}-2 ブロック分を超えることはできません。
- **最終ブロック**には、表 321 に示すとおり、AAD ヘッダ長（ビット [32:63]）とペイロード長（ビット [96:127]）の情報が含まれます。

GCM 規格では、暗号文 C は平文 P と同じビット長であることと規定されています。

メッセージ（AAD または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。

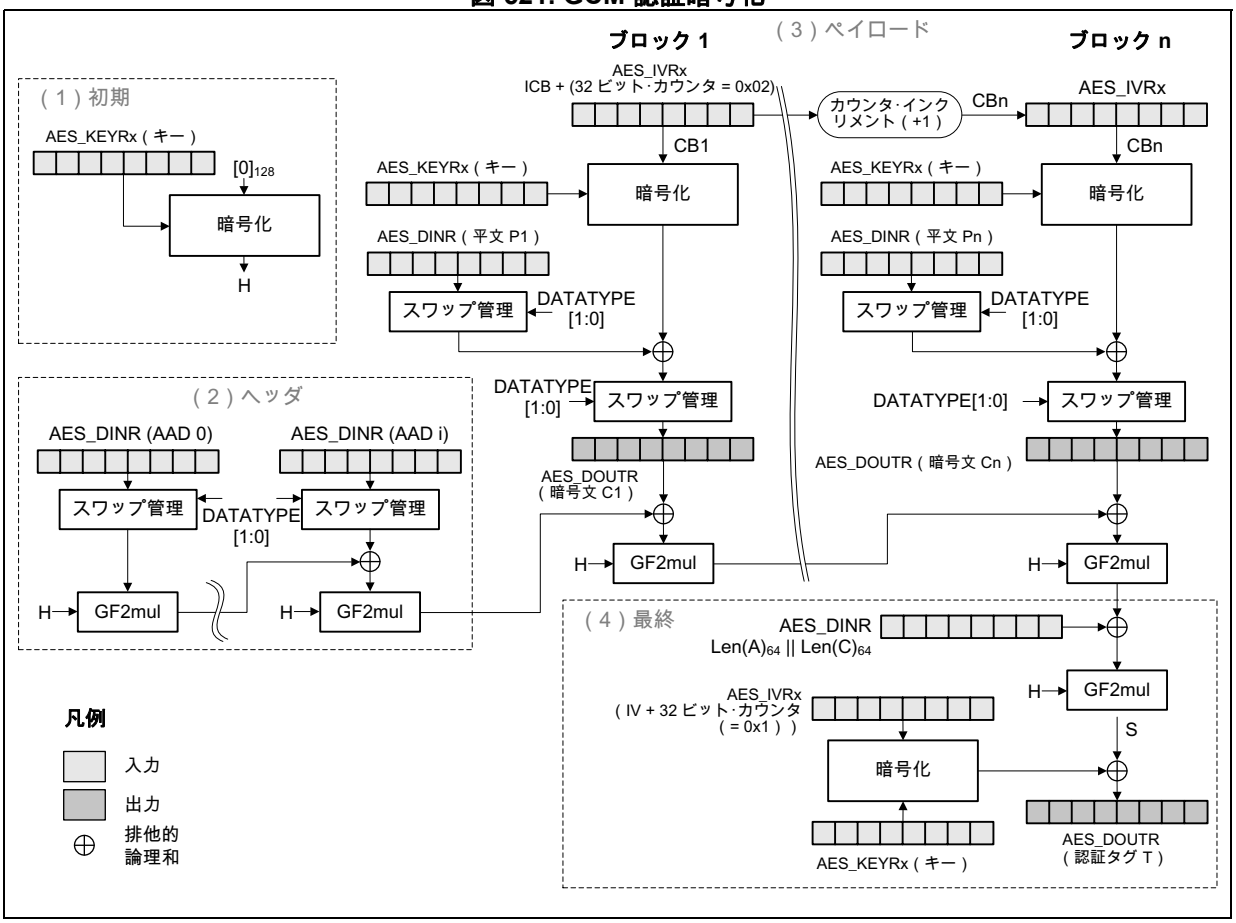
表 321. GCM 最終ブロックの定義

エンディアン	ビット [0] --- ビット [31]	ビット [32] --- ビット [63]	ビット [64] ---ビット [95]	ビット [96] --- ビット [127]
入力データ	0x0	AAD 長 [31:0]	0x0	ペイロード長 [31:0]

GCM 処理

図 521 には、AES ペリフェラルでの GCM の実装について示しています。GCM は、AES_CR レジスタの CHMOD[2:0] ビットフィールドに 011 を書き込んで選択します。

図 521. GCM 認証暗号化



GCM モードにおける平文の機密性のメカニズムはカウンタモードと類似しており、入力カウンタブロックのシーケンスを生成する特別なインクリメント機能（32 ビットインクリメントと示される）を備えています。

各データブロックの処理には AES_IVRx レジスタ（データの**カウンタブロック**を保持）が使用されます。AES ペリフェラルが、カウンタ [31:0] ビットフィールドを自動的にインクリメントします。最初のカウンタブロック（CB1）は、アプリケーションソフトウェアによって初期カウンタブロック ICB から導出されます（表 322 参照）。

表 322. GCM モードでの AES_IVRx レジスタの初期化

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
ICB[127:96]	ICB[95:64]	ICB[63:32]	ICB[31:0] 32 ビット･カウンタ = 0x0002

注 : このモードでは、MODE[1:0] ビットフィールドに 01 と 11 を設定することが禁止されています。

GCM モードにおける認証メカニズムは **GF2mul** と呼ばれるハッシュ機能に基づいており、このハッシュ機能では、バイナリ･ガロア体の中で、ハッシュ･サブキー（H）と呼ばれる固定パラメータによる乗算が行われます。

GCM メッセージは、次の項目に詳述されている以下のフェーズで処理されます。

- **初期フェーズ** : AES が GCM ハッシュサブキー（H）を準備します。
- **ヘッダフェーズ** : AES が追加認証データ（AAD）を処理して、ハッシュの計算のみを行います。
- **ペイロードフェーズ** : AES が平文（P）を処理して、ハッシュの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- **最終フェーズ** : AES がメッセージの最終ブロックを使って認証タグ（T）を生成します。

GCM 初期フェーズ

この最初のステップでは、GCM ハッシュサブキー（H）が計算され、すべてのブロックの処理に使用されるように内部に保存されます。推奨シーケンスは、次のとおりです。

1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR レジスタの CHMOD[2:0] ビットフィールドを 011 に設定して、GCM 連鎖モードを選択し、必要に応じて DATATYPE[1:0] ビットフィールドを設定します。
3. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 00 にセットして、初期フェーズを示します。
4. AES_CR レジスタの MODE[1:0] ビットフィールドを 00 または 10 にセットします。このビットフィールドはペイロードフェーズでのみ使用されますが、初期フェーズで設定し、後続のすべてのフェーズで変更しないことをお勧めします。
5. キーで AES_KEYRx レジスタを初期化して、AES_IVRx レジスタを表 322 で定義された情報で初期化します。
6. AES_CR レジスタの EN ビットを 1 にセットしてハッシュキーの計算を開始します（EN は計算が終了すると自動的にリセットされます）。
7. AES_SR の CCF フラグが 1 に変わって計算が終了するまで待ちます。あるいは、対応する割込みを使用します。
8. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。

GCM ヘッダ･フェーズ

このフェーズは GCM 初期フェーズの後に発生し、ペイロードフェーズの前に完了させる必要があります。実行するシーケンスは、暗号化および復号化と同じで次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 01 にセットして、ヘッダフェーズを示します。初期フェーズで設定した MODE[1:0] ビットフィールドは変更しないでください。
2. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、ブロックの AAD サイズが 128 ビットを下回る場合、ブロックの残り部分を 0 でパディングします。次に、AES に セクション 34.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加します。このフェーズでは、データは一切読み出されません。
4. 最後の追加認証データ･ブロックが処理されるまで手順 3 を繰り返します。

注： **ヘッダフェーズは、AAD がなければ（Len(A) = 0 であれば）スキップできます。**

GCM ペイロードフェーズ

このフェーズは､暗号化および復号化と同じで GCM ヘッダフェーズの後に実行されます｡このフェーズでは、暗号化／復号化したペイロードが AES_DOUTR レジスタに格納されます。実行するシーケンスは、次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 10 にセットして、ペイロードフェーズを示します。初期フェーズで設定した MODE[1:0] ビットフィールドは変更しないでください。
2. ヘッダフェーズをスキップした場合、AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、平文（暗号化）または暗号文（復号化）のサイズが 128 ビットを下回る場合、ブロックの残り部分を 0 でパディングします。
4. AES に 1466 ページのセクション 34.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加して、結果を読み出します。
5. 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号化されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を実行します。最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、ペイロードの一部ではないビットを破棄します。

注 : **ペイロードフェーズは、ペイロードデータがなければ (Len(C) = 0 であれば) スキップできます (GMAC モード参照)。**

GCM 最終フェーズ

この最終フェーズでは､AES ペリフェラルが GCM 認証タグを生成して AES_DOUTR に格納します。実行するシーケンスは、次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 11 に設定して、最終フェーズを指定します。
2. 表 321 に示すとおり､AAD とペイロードのビット長を連結してブロックのデータを構成します。AES_DINR レジスタにブロック書き込みます。
3. AES_SR の CCF フラグが 1 に変わって計算が終了するまで待ちます。
4. AES_DOUTR レジスタを 4 回読み出して GCM 認証タグを取得します。
5. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。
6. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。認証された復号化である場合には、生成されたタグをメッセージとともに渡された予測タグと比較します。

注 : **最終フェーズでは、データが AES_DINR に通常の方法で書き込まれます (スワッピングなし)。一方、AES_DOUTR から読み出されるタグ･データにはスワッピングが適用されます。**

ヘッダ･フェーズまたはペイロード･フェーズから最終フェーズに遷移する際に、AES ペリフェラルを無効にしないでください。そうしないと、誤った結果になります。

GCM モードでのサスペンド／レジューム動作

メッセージの処理をサスペンドするには、次の手順に従います。

1. DMA が使用されている場合には､AES_CR レジスタの DMAINEN ビットをクリアして､IN FIFO への AES DMA 転送を停止します。DMA が使用されていない場合は､AES_SR レジスタの CCF フラグが 1 にセットされ、現在の計算が完了したことを確認します。
2. ペイロードフェーズで､DMA が使用されていない場合は､AES_DOUTR レジスタを 4 回読み出して、最後に処理されたブロックを保存します。DMA が使用されている場合には、AES_SR レジスタで CCF フラグがセットされるまで待ち、AES_CR レジスタの DMAOUTEN ビットをクリアして、OUT FIFO からの DMA 転送を停止します。
3. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。
4. メモリに AES_SUSPxR レジスタを保存します（ここで、x は 0 から 7 です）。
5. ペイロードフェーズでは､AES_IVRx レジスタがデータ処理中に初期値から変化するため､それらを保存します。ヘッダフェーズでは、この手順は不要です。
6. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
7. メモリに現在の AES 設定を保存します（初期化ベクタレジスタ AES_IVRx 以外）。アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。
8. DMA が使用されている場合には､DMA コントローラのステータス（ IN データ転送のポインタ、残りバイト数など）を保存します。ペイロードフェーズでは、OUT データ転送のポインタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMA が使用されている場合には、DMA コントローラを設定して残りの FIFO IN 転送を完了させます。ペイロードフェーズでは､残りの FIFO OUT 転送も DMA コントローラで設定する必要があります。
2. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
3. メモリに事前に保存したサスペンド･レジスタの値を、対応する AES_SUSPxR レジスタに書き戻します（ここで、x は 0 から 7 です）。
4. ペイロードフェーズでは、メモリに事前に保存した初期化ベクタレジスタの値を、対応する AES_IVRx レジスタに書き戻します。ヘッダ･フェーズでは、初期設定値を AES_IVRx レジスタに書き戻します。
5. AES_CR と AES_KEYRx のレジスタに初期設定値を復元します。
6. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。

DMA が使用されている場合には、AES_CR レジスタの DMAINEN ビット（ペイロードフェーズでは DMAOUTEN ビットも）をセットして、AES DMA リクエストを有効にします。

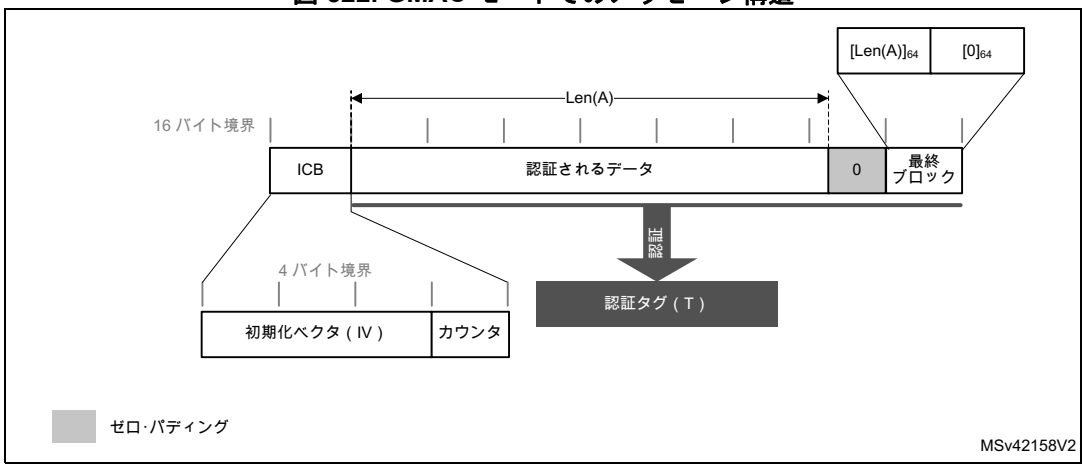
34.4.11 AES ガロア･メッセージ認証コード（GMAC）

概要

ガロアメッセージ認証コード（GMAC）では、平文を認証して、対応するタグ情報（メッセージ認証コードとも言います）を生成することができます。NIST 特別公報 800-38D『ブロック暗号の推奨動作モード - ガロア／カウンタモード（GCM）および GMAC』に定義されている GCM アルゴリズムに基づいています。

GMAC での典型的なメッセージ構造を図 522 に示します。

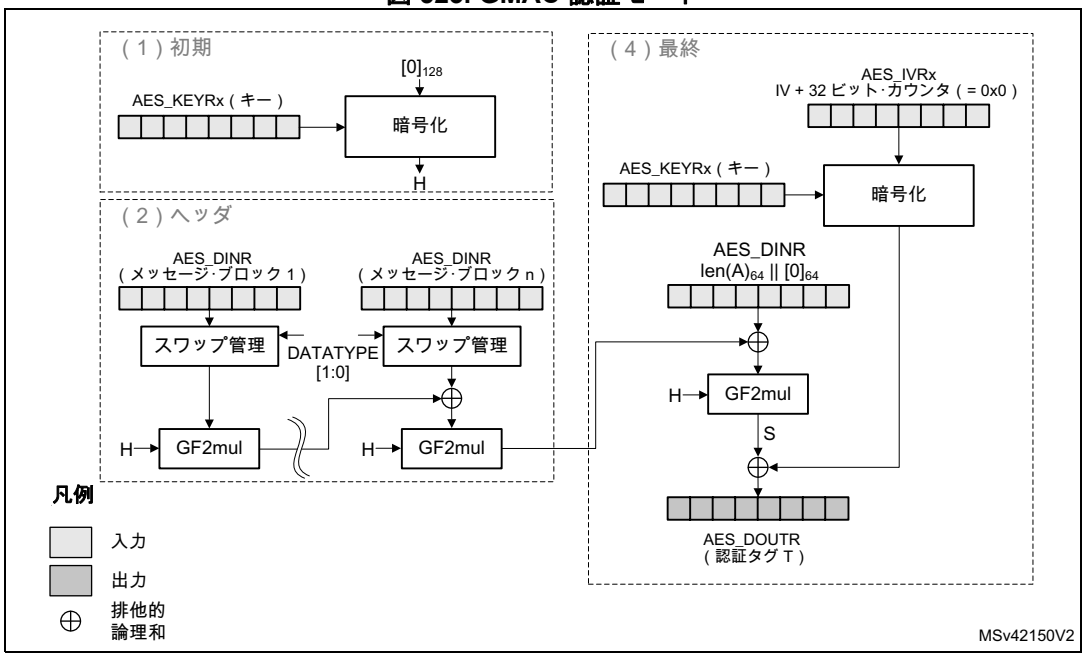
図 522. GMAC モードでのメッセージ構造



AES GMAC 処理

図 523 には、AES ペリフェラルでの GMAC モードの実装について示しています。このモードは、AES_CR レジスタの CHMOD[2:0] ビットフィールドに 011 を書き込んで選択します。

図 523. GMAC 認証モード



GMAC アルゴリズムはヘッダのみ含まれているメッセージに適用される GCM アルゴリズムに相当します。その結果、ペイロードフェーズが省略されること以外の手順と設定がすべて GCM と同じです。

GMAC でのサスペンド／レジューム動作

GMAC モードでは、ヘッダフェーズのみが割込み可能であることを除いて、GCM で示されたシーケンスが適用されます。

34.4.12 CBC-MAC 付き AESカウンタ（CCM）

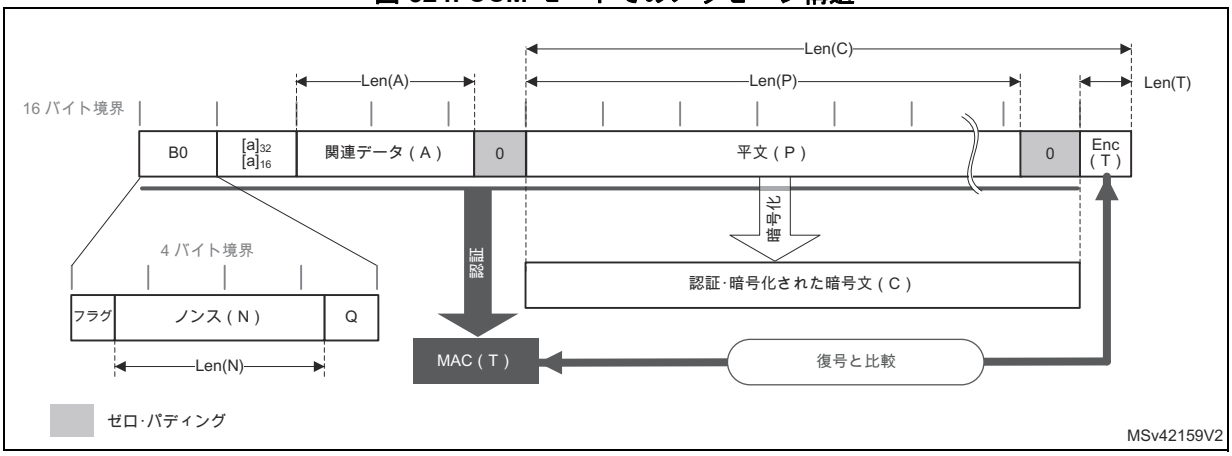
概要

AES **暗号ブロック連鎖-メッセージ認証コード付きカウンタ**（CCM）アルゴリズムでは、平文を暗号化して認証し、対応する暗号文およびタグ（メッセージ認証コードとも言います）を生成することができます。機密性の確保のため、CCM アルゴリズムはカウンタモードでの AES に基づいています。暗号ブロック連鎖技術を使ってメッセージ認証コードを生成します。これは、一般的に CBC-MAC と呼ばれます。

注： **NIST は、CCM 仕様の関係以外では CBC-MAC を認証モードとして承認していません。**

CCM 連鎖は、NIST 特別公報 800-38C『ブロック暗号の推奨動作モード - 認証および機密性のための CCM モード』の中に規定されています。CCM での典型的なメッセージ構造を図 524 に示します。

図 524. CCM モードでのメッセージ構造



メッセージの構造は次のとおりです。

- **16 バイトの初期認証ブロック（B0）**は、次の 3 つの異なるフィールドで構成されています。
  - **Q**：P のオクテット長（Len(P)）を表すビットストリングです。
  - **ノンス（N）**：Len(N) サイズの使い捨て値です（新しい通信のたびに新しいノンスを割り当てる必要があります）。Len(N) + Len(P) の合計は、15 バイトにする必要があります。
  - **フラグ**：規格に規定されている、制御情報のための 4 つのフラグを格納する最上位オクテットです。値 **t**（バイト単位での MAC の長さ）と **Q**（Len(P) < 2^{8q} バイトとなる平文の長さ）をエンコードするための 3 ビットのストリングが 2 つ格納されています。**Q** に関連付けられるカウンタ･ブロックの範囲は 2^{8Q-4} となり、**Q** の最大値が 8 である場合、暗号で使用されるカウンタ･ブロックは 60 ビットとなります。

- **16 バイト･ブロック（B）**は関連データ（A）に関連付けられます。
  メッセージのこの部分には認証だけ行われて、暗号化はされません。この部分は既知の長さである Len(A) を持っており、この長さは 16 バイトの倍数でなくとも構いません（図 524 参照）。規格には、最初のメッセージブロック（B1）の上位側ビットにおいて、バイト数で表現された関連データ長（a）は次の定義のようにエンコードされなければならないとも示されています。
  - $0 < a < 2^{16} - 2^8$ である場合には、$[a]_{16}$ のようにエンコードされ、2 バイトとなります。
  - $2^{16} - 2^8 < a < 2^{32}$ である場合には、0xff || 0xfe || $[a]_{32}$ のようにエンコードされ、6 バイトとなります。
  - $2^{32} < a < 2^{64}$ である場合には、0xff || 0xff || $[a]_{64}$ のようにエンコードされ、10 バイトとなります。
- **16 バイトブロック（B）**は既知の長さ Len(P) を持ち、暗号文 C として認証と暗号化の両方が行われている平文メッセージ P に関連付けられます。この長さは 16 バイトの倍数でなくとも構いません（図 524 参照）。
- 長さ Len(T) の**暗号化された MAC (T)** が、全長が Len(C) である暗号文 C に追加されます。

メッセージ（A または P）の長さが 16 バイトの倍数ではない場合には、特別なパディング処理が必要となります。

注： **CCM 連鎖モードは、関連データのみ（ペイロードなし）にも使用できます。**

例として、NIST 特別公報 800-38C のセクション C.1 には次の値（16 進数）が示されています。

N：10111213 141516（Len(N) = 56 ビットまたは 7 バイト）

A：00010203 04050607（Len(A) = 64 ビットまたは 8 バイト）

P：20212223（Len(P) = 32 ビットまたは 4 バイト）

T：6084341B（Len(T) = 32 ビットまたは t = 4）

B0：4F101112 13141516 00000000 00000004
B1：00080001 02030405 06070000 00000000
B2：20212223 00000000 00000000 00000000

CTR0：0710111213 141516 00000000 00000000

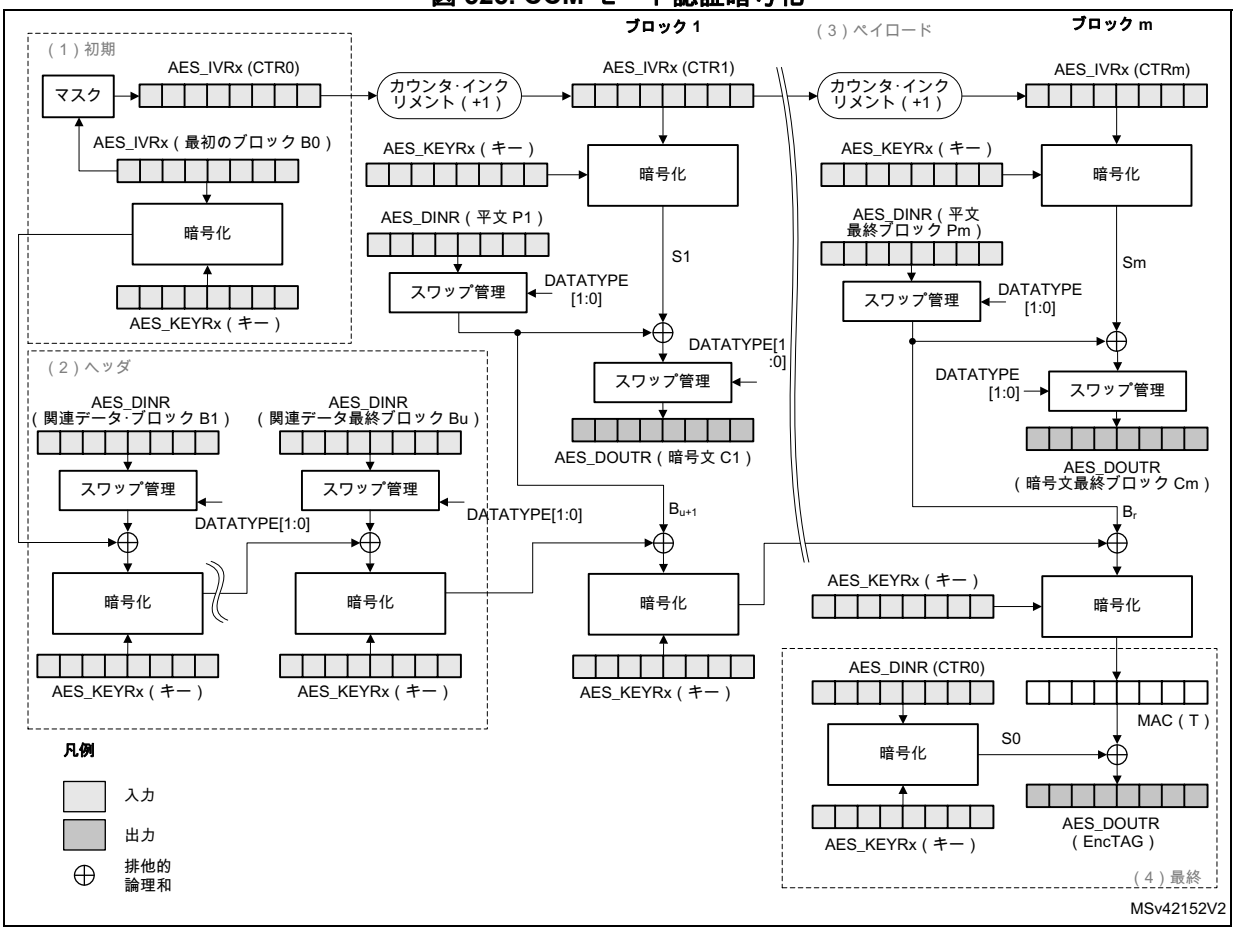
CTR1：0710111213 141516 00000000 00000001

フォーマットされた入力データ･ブロック Bx（特に B0 と B1）の生成は、アプリケーションで管理する必要があります。

CCM 処理

図 525 には、AES ペリフェラルでの CCM の実装（暗号化の例）について示しています。このモードは、AES_CR レジスタの CHMOD[2:0] ビットフィールドに 100 を書き込んで選択します。

図 525. CCM モード認証暗号化



生成-暗号化処理に入力するデータは、有効なナンス、有効なペイロードストリング、有効な関連データストリングであり、すべて適切にフォーマットされているものです。CBC 連鎖メカニズムはフォーマットされた平文データに適用されて、既知の長さを持つ MAC を生成します。入力として十分な長さのカウンタブロックのシーケンスが必要であるカウンタモードの暗号化は、ペイロードストリングと MAC に個別に適用されます。結果の暗号文 C は、平文 P に対する生成-暗号化処理の出力です。

AES_IVRx レジスタは各データブロックを処理するために使用され、AES によって、最初のブロック B0 により定義されるビット長を持つ CTR カウンタが自動的にインクリメントされます。表 323 には、アプリケーションが B0 データをどのように読み込むかについて示しています。

注： **CCM モードでの AES ペリフェラルは､NIST が指定する最大 64 ビットのカウンタをサポートしています。**

表 323. CCM モードでの AES_IVRx レジスタの初期化

AES_IVR3[31:0]	AES_IVR2[31:0]	AES_IVR1[31:0]	AES_IVR0[31:0]
B0 [127:96]	B0 [95:64]	B0 [63:32]	B0 [31:0]

注： **このモードでは、MODE[1:0] ビットフィールドに 01 と 11 を設定することが禁止されています。**

CCM メッセージは、次のサブセクションで詳述する以下のフェーズに沿って処理されます。

- **初期フェーズ**：AES が最初のブロックを処理して最初のカウンタブロックを準備します。
- **ヘッダフェーズ**：AES が関連データ（A）を処理して、タグの計算のみを行います。
- **ペイロードフェーズ**：IP が平文（P）を処理して、タグの計算、カウンタブロックの暗号化、データの排他的論理和を行います。暗号文（C）にも同様な操作を行います。
- **最終フェーズ**：AES がメッセージ認証コード（MAC）を生成します。

CCM 初期フェーズ

このフェーズでは、CCM メッセージの最初のブロック B0 が AES_IVRx レジスタに書き込まれます。AES_DOUTR レジスタに出力データは一切格納されません。推奨シーケンスは、次のとおりです。

1. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
2. AES_CR レジスタの CHMOD[2:0] ビットフィールドを 100 に設定して、CCM 連鎖モードを選択し、必要に応じて DATATYPE[1:0] ビットフィールドを設定します。
3. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 00 にセットして、初期フェーズを示します。
4. AES_CR レジスタの MODE[1:0] ビットフィールドを 00 または 10 にセットします。このビットフィールドはペイロードフェーズでのみ使用されますが、初期フェーズで設定し、後続のすべてのフェーズで変更しないことをお勧めします。
5. キーで AES_KEYRx レジスタを初期化して、AES_IVRx レジスタを表 323 に示された B0 データで初期化します。
6. AES_CR レジスタの EN ビットを 1 にセットしてカウンタの計算を開始します（EN は計算が終了すると自動的にリセットされます）。
7. AES_SR の CCF フラグが 1 に変わって計算が終了するまで待ちます。あるいは、対応する割込みを使用します。
8. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。

CCM ヘッダ･フェーズ

このフェーズは GCM 初期フェーズの後に発生し、ペイロードフェーズの前に完了させる必要があります。このフェーズでは、AES_DOUTR レジスタには出力データは一切格納されません。

実行するシーケンスは、暗号化および復号化と同じで次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 01 にセットして、ヘッダフェーズを示します。初期フェーズで設定した MODE[1:0] ビットフィールドは変更しないでください。
2. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
3. これが最終ブロックであり、ブロックの AAD サイズが 128 ビットを下回る場合、ブロックの残り部分を 0 でパディングします。次に、AES に セクション 34.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加します。このフェーズでは、データは一切読み出されません。
4. 最後の追加認証データ･ブロックが処理されるまで手順 3 を繰り返します。

注： **ヘッダフェーズは、関連データがなければ（Len(A) = 0 であれば）スキップできます。**

関連データ（B1）の最初のブロックは、ソフトウェアで関連データ長でフォーマットする必要があります。

CCM ペイロードフェーズ（暗号化または復号化）

このフェーズは、暗号化および復号化と同じで CCM ヘッダフェーズの後に実行されます。このフェーズでは、暗号化／復号化したペイロードが AES_DOUTR レジスタに格納されます。実行するシーケンスは、次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 10 にセットして、ペイロードフェーズを示します。初期フェーズで設定した MODE[1:0] ビットフィールドは変更しないでください。
2. ヘッダフェーズをスキップした場合、AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
3. これが暗号化する最終データブロックであり、ブロックの平文のサイズが 128 ビットを下回る場合、ブロックの残り部分を 0 でパディングします。
4. AES に 1466 ページのセクション 34.4.4 : 暗号操作を行うための AES の手順で示されたいずれかの方法でデータブロックを追加して、結果を読み出します。
5. 最後から 2 番目の平文ブロックが暗号化されるか、暗号文の最終ブロックが復号化されるまで前の手順を繰り返します。平文の最終ブロック（暗号化のみ）に対しては、前の 2 つの手順を適用します。最終ブロックでは、最終ブロックのサイズが 16 バイト未満である場合に、ペイロードの一部ではないデータを破棄します。

注： **ペイロード・フェーズは、ペイロード・データがなければ（Len(P) = 0 または Len(C) = Len(T) であること）スキップできます。**

暗号文 C を復号するときには、暗号化されたタグ情報である $LSB_{Len(T)}(C)$ を削除します。

CCM 最終フェーズ

この最終フェーズでは、AES ペリフェラルが GCM 認証タグを生成して AES_DOUTR に格納します。実行するシーケンスは、次のとおりです。

1. AES_CR レジスタの GCMPH[1:0] ビットフィールドを 11 にセットして、最終フェーズを示します。
2. AES_SR レジスタの計算終了フラグ CCF がセットされるまで待ちます。
3. AES_DOUTR レジスタを 4 回読み出します。この出力が CCM 認証タグに相当します。
4. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。
5. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
6. 認証された復号化である場合には、生成された暗号化タグを、暗号文でパディングされた暗号化タグと比較します。

注： **この最終フェーズでは、AES_DOUTR レジスタから読み出されるタグ・データにスワップが適用されます。**

ヘッダ・フェーズから最終フェーズに遷移する際に、AES ペリフェラルを無効にしないでください。そうしないと、誤った結果になります。

アプリケーションは、有効なタグを取得するために、タグの長さで認証タグ出力をマスクする必要があります。

CCM モードでのサスペンド／レジューム動作

ヘッダフェーズおよびペイロードフェーズで**メッセージの処理をサスペンドするには**、次の手順に従います。

1. DMA が使用されている場合には、AES_CR レジスタの DMAINEN ビットをクリアして、IN FIFO への AES DMA 転送を停止します。DMA が使用されていない場合は、AES_SR レジスタの CCF フラグが 1 にセットされ、現在の計算が完了したことを確認します。
2. ペイロードフェーズで、DMA が使用されていない場合は、AES_DOUTR レジスタを 4 回読み出して、最後に処理されたブロックを保存します。DMA が使用されている場合には、AES_SR レジスタで CCF フラグがセットされるまで待ち、AES_CR レジスタの DMAOUTEN ビットをクリアして、OUT FIFO からの DMA 転送を停止します。
3. AES_CR レジスタの CCFC ビットをセットして、AES_SR レジスタの CCF フラグをクリアします。
4. メモリに AES_SUSPxR レジスタを保存します（ここで、x は 0 から 7 です）。
5. AES_IVRx レジスタがデータ処理中に初期値から変化するため、それらを保存します。
6. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
7. メモリに現在の AES 設定を保存します（初期化ベクタレジスタ AES_IVRx 以外）。アプリケーションが元のキー値を知っていますので、キーレジスタを保存する必要はありません。
8. DMA が使用されている場合には、DMA コントローラのステータス（ IN データ転送のポインタ、残りバイト数など）を保存します。ペイロードフェーズでは、OUT データ転送のポインタも保存する必要があります。

メッセージの処理をレジュームするには、次の手順に従います。

1. DMA が使用されている場合には、DMA コントローラを設定して残りの FIFO IN 転送を完了させます。ペイロードフェーズでは、残りの FIFO OUT 転送も DMA コントローラで設定する必要があります。
2. AES_CR レジスタの EN ビットをクリアして、AES ペリフェラルを無効にします。
3. メモリに事前に保存したサスペンド･レジスタの値を、対応する AES_SUSPxR レジスタに書き戻します（ここで、x は 0 から 7 です）。
4. メモリに事前に保存した初期化ベクタレジスタの値を、対応する AES_IVRx レジスタに書き戻します。
5. AES_CR と AES_KEYRx のレジスタに初期設定値を復元します。
6. AES_CR レジスタの EN ビットをセットして、AES ペリフェラルを有効にします。
7. DMA が使用されている場合には、AES_CR レジスタの DMAINEN ビット（ペイロードフェーズでは DMAOUTEN ビットも）を 1 にセットして、AES DMA リクエストを有効にします。

34.4.13 AES データ･レジスタおよびデータ･スワッピング

データの入出力

128 ビットのデータ･ブロックは、AES_DINR レジスタ（ビットフィールド DIN[31:0]）に、4 つの 32 ビット･ワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）の順序で連続して書き込むことで AES ペリフェラルに入力されます。

128 ビットのデータ･ブロックは、AES_DOUTR レジスタ（ビットフィールド DOUT[31:0]）から、4 つの 32 ビット･ワードを最上位ワード（ビット [127:96]）から最下位ワード（ビット [31:0]）の順序で連続して読み出すことで、AES ペリフェラルから取得されます。

AES_DINR レジスタまたは AES_DOUTR レジスタの 32 ビットのデータ･ワードは、次のいずれかのビッグ･エンディアン順で整理されます。

- AES_DINR に書き込むワードの最上位バイトは、書き込むワードを保持する 4 つの隣接したメモリ位置のうち、一番小さいアドレスに配置する必要があります。
- AES_DOUTR から読み出すワードの最上位バイトは、ワードを受け取る 4 つの隣接したメモリ位置のうち、一番小さいアドレスになります。

AES への入力データブロック書込みに DMA を使用する場合は、入力ブロックの 4 ワードをビックエンディアン順（一番小さいアドレスに最上位ワード）で連続してメモリに格納する必要があります。セクション 34.4.16：AES DMA インタフェースを参照してください。

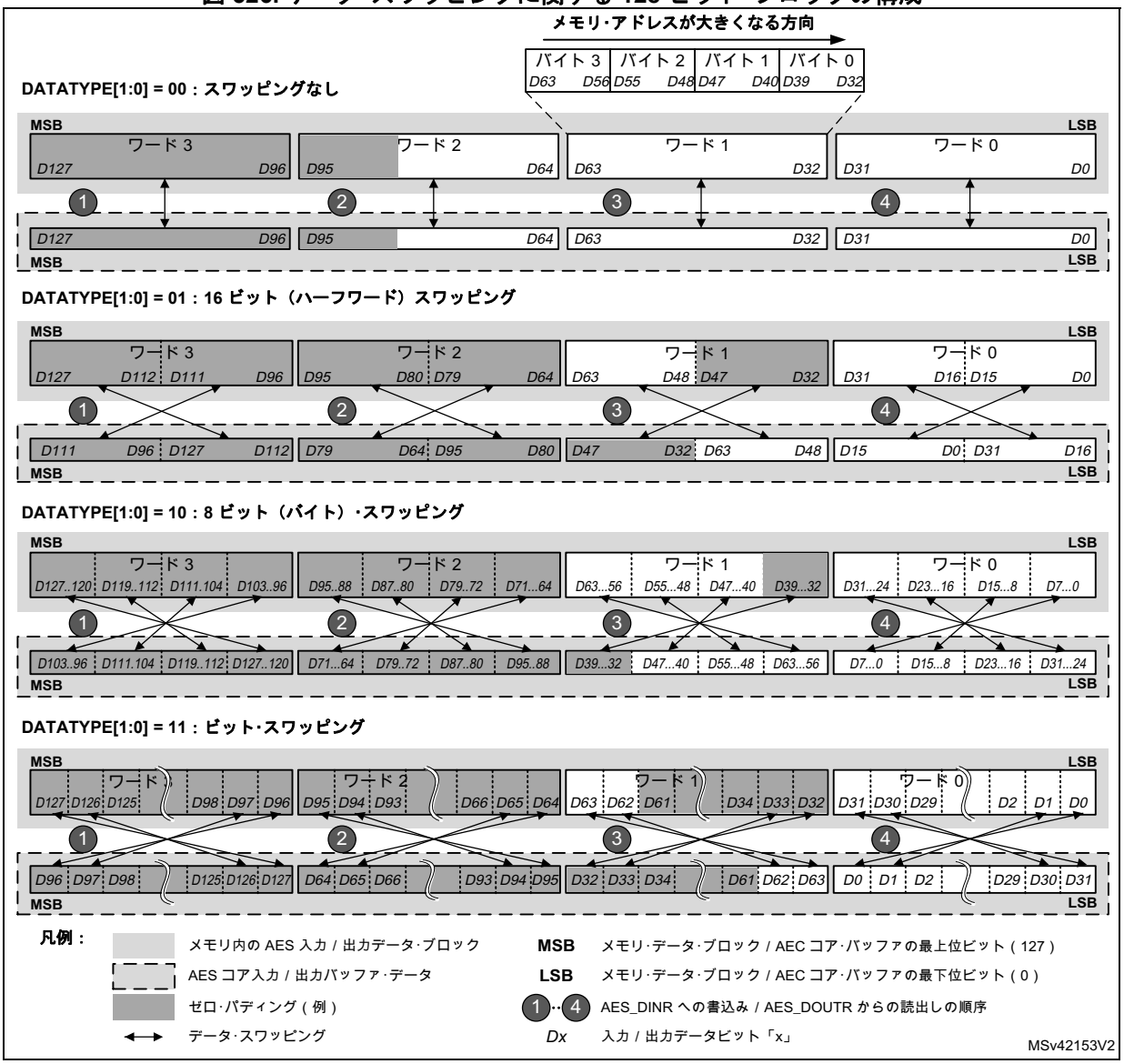
データスワッピング

AES ペリフェラルは、AES 処理コアに読み込まれる前の AES_DINR レジスタの入力データワード、および AES_DOUTR レジスタに送られる前の AES 処理コアから出力されたデータについて、ビット、バイト、ハーフワードのスワッピングまたはスワッピングなしに設定できます。データのタイプによって選択します。たとえば、ASCII テキストストリームの場合、バイトスワッピングを使用します。

データスワッピングのタイプは、AES_CR レジスタの DATATYPE[1:0] ビットフィールドで選択します。この選択は、AES コアの入力と出力の両方に適用されます。

それぞれのデータ･スワッピングのタイプについては、図 526 に AES_DINR レジスタから入力されたデータの AES 処理コアの入力バッファ･データ P127 ~ P0 の構成、または AES_DOUTR レジスタから得られる出力データの AES 処理コアの出力バッファ･データ P127 ~ P0 の構成を示しています。

図 526. データ･スワッピングに関する 128 ビット･ブロックの構成



注： **AES キーレジスタ（AES_KEYRx）と初期化レジスタ（AES_IVRx）のデータは、選択されたスワップモードの影響を受けません。**

データパディング

図 526 では、データスワッピング後の 0 にされたビットによって AES コア入力バッファの最上位ビット側に隣接したゾーンが形成される場合など、0 でパディングされるメモリデータブロックの例も示しています。この例では、次を含む入力データブロックのパディングが示されています。

- 48 メッセージビット（DATATYPE[1:0] = 01）
- 56 メッセージビット（DATATYPE[1:0] = 10）
- 34 メッセージビット（DATATYPE[1:0] = 11）

34.4.14 AES キーレジスタ

AES_KEYRx レジスタは、暗号化または復号化キービットフィールドの KEY[127:0] や KEY[255:0] を格納します。各レジスタに書き込むデータまたは各レジスタから読み出すデータは、最上位バイトが最上位アドレスにセットされるリトルエンディアンの順に配列されます。

表 324 に示すように、キーは 8 個のレジスタに跨ります。

表 324. AES_KEYRx レジスタでのキーのエンディアン設定（128 ビットまたは 256 ビットのキー長）

AES_KEYR7 [31:0]	AES_KEYR6 [31:0]	AES_KEYR5 [31:0]	AES_KEYR4 [31:0]	AES_KEYR3 [31:0]	AES_KEYR2 [31:0]	AES_KEYR1 [31:0]	AES_KEYR0 [31:0]
-	-	-	-	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]
KEY[255:224]	KEY[223:192]	KEY[191:160]	KEY[159:128]	KEY[127:96]	KEY[95:64]	KEY[63:32]	KEY[31:0]

暗号化または復号のキーは、AES_CR レジスタの EN ビットをクリアして AES ペリフェラルを無効にした状態で、これらのレジスタに書き込むことができます。

キー･レジスタは、AES_CR レジスタの DATATYPE[1:0] ビットフィールドで制御されるデータ･スワッピングによる影響を受けません。

34.4.15 AES 初期化ベクタ･レジスタ

4 つの AES_IVRx レジスタで初期化ベクタの入力ビットフィールド IVI[127:0] を保持します。各レジスタで読み書きするデータは、リトルエンディアン順で最上位バイトをメモリ内の一番大きいアドレスに配置するように整理されます。レジスタも一番小さいアドレス（AES_IVR0）から一番大きいアドレス（AES_IVR3）の順になります。

ビットフィールドのデータの位は、選択された連鎖モードに基づきます。使用時は、ビットフィールドが AES コアの計算サイクルごとに更新されます。

AES ペリフェラルが有効であるときの AES_IVRx レジスタへの書込み操作は、レジスタの内容に影響を与えません。AES_IVRx レジスタの内容を変更する場合はまず AES_CR レジスタの EN ビットをクリアする必要があります。

AES_IVRx レジスタを読み出すと、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

AES_IVRx レジスタは、AES_CR レジスタの DATATYPE[1:0] ビットフィールドで制御されるデータ･スワッピング機能による影響を受けません。

34.4.16 AES DMA インタフェース

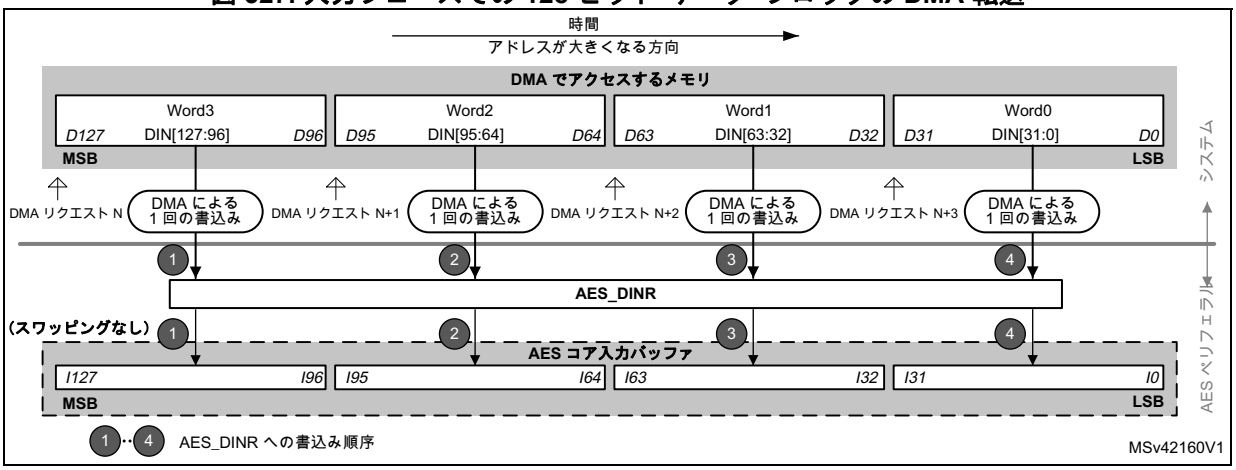
AES ペリフェラルには、DMA（ダイレクトメモリアクセス）コントローラに接続するインタフェースが搭載されています。DMA 操作は AES_CR レジスタで制御します。

DMA を用いたデータ入力

AES_CR レジスタの DMAINEN ビットをセットすると、AES への DMA 書込みが有効になります。図 527 に示すように、AES ペリフェラルは、入力フェーズ中に AES_DINR レジスタに 128 ビットのブロック（4 倍ワード）を書き込む必要があるたびに、DMA リクエストを開始します。

注： **選択されたアルゴリズムとモードに応じて、特別なパディングや暗号文借用（CTS）が必要となることがあります。たとえば、AES GCM 暗号化や AES CCM 復号の場合は、DMA 転送に最終ブロックを含めないでください。詳細については、セクション 34.4.4：暗号操作を行うための AES の手順を参照してください。**

図 527. 入力フェーズでの 128 ビット・データ・ブロックの DMA 転送

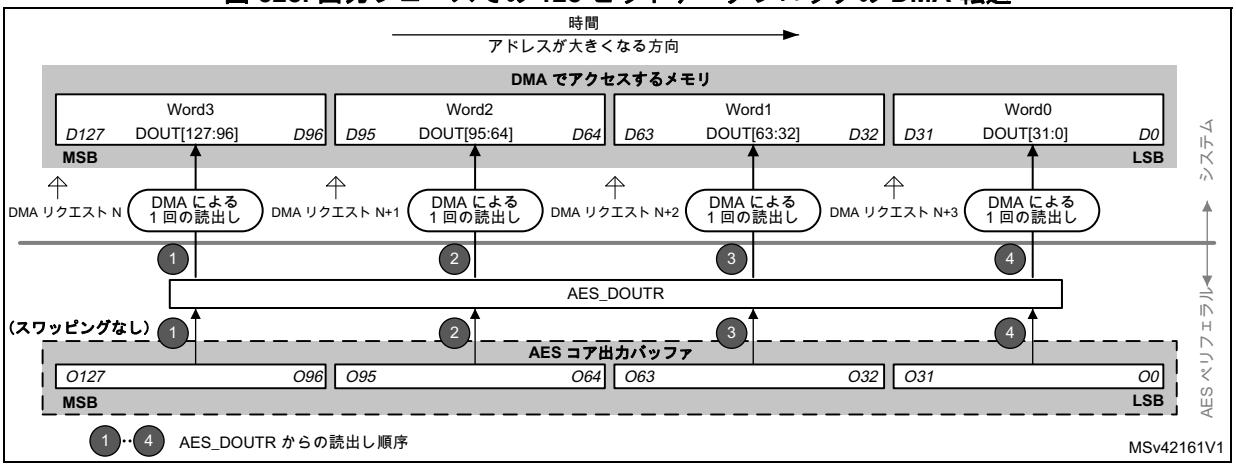


DMA を用いたデータ出力

AES_CR レジスタの DMAOUTEN ビットをセットすると、AES からの DMA 読出しが有効になります。図 528 に示すように、AES ペリフェラルは、出力フェーズ中に AES_DINR レジスタから 128 ビットのブロック（4 倍ワード）を読み出す必要があるたびに、DMA リクエストを開始します。

注： **メッセージ・サイズによっては、最後のブロックでアプリケーションによって余分なバイトを破棄する必要が生じることがあります。**

図 528. 出力フェーズでの 128 ビットデータブロックの DMA 転送



さまざまな動作モードにおける DMA 操作

DMA 操作は、モード 1（暗号化）またはモード 3（復号）が AES_CR レジスタの MODE[1:0] ビットフィールドで選択されている場合に使用できます。モード 2（キー導出）では、AES_KEYRx レジスタはソフトウェアで書き込む必要があり、AES_CR レジスタの DMAINEN および DMAOUTEN ビットで DMA 転送を有効にしても、このモードでは影響がありません。

DMA シングルリクエストは、無効にされるまで AES によって生成されます。そのため、128 ビットデータブロック処理終了時のデータ出力フェーズの後に、AES によって次のデータブロックの新しいデータ入力フェーズ（ある場合）に自動的に切り替えられます。

AES とメモリ間のデータ転送を DMA で処理する場合、CCF フラグは不要です。AES_DOUTR レジスタの読出しが、計算フェーズの最後に DMA によって自動的に処理されるからです。CCF フラグのクリアが必要になるのは、ソフトウェア管理のデータ転送に戻るときだけです。詳細については、セクション 34.4.4：暗号操作を行うための AES の手順のサブセクション データ追加を参照してください。

34.4.17 AES エラー管理

AES の設定は、AES_CR レジスタの EN ビットをクリアすることで、いつでも変更できます。

読出しエラー・フラグ（RDERR）

AES_DOUTR レジスタに対して予期せぬ読出しが試みられるとAES_SR レジスタの RDERR フラグがセットされ、ゼロが返されます。

RDERR は、計算フェーズまたは入力フェーズでトリガされます。

注： **AES は、RDERR エラー検出で無効化されずに処理を続けます。**

AES_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 34.5：AES 割込みを参照してください。

RDERR フラグをクリアするには、AES_CR レジスタの ERRIE ビットをセットします。

書込みエラー・フラグ（WDERR）

AES_DINR レジスタに対して予期せぬ書込みが試みられると AES_SR レジスタの WRERR フラグがセットされ、AES_DINR レジスタには何の影響も現れません。WRERR は、計算フェーズまたは出力フェーズでトリガされます。

注： **AES は、WRERR エラー検出で無効化されずに処理を続けます。**

AES_CR レジスタの ERRIE ビットがセットされている場合、割込みが生成されます。詳細については、セクション 34.5：AES 割込みを参照してください。

WRERR フラグをクリアするには、AES_CR レジスタの ERRC ビットをセットします。

34.5 AES 割込み

AES ペリフェラルによって生成され、個別にマスク可能な割込みソースは、次のイベントを通知します。

- 計算完了
- 読出しエラー
- 書込みエラー

これらの個別ソースを組み合わせて共通の割込み信号 aes_it にします。これが NVIC（ネスト化されたベクタ割込みコントローラ）に接続します。各ソースは、AES_CR レジスタの対応するイネーブル・ビットをセットまたはクリアすることで個別にそれぞれ有効化または無効化され、AES_CR レジスタの対応するビットをセットすることでクリアされます。

各ソースのステータスは AES_SR レジスタから読み出すことができます。

割込みソースの概要、イベント・フラグ、イネーブル・ビットについては、表 325 を参照してください。

表 325. AES 割込みリクエスト

項目（割込みの略称）	AES 割込みイベント	イベントフラグ	イネーブルビット	割込みのクリア方法
AES	計算完了フラグ	CCF	CCFIE	CCFC のセット(1)
	読出しエラー･フラグ	RDERR	ERRIE	ERRC のセット (1)
	書込みエラー･フラグ	WRERR		

1. AES_CR レジスタのビット

34.6 AES 処理遅延時間

以下の表では、各動作モードで 128 ビット･ブロックを処理する際の遅延時間の概要を示します。

表 326. ECB、CBC および CTR での処理遅延時間

キー長	動作モード	アルゴリズム	クロック サイクル数
128 ビット	モード 1：暗号化	ECB、CBC、CTR	51
	モード 2：キー派生	-	59
	モード 3：復号化	ECB、CBC、CTR	51
	モード 4：キー導出から復号化	ECB、CBC	106
256 ビット	モード 1：暗号化	ECB、CBC、CTR	75
	モード 2：キー派生	-	82
	モード 3：復号化	ECB、CBC、CTR	75
	モード 4：キー導出から復号化	ECB、CBC	145

表 327. GCM および CCM での処理遅延時間（クロック･サイクル数）

キー長	動作モード	アルゴリズム	初期フェーズ	ヘッダ フェーズ(1)	ペイロード フェーズ (1)	タグ･フェーズ (1)
128 ビット	モード 1：暗号化／ モード 3：復号	GCM	64	35	51	59
		CCM	63	55	114	58
256 ビット	モード 1：暗号化／ モード 3：復号	GCM	88	35	75	75
		CCM	87	79	162	82

1. データ挿入時間には、AES により強制される AHB バスのウェイト･ステートが加算されることがあります（最大 3 サイクル、通常は 1 サイクル）。

34.7 AES レジスタ

34.7.1 AES 制御レジスタ （AES_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NPBLB[3:0]				Res.	KEYSIZE	Res.	CHMOD[2]
								rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	GCMPH[1:0]		DMAOUTEN	DMAINEN	ERRIE	CCFIE	ERRC	CCFC	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **NPBLB[3:0]**：最終ブロックにおけるパディング･バイト数

このビットフィールドで、ペイロードの最終ブロックにおけるパディングバイト数をセットします。

0000：全バイトが有効（パディングなし）

0001：最終ブロックの最下位バイト 1 個に対するパディング

.....

1111：最終ブロックの最下位バイト 15 個に対するパディング

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **KEYSIZE**：キー長選択

このビットフィールドは、AES 暗号コアに使用するキーの長さ（ビット単位）を定義します。

0：128

1：256

AES_CR レジスタの EN ビットがセットされていると、このビットに書き込もうとしても無視されます。したがって、その書込みアクセスによって、このビットがクリアされることはありません。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:13 **GCMPH[1:0]**：GCM または CCM フェーズの選択

このビットフィールドで GCM、GMAC、または CCM アルゴリズムを選択します。

00：初期フェーズ

01：ヘッダフェーズ

10：ペイロードフェーズ

11：最終フェーズ

GCM、GMAC、または CCM アルゴリズム以外が（ALGOMODE ビットフィールドで）選択されている場合、このビットフィールドの影響はありません。

ビット 12 **DMAOUTEN**：DMA 出力有効

このビットは、出力フェーズで DMA でのデータ転送を有効または無効にします。

0：無効化

1：イネーブル

このビットがセットされると、出力データフェーズ中に AES によって DMA リクエストが自動生成されます。

この機能は、モード 1 またはモード 3 が MODE[1:0] ビットフィールドで選択されている場合のみ有効です。

モード 2（キー導出）では有効ではありません。

モード 4（単一復号）に DMA を使用することは推奨されません。

ビット 11 **DMAINEN：**DMA 入力イネーブル

このビットは、入力フェーズで DMA でのデータ転送を有効／無効にします。

0：無効化

1：イネーブル

このビットがセットされると、入力データフェーズ中に AES によって DMA リクエストが自動生成されます。

この機能は、モード 1 またはモード 3 が MODE[1:0] ビットフィールドで選択されている場合のみ有効です。

モード 2（キー導出）では有効ではありません。

モード 4（単一復号）に DMA を使用することは推奨されません。

ビット 10 **ERRIE**：エラー割込み有効化

このビットは、RDERR または WRERR がセットされている場合に AES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：イネーブル

ビット 9 **CCFIE**：CCF 割込み有効

このビットは、CCF（計算完了フラグ）がセットされたときの AES 割込み生成を有効化または無効化（マスク）します。

0：無効化（マスク）

1：イネーブル

ビット 8 **ERRC**：エラー･フラグ･クリア

このビットで 1 を書き込むと、AES_SR レジスタの RDERR および WRERR エラーフラグがクリアされます。

0：影響なし。

1：RDERR および WRERR フラグをクリアします。

フラグを読み出すと常に 0 が返されます。

ビット 7 **CCFC：**計算完了フラグ･クリア

このビットで 1 を書き込むと、AES_SR レジスタの計算完了フラグ（CCF）がクリアされます。

0：影響なし。

1：CCF のクリア

フラグを読み出すと必ず 0 が返されます。

ビット 16、6:5 **CHMOD[2:0]**：連鎖モードの選択

このビットフィールドで、AES 連鎖モードを選択します。

000：電子コードブック（ECB）

001：暗号ブロック連鎖（CBC）

010：カウンタ･モード（CTR）

011：ガロア･カウンタ･モード（GCM）およびガロア･メッセージ認証コード（GMAC）

100：CBC-MAC 付きカウンタ（CCM）

上記以外：予約済みです。

AES_CR レジスタの EN ビットがセットされていると、このビットフィールドに書き込もうとしても無視されます。したがって、その書込みアクセスによって、このビットフィールドがクリアされることはありません。

ビット 4:3 **MODE[1:0]**：AES 動作モード

このビットフィールドで、AES 動作モードを選択します。

00：モード 1：暗号化

01：モード 2：キー導出（ECB/CBC 復号のキーの準備）

10：モード 3：復号

11：モード 4：キー導出後に単一復号

AES_CR レジスタの EN ビットがセットされていると、このビットフィールドに書き込もうとしても無視されます。したがって、その書込みアクセスによって、このビットフィールドがクリアされることはありません。ECB または CBC 連鎖モードのどちらも選択されていない場合にモード 4 を選択しようとすると、デフォルトで有効な選択であるモード 3 になります。モード 4 からモード 3 を選択することはできません。

ビット 2:1 **DATATYPE[1:0]**：データ型選択

このビットフィールドは、データスワッピングのモードを選択して、AES_DINR レジスタに書き込まれるデータまたは AES_DOUTR レジスタから読み出されるデータのフォーマットを定義します。

00：なし

01：ハーフワード（16 ビット）

10：バイト（8 ビット）

11：ビット

詳細については、セクション 34.4.13 : AES データ・レジスタおよびデータ・スワッピングを参照してください。

AES_CR レジスタの EN ビットがセットされていると、このビットフィールドに書き込もうとしても無視されます。したがって、その書込みアクセスによって、このビットフィールドがクリアされることはありません。

ビット 0 **EN**：AES イネーブル

このビットは、AES ペリフェラルを有効または無効にします。

0：無効化

1：イネーブル

このビットをクリアしてからセットすれば、いつでも AES ペリフェラルを再初期化できます。

このビットは、キーの準備（モード 2）の完了時と GCM/GMAC/CCM 初期化フェーズの完了時に、ハードウェアによって自動的にクリアされます。

34.7.2 AES ステータス・レジスタ（AES_SR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
												r	r	r	r

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **BUSY**：ビジー

このフラグは、AES が GCM ペイロード**暗号化**フェーズ中にアイドルかビジーかを示します。

0：アイドル

1：ビジー

このフラグが「アイドル」を示している場合は、より優先順位の高いメッセージを処理するために現在の GCM 暗号化処理をサスペンドできます。他の連鎖モードまたはペイロード暗号化以外の GCM フェーズでは、サスペンドされたプロセスに対するフラグを無視する必要があります。

ビット 2 **WRERR**：書込みエラー

このフラグは、AES_DINR レジスタへの予期しない書込み操作（計算中またはデータ出力フェーズ中）の検出を示します。

0：検出なし

1：検出

フラグはハードウェアによってセットされます。AES_CR レジスタの ERRC ビットをセットすると、ソフトウェアによってクリアされます。

AES_CR レジスタの ERRIE ビットで割込みが有効になっている場合は、このフラグがセットされると割込みが生成されます。

このフラグのセットは、AES 動作に影響しません。予期せぬ書込みは無視されます。

ビット 1 **RDERR**：読出しエラー･フラグ

このフラグは、AES_DOUTR レジスタからの予期しない読出し操作（計算中またはデータ入力フェーズ中）の検出を示します。

0：検出なし

1：検出

フラグはハードウェアによってセットされます。AES_CR レジスタの ERRC ビットをセットすると、ソフトウェアによってクリアされます。

AES_CR レジスタの ERRIE ビットで割込みが有効になっている場合は、このフラグがセットされると割込みが生成されます。

このフラグのセットは、AES 動作に影響しません。予期せぬ読出しにはゼロが返されます。

ビット 0 **CCF**：計算完了フラグ

このフラグは計算が完了したかどうかを示します。

0：未完了

1：完了

フラグは、計算完了時にハードウェアによってセットされます。AES_CR レジスタの CCFC ビットをセットすると、このフラグがソフトウェアによってクリアされます。

AES_CR レジスタの CCFIE ビットで割込みが有効になっている場合､このフラグがセットされると割込みが生成されます。

フラグは DMAOUTEN ビットが 0 の場合のみ有効です。DMA_EN が 1 の場合、ハイのままとなります。

34.7.3 AES データ入力レジスタ（AES_DINR）

アドレスオフセット：0x08

リセット値：0x0000 0000

32 ビット･アクセス･タイプのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DIN[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DIN[31:0]**：入力データ･ワード

入力フェーズでこのビットフィールドに 4 回連続で書き込むことによって、AES ペリフェラルに 128 ビット･ブロックの入力データがすべて書き込まれます。1 回目から 4 回目の書込みにかけて、対応するデータの重みは [127:96]、[95:64]、[63:32]、[31:0] になります。書込みごとに、32 ビット入力バッファのデータが DATATYPE[1:0] ビットフィールドに応じたデータ･スワッピング･ブロックによって処理されたうえで、AES コアの 128 ビット入力バッファに書き込まれます。

入力データ･ブロックのデータの意味は、AES の動作モードによって異なります。

- **モード 1**（暗号化）：平文
- **モード 2**（キー導出）：このビットフィールドは使用しません（AES_KEYRx レジスタを入力に使用）
- **モード 3**（復号）および **モード 4**（キー導出に続いて単一復号）：暗号文

データスワッピング操作の手順は、1491 ページのセクション 34.4.13 : AES データ･レジスタおよびデータ･スワッピングを参照してください。

34.7.4 AES データ出力レジスタ（AES_DOUTR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

32 ビット読出しアクセス･タイプのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DOUT[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOUT[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **DOUT[31:0]**：出力データ･ワード

この読出し専用のビットフィールドは、32 ビット出力バッファをフェッチします。計算完了（CCF セット）時にこのビットフィールドを 4 回連続で読み出すことによって、実質的には AES ペリフェラルから 128 ビット･ブロックの出力データがすべて読み出されます。AES コアによって生成されたデータは、出力バッファに達する前に DATATYPE[1:0] ビットフィールドに応じたデータ･スワッピング･ブロックによって処理されます。

1 回目から 4 回目までの読出し操作でのデータの重みは [127:96]、[95:64]、[63:32]、[31:0] になります。

出力データ･ブロックのデータの意味は、AES の動作モードによって異なります。

- **モード 1**（暗号化）：暗号文
- **モード 2**（キー派生）：ビットフィールド不使用（出力は AES_KEYRx レジスタを使用）
- **モード 3**（復号化）および**モード 4**（キー派生に続いて単一キー復号化）：平文

データスワッピング操作の手順は、1491 ページのセクション 34.4.13 : AES データ･レジスタおよびデータ･スワッピングを参照してください。

34.7.5 AES キー･レジスタ 0（AES_KEYR0）

アドレス･オフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[31:0]**：暗号化キー、ビット [31:0]

このビットフィールドには、動作モードに応じて AES 暗号化または復号キーのビット [31:0] が含まれます。

- **モード 1**（暗号化）、**モード 2**（キー導出）、および**モード 4**（キー導出後の単一復号）では、このビットフィールドに書き込む値は暗号化キーです。
- **モード 3**（復号化）：ビットフィールドに書き込む値は、復号化に使用する前に導出される暗号化キーです。ビットフィールドに暗号化キーを書き込んだ後、AES をイネーブルする前にそれを読み出すと同じ値が返されます。AES をイネーブルした後かつCCF フラグをセットした後にそれを読み出すと、暗号化キーから派生した暗号化キーが返されます。

注： モード 4（キー派生に続いて単一キー復号化）では、ビットフィールドには常に暗号化キーが含まれます。

AES_KEYRx レジスタに書き込めるのは、KEYSIZE の値が正しく、AES ペリフェラルが無効化（AES_CR レジスタの EN ビットがクリア）されているときのみです。キーが AES_KEYRx レジスタに直接読み込まれる場合があることに注意してください（その場合はキー･レジスタへの書込みが無視され、KEIF がセットされます）。詳細については、1493 ページのセクション 34.4.14：AES キーレジスタを参照してください。

34.7.6 AES キー･レジスタ 1（AES_KEYR1）

アドレス･オフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[63:32]**：暗号化キー、ビット [63:32]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.7 AES キー・レジスタ 2（AES_KEYR2）

アドレス・オフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[95:80]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[79:64]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[95:64]**：暗号化キー、ビット [95:64]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.8 AES キー・レジスタ 3（AES_KEYR3）

アドレス・オフセット：0x1C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[127:112]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[111:96]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[127:96]**：暗号化キー、ビット [127:96]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.9 AES 初期化ベクタ・レジスタ 0（AES_IVR0）

アドレス・オフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVI[31:0]**：初期化ベクタ入力、ビット [31:0]

IVI[127:0] フィールドの説明については、1493 ページのセクション 34.4.15 : AES 初期化ベクタ・レジスタを参照してください。

初期化ベクタは、ECB 以外の連鎖モードでのみ使用します。

AES_IVRx レジスタは、AES ペリフェラルが無効である場合にのみ書き込むことができます。

34.7.10 AES 初期化ベクタ･レジスタ 1（AES_IVR1）

アドレス･オフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[63:48]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[47:32]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVI[63:32]**：初期化ベクタ入力、ビット [63:32]
IVI[128:0] ビットフィールドの説明については、AES_IVR0 レジスタを参照してください。

34.7.11 AES 初期化ベクタ･レジスタ 2（AES_IVR2）

アドレス･オフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[95:80]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[79:64]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVI[95:64]**：初期化ベクタ入力、ビット [95:64]
IVI[128:0] ビットフィールドの説明については、AES_IVR0 レジスタを参照してください。

34.7.12 AES 初期化ベクタ･レジスタ 3（AES_IVR3）

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IVI[127:112]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IVI[111:96]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **IVI[127:96]**：初期化ベクタ入力、ビット [127:96]
IVI[128:0] ビットフィールドの説明については、AES_IVR0 レジスタを参照してください。

34.7.13 AES キー･レジスタ 4（AES_KEYR4）

アドレス･オフセット：0x30

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[159:144]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[143:128]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[159:128]**：暗号化キー、ビット [159:128]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.14 AES キー･レジスタ 5（AES_KEYR5）

アドレス･オフセット：0x34

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[191:176]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[175:160]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[191:160]**：暗号化キー、ビット [191:160]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.15 AES キー･レジスタ 6（AES_KEYR6）

アドレス･オフセット：0x38

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[223:208]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[207:192]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[223:192]**：暗号化キー、ビット [223:192]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

34.7.16 AES キー･レジスタ 7（AES_KEYR7）

アドレス･オフセット：0x3C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[255:240]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[239:224]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **KEY[255:224]**：暗号化キー、ビット [255:224]

KEY[255:0] ビットフィールドの説明については、AES_KEYR0 レジスタを参照してください。

注： **4〜7 のキーレジスタは、256 ビットのキー長を選択した場合のみ使用されます。128 ビットのキー長を選択した場合は有効ではありません（この場合、0〜3 のキーレジスタのみが使用されます）。**

34.7.17 AES サスペンド･レジスタ（AES_SUSPxR）

アドレスオフセット：0x040 + x * 0x4、（x = 0〜7）

リセット値：0x0000 0000

現在のタスクの AES 処理が優先順位の高いタスクを処理するためにサスペンドされる場合、AES プロセッサの内部レジスタステータスはすべてこれらのレジスタ内にあります。

サスペンド時、ソフトウェアは AES_SUSPxR レジスタ（x は 0〜7）の内容を読み出し、メモリに保存してから優先順位の高いタスクに AES プロセッサを使用します。完了時、ソフトウェアは保存した内容を、対応するサスペンドレジスタに復元してから、元のタスクを再開します。

注： **これらのレジスタは、GCM、GMAC または CCM 連鎖モードが選択されているときにのみ使用します。**

これらのレジスタは AES が有効である場合のみ読み出すことができます。AES が無効になっている状態で、これらのレジスタを読み出すと、0x0000 0000 が返されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SUSP[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SUSP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **SUSP[31:0]**：AES サスペンド

サスペンド動作時、対応する AES_SUSPxR レジスタのこのビットフィールドには、いずれかの内部 AES レジスタの値が格納されます。

34.7.18 AES レジスタ･マップ

表 328. AESレジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	AES_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NPBLB[3:0]				Res.	KEYSIZE	Res.	CHMOD[2]	Res.	GCMPH[1:0]		DMAOUTEN	DMAINEN	ERRIE	CCFIE	ERRC	CCFC	CHMOD[1:0]		MODE[1:0]		DATATYPE[1:0]		EN
	リセット値									0	0	0	0		0		0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	AES_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BUSY	WRERR	RDERR	CCF
	リセット値																													0	0	0	0
0x008	AES_DINR	DIN[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00C	AES_DOUTR	DOUT[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x010	AES_KEYR0	KEY[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x014	AES_KEYR1	KEY[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x018	AES_KEYR2	KEY[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x01C	AES_KEYR3	KEY[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	AES_IVR0	IVI[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x024	AES_IVR1	IVI[63:32]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x028	AES_IVR2	IVI[95:64]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x02C	AES_IVR3	IVI[127:96]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x030	AES_KEYR4	KEY[159:128]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x034	AES_KEYR5	KEY[191:160]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	AES_KEYR6	KEY[223:192]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x03C	AES_KEYR7	KEY[255:224]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x040	AES_SUSP0R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x044	AES_SUSP1R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 328. AESレジスタマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x048	AES_SUSP2R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04C	AES_SUSP3R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x050	AES_SUSP4R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x054	AES_SUSP5R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x058	AES_SUSP6R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x05C	AES_SUSP7R	SUSP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x060-0x3FF	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

35 リアルタイム･クロック（RTC）

35.1 概要

RTC は、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック（RTC）は、独立した BCD タイマ／カウンタです。RTC は、プログラム可能なアラーム割込みを備えた時刻クロック／カレンダを搭載しています。

供給電圧が動作範囲内にある間は、デバイスのステータス（実行モード、低電力モード、またはリセット中）に関係なく、RTC が停止することはありません。

RTC は V_{BAT} モードで機能します。

35.2 RTC の主な機能

RTC は、次の機能をサポートしています（図 529：RTC ブロック図を参照）。

- サブセカンド、秒、分、時（12 または 24 時間形式）、曜日、日、月、年に BCD（2 進化 10 進数）形式で対応するカレンダ。
- 月の 28 日、29 日（うるう年）、30 日、31 日の自動補正
- 2 つのプログラム可能なアラーム
- 1～32767 の RTC クロックパルスの動作中の補正。マスタクロックとの同期に使用可能
- リファレンスクロック検出：より正確な秒のクロックソース（50 または 60 Hz）の使用で、カレンダの精度を向上。
- 0.95 ppm の分解能を持つデジタル較正回路で、クオーツクリスタルの精度を補償
- カレンダの内容の保存に使用できるタイムスタンプ機能。この機能は、タイムスタンプピンでのイベント、またはタンパイベント、あるいは V_{BAT} モードへの切り替えでトリガ可能
- 分解能と周期がプログラム可能な周期的なイベント用 17 ビット自動再ロードウェイクアップタイマ（WUT）

RTC の電源は、V_{DD} 供給（存在する場合）から、または V_{BAT} ピンから電源を取得するスイッチ経由で供給されます。

RTC クロックソースは次のいずれかです。

- 32.768 kHz 外部クリスタル（LSE）
- 外部発振子またはオシレータ（LSE）
- 内部低電力 RC オシレータ（LSI、標準周波数 32 kHz）
- RCC でプリスケーラによって分周される高速外部クロック（HSE）

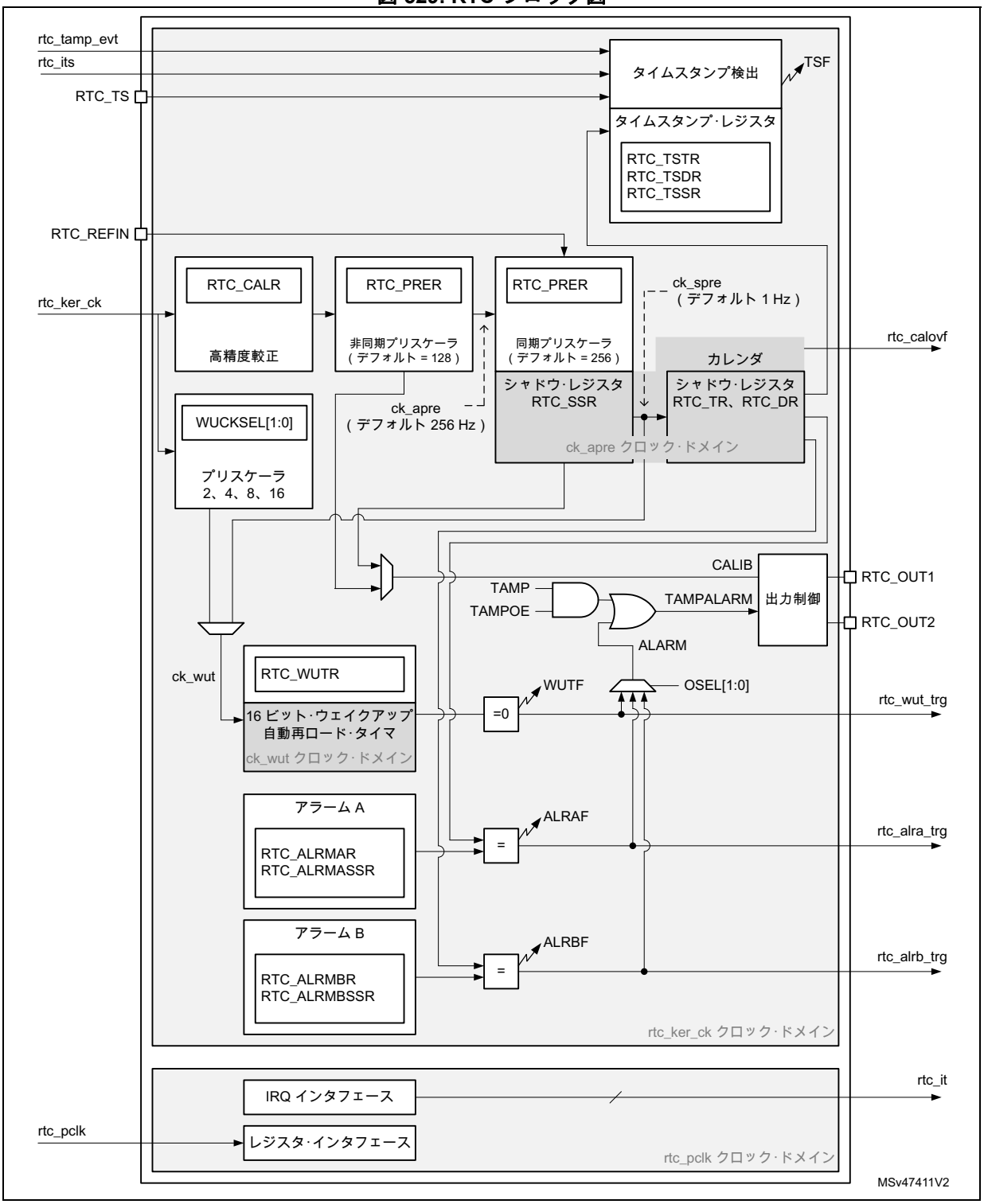
RTC は LSE によってクロック供給されている場合、V_{BAT} モードおよびすべての低電力モードで機能します。LSI によるクロック供給の場合、RTC は V_{BAT} モードでは動作しませんが、SHUTDOWN モードを除くすべての低電力モードで動作します。

すべての RTC イベント（アラーム、ウェイクアップタイマ、タイムスタンプ）で割込みを生成して、デバイスを低電力モードからウェイクアップさせることができます。

35.3 RTC の機能説明

35.3.1 RTC ブロック図

図 529. RTC ブロック図



35.3.2 RTC ピンおよび内部信号

表 329. RTC の入出力ピン

ピン名	信号タイプ	説明
RTC_TS	入力	RTC タイムスタンプ入力
RTC_REFIN	入力	RTC 50 または 60 Hz リファレンスクロック入力
RTC_OUT1	出力	RTC 出力 1
RTC_OUT2	出力	RTC 出力 2

- RTC_OUT1 および RTC_OUT2 は、次の 2 つの出力のうち 1 つを選択します。
  - CALIB：512 Hz または 1 Hz のクロック出力（LSE 周波数 32.768 kHz の場合）。この出力は、RTC_CR レジスタの COE ビットをセットして有効にします。
  - TAMPALRM：この出力は、TAMP 出力と ALARM 出力の論理和（OR）です。

ALARM は、アラーム A、アラーム B、またはウェイクアップ出力を選択する RTC_CR レジスタの OSEL [1:0] ビットを設定することで有効にします。TAMP は、タンパイベント出力を選択する RTC_CR レジスタの TAMPOE ビットをセットすることで有効にします。

表 330. RTC 内部入力／出力信号

内部信号名	信号タイプ	説明
rtc_ker_ck	入力	RTC カーネルクロック（本書では RTCCLK とも言います）
rtc_pclk	入力	RTC APB クロック
rtc_its	入力	RTC 内部タイムスタンプイベント
rtc_tamp_evt	入力	TAMP ペリフェラルで検出されたタンパイベント（内部または外部）
rtc_it	出力	RTC 割込み（詳細については、セクション 35.5 : RTC 割込みを参照してください）
rtc_alra_trg	出力	RTC アラーム A イベント検出トリガ
rtc_alrb_trg	出力	RTC アラーム B イベント検出トリガ
rtc_wut_trg	出力	RTC ウェイクアップタイマイベント検出トリガ
rtc_calovf	出力	RTC カレンダオーバーフロー

通常、RTC カーネルクロックは、RCC で他のクロックソースを選択できますが、32.768 kHz での LSE です（詳細は RCC を参照してください）。選択したクロックが LSE ではない場合、一部の低電力モードや V_{BAT} で使用できない機能があります。詳細については、セクション 35.4 : RTC 低電力モードを参照してください。

表 331. RTC 相互接続

信号名	転送元／転送先
rtc_its	電源制御（PWR）から：V_{BAT} 検出出力の主電源の損失／切り替え
rtc_tamp_evt	TAMP ペリフェラルから：tamp_evt
rtc_calovf	TAMP ペリフェラルへ：tamp_itamp5

トリガ出力は、その他のペリフェラルのトリガとして使用できます。

35.3.3 RTC および TAMP によって制御される GPIO

バッテリバックアップドメイン（V_{BAT}）に含まれる GPIO は、どの GPIO 設定であっても、これらの I/O での機能を備えたペリフェラルで直接制御されます。

RTC および TAMP の両方のペリフェラルには、これらの I/O での機能が搭載されています（セクション 36：タンパおよびバックアップレジスタ（TAMP）を参照）。

RTC_OUT1、RTC_TS、および TAMP_IN1 は同一ピン（PC13）上に配置されます。PC13 に配置された RTC および TAMP の機能は、すべての低電力モードと V_{BAT} モードで使用できます。

出力形式は、表 332に示す優先順位に従います。

表 332. PC13 設定(1)

PC13 ピンの機能		OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	TAMP1E（TAMP_IN1 入力イネーブル）	TSE（RTC_TS 入力イネーブル）
TAMPALRM 出力プッシュプル		01、10、または 11	0	無視	無視	0	0	無視	無視
		00	1						
		01、10、または 11	1						
TAMPALRM 出力オープンドレイン(2)	プルアップ・プルダウンなし	01、10、または 11	0	無視	無視	1	0	無視	無視
		00	1						
		01、10、または 11	1						
	内部プルアップ	01、10、または 11	0	無視	無視	1	1	無視	無視
		00	1						
		01、10、または 11	1						
CALIB 出力 PP		00	0	1	0	無視	無視	無視	無視
TAMP_IN1 入力フローティング		00	0	0	無視	無視	無視	1	0
		00	0	1	1				
		無視	無視	0					
RTC_TS および TAMP_IN1 入力フローティング		00	0	0	無視	無視	無視	1	1
		00	0	1	1				
		無視	無視	0					
RTC_TS 入力フローティング		00	0	0	無視	無視	無視	0	1
		00	0	1	1				
		無視	無視	0					

表 332. PC13 設定(1) （続き）

PC13 ピンの機能	OSEL[1:0]（ALARM 出力イネーブル）	TAMPOE（TAMPER 出力イネーブル）	COE（CALIB 出力イネーブル）	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	TAMP1E（TAMP_IN1 入力イネーブル）	TSE（RTC_TS 入力イネーブル）
ウェイクアップピンまたは標準 GPIO	00	0	0	無視	無視	無視	0	0
	00	0	1	1				
	無視	無視	0					

1. OD：オープンドレイン、PP：プッシュプル
2. この設定では、GPIO を入力で設定する必要があります。

さらに、OUT2EN ビットにより、RTC_OUT2 を PB2 ピンに出力することも可能です。この出力は、V_{BAT} モードでは使用できません。表 333 に示すとおり、OSEL、COE および OUT2EN の設定によって、RTC_OUT1 または RTC_OUT2 に別の機能を配置することができます。

PB2 では、GPIO はオルタネート機能として設定してください。

表 333. RTC_OUT の配置

OSEL[1:0] ビット（ALARM 出力イネーブル）	COE ビット（CALIB 出力イネーブル）	OUT2EN ビット	PC13 での RTC_OUT1	PB2 での RTC_OUT2
00	0	0	-	-
00	1		CALIB	-
01、10、または 11	無視		TAMPALRM	-
00	0	1	-	-
00	1		-	CALIB
01、10、または 11	0		-	TAMPALRM
01、10、または 11	1		TAMPALRM	CALIB

35.3.4 クロックとプリスケーラ

RTC クロックソース（RTCCLK）は、LSE クロック、LSI オシレータクロック、HSE クロックのうちから、クロックコントローラを介して選択されます。RTC クロックソースの設定に関する詳細は、セクション 7：リセットおよびクロック制御（RCC）を参照してください。

プログラム可能なプリスケーラステージで、カレンダの更新に使用する1 Hz のクロックを生成します。消費電力を最少に抑えるため、プリスケーラは以下に示す 2 つのプログラム可能なプリスケーラに分割されます（ 図 529：RTC ブロック図を参照）。

- RTC_PRER レジスタの PREDIV_A ビットで設定される 7 ビットの非同期プリスケーラ
- RTC_PRER レジスタの PREDIV_S ビットで設定される 15 ビットの同期プリスケーラ

注： **両方のプリスケーラを使用する場合は、非同期プリスケーラを高い値に設定して消費を最低限に抑えることをお勧めします。**

LSE 周波数 32.768 kHz で 1 Hz （ck_spre）の内部クロック周波数を得るため、非同期プリスケーラ分周比は 128、同期プリスケーラの分周比は 256 に設定されます。

最低分周比は 1、最大分周比は 2^{22} です。

これは、約 4 MHz の最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apre クロックは、サブセカンドダウンカウンタであるバイナリ RTC_SSR にクロックを供給するために使用されます。値がゼロになると、RTC_SSR は、PREDIV_S の内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spre クロックは、カレンダの更新に、または 16 ビットウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16 ビットウェイクアップ自動再ロードタイマを、プログラム可能な 4 ビット非同期プリスケーラで分周した RTCCLK で動作させることもできます（詳細は セクション 35.3.7：周期的自動ウェイクアップを参照）。

35.3.5 リアルタイムクロックとカレンダ

RTC カレンダの時刻および日付レジスタには、PCLK（APB クロック）と同期するシャドウレジスタからアクセスします。同期するための待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用 RTC_SSR
- 時刻用 RTC_TR
- 日付用 RTC_DR

RTCCLK サイクルごとに現在のカレンダ値がシャドウレジスタにコピーされ、RTC_ICSR レジスタの RSF ビットがセットされます（ セクション 35.6.10：RTC シフト制御レジスタ（RTC_SHIFTR）を参照）。STOP モードおよび STANDBY モードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 4 サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CR レジスタの BYPSHAD 制御ビットをセットすることにより、カレンダレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TR または RTC_DR レジスタを BYPSHAD = 0 の状態で読み出す際は、APB クロックの周波数 （f_{APB}）は、RTC クロック（f_{RTCCLK}）の周波数の 7 倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

35.3.6 プログラム可能なアラーム

RTC ユニットにはプログラム可能なアラーム（アラーム A とアラーム B）が搭載されています。以下にアラーム A の説明を示していますが、アラーム B にも転用できます。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAE ビットを通じて有効にします。

ALRAF は、カレンダのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASSR および RTC_ALRMAR にプログラムされている値と一致する場合は 1 にセットされます。各カレンダ項目は、RTC_ALRMAR レジスタの MSKx ビットおよび RTC_ALRMASSR レジスタの MASKSSx ビットで個別に選択できます。

アラームの割込みは、RTC_CR レジスタの ALRAIE ビットを通じて有効にします。

注意： **秒の項目が選択されている場合（RTC_ALRMAR で MSK1 ビットがリセットされている場合）、正しい動作を保証するため、RTC_PRER レジスタでセットされる同期プリスケーラの分周比は 3 以上である必要があります。**

アラーム A およびアラーム B（RTC_CR レジスタの OSEL[1:0] ビットで有効になっている場合）は、TAMPALRM 出力に送ることができます。TAMPALRM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

35.3.7 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16 ビットのプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は 17 ビットまで拡張できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットを通じて有効にします。

ウェイクアップタイマクロック入力 ck_wut には、次のものが使用できます。

- 2、4、8、または 16 分周した RTC クロック（RTCCLK）

  RTCCLK が LSE（32.768 kHz）である場合、最小分解能 61 µs で、ウェイクアップ割込み周期を 122 µs から 32 秒の範囲で設定できます。
- ck_spre（通常は 1 Hz の内部クロック）

  ck_spre 周波数が 1 Hz の場合、1 秒の分解能でウェイクアップ時間を 1 秒からおよそ 36 時間までの範囲で設定できます。このプログラム可能な広い時間範囲は、2 つの部分に分かれます。
  - WUCKSEL[2:1] = 10 の場合は 1 秒から 18 時間です。
  - WUCKSEL[2:1] = 11 の場合は約 18 時間から 36 時間です。この最後のケースでは、2^{16} が 16 ビットカウンタの現在値に追加されます。初期化シーケンスが完了すると（1518 ページのウェイクアップタイマのプログラミングを参照）、タイマがカウントダウンを開始します。ウェイクアップ機能が有効である場合、カウントダウンは低電力モードでアクティブのままとなります。さらに、カウンタが 0 に到達すると、RTC_SR レジスタの WUTF フラグがセットされ、ウェイクアップカウンタが再ロード値（RTC_WUTR レジスタ値）で自動的に再ロードされます。

その後、WUTF フラグはソフトウェアでクリアする必要があります。

RTC_CR レジスタの WUTIE ビットをセットして周期的ウェイクアップ割込みを有効にすると、デバイスは低電力モードを終了できます。

周期的ウェイクアップフラグは、RTC_CR レジスタの OSEL[1:0] ビットを通じて有効になっている場合に、TAMPALRM 出力に送ることができます。TAMPALRM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

低電力モード（SLEEP、STOP、STANDBY）と同様に、システムリセットもウェイクアップタイマには影響しません。

35.3.8 RTC の初期化と設定

RTC レジスタアクセス

RTC レジスタは、32 ビットのレジスタです。APB インタフェースは、RTC レジスタアクセスに 2 ウェイトステートを挿入します。ただし、BYPSHAD = 0 のときのカレンダシャドウレジスタへの読出しアクセスは除きます。

RTC レジスタ書込み保護

システムリセット後、電源制御ペリフェラルの DBP ビットによって、RTC レジスタは不要な書込みアクセスから保護されます（PWR 電源制御のセクションを参照）。RTC レジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

バックアップ ドメインリセット後、いくつかの RTC レジスタは書込み保護されます。

RTC レジスタへの書込みは、書込み保護レジスタ RTC_WPR にキーを書き込むことにより有効になります。

保護された RTC レジスタの書込み保護を解除するには、次のステップが必要です。

1. RTC_WPR レジスタに 0xCA を書き込みます。
2. RTC_WPR レジスタに 0x53 を書き込みます。

誤ったキーを書き込むと、書込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

カレンダの初期化と設定

時間形式やプリスケーラ設定を含むカレンダ時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ICSR レジスタで INIT ビットを 1 にセットして、初期化モードに入ります。このモードでは、カレンダカウンタが停止し、その値を更新することができます。
2. RTC_ICSR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. カレンダカウンタのための 1 Hz クロックを生成するには、RTC_PRER レジスタで両方のプリスケーラ分周比をプログラムします。
4. シャドウレジスタ（RTC_TR および RTC_DR）に時刻と日付の初期値をロードし、RTC_CR レジスタの FMT ビットを介して時間形式（12 時間または 24 時間）を設定します。
5. INIT ビットをクリアして初期化モードを終了します。その後、カレンダカウンタの実際の値が自動的にロードされ、4 RTCCLK クロックサイクル後にカウントが再開します。

初期化シーケンスが完了すると、カレンダがカウントを開始します。

注： **システムリセット後、アプリケーションは RTC_ICSR レジスタの INITS フラグを読み出し、カレンダが初期化されたか否かを確認できるようになります。このフラグが 0 であれば、カレンダの年の項目が バックアップ ドメインリセットデフォルト値（0x00）にセットされているため、初期化されていません。**

初期化後にカレンダを読み出すには、まずソフトウェアで RTC_ICSR レジスタの RSF フラグがセットされていることを確認する必要があります。

サマータイム

サマータイム管理は、RTC_CR レジスタの SUB1H ビット、ADD1H ビット、BKP ビットを介して行われます。

SUB1H または ADD1H を使用すると、ソフトウェアは初期化手順を踏まずに 1 度の操作で、カレンダから 1 時間引いたり足したりすることができます。

さらに、ソフトウェアは BKP ビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下にアラーム A の手順を示していますが、アラーム B にも転用できます。

1. RTC_CR の ALRAE をクリアしてアラーム A を無効にします。
2. アラーム A レジスタ（RTC_ALRMASSR/RTC_ALRMAR）をプログラムします。
3. RTC_CR レジスタで ALRAE をセットしてアラーム A を再び有効にします。

注： **RTC_CR レジスタの各変更は、クロック同期のため RTCCLK クロック約 2 サイクル後に有効になります。**

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値（RTC_WUTR の WUT[15:0]）の設定または変更には、次の手順が必要です。

1. RTC_CR の WUTE をクリアしてウェイクアップタイマを無効にします。
2. RTC_ICSR の WUTWF がセットされ、ウェイクアップ自動再ロードカウンタおよび WUCKSEL[2:0] ビットへのアクセスが許可されていることが確認されるまで WUTWF をポーリングします。このステップは、カレンダ初期化モードではスキップする必要があります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. ウェイクアップ自動再ロード値 WUT[15:0] およびウェイクアップクロック選択（RTC_CR の WUCKSEL[2:0] ビット）をプログラムします。RTC_CR で WUTE をセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウン再びを開始します。WUTWF ビットは、クロックの同期化により、WUTE クリア後、2 RTCCLK クロックサイクルまでにクリアされます。

35.3.9 カレンダの読出し

RTC_CR レジスタの BYPSHAD 制御ビットがクリアされている場合

RTC カレンダレジスタ（RTC_SSR、RTC_TR、および RTC_DR）を正しく読み出すには、APB1 クロック周波数（f_{PCLK}）が RTC クロック周波数（f_{RTCCLK}）の 7 倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APB1 クロック周波数が RTC クロック周波数の 7 倍未満である場合、ソフトウェアによってカレンダ時間と日付レジスタを 2 回読み出す必要があります。RTC_TR の 2 回目の読出しが 1 回目の読出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読出しアクセスを行う必要があります。どの場合も、APB1 クロック周波数は必ず RTC クロック周波数以上である必要があります。

RTC_ICSR レジスタの RSF ビットは、カレンダレジスタが RTC_SSR、RTC_TR、および RTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーはRTCCLK サイクルごとに行われます。3 つの値における一貫性を保証するため、RTC_SSR または RTC_TR のどちらかを読み出すと、高次カレンダシャドウレジスタの値は RTC_DR が読み出されるまでロックされます。ソフトウェアが 1 RTCCLK サイクル未満の間隔でカレンダの読出しアクセスを行う場合､最初のカレンダ読出し後

に RSF をソフトウェアでクリアする必要があり、その後ソフトウェアは、RSF ビットがセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再度読み出す必要があります。

低電力モード（STOP または STANDBY）からのウェイクアップ後は、RSF をソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度 RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再度読み出す必要があります。

RSF ビットは、ウェイクアップ後にクリアする必要がありますが、低電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化（1517 ページのカレンダの初期化と設定を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

同期（セクション 35.3.11：RTC の同期を参照）後、ソフトウェアは RSF がセットされるまで待ってから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

RTC_CR レジスタ（バイパスシャドウレジスタ）の BYPSHAD 制御ビットがセットされている場合

カレンダレジスタを読み出すと、カレンダカウンタの値が直接与えられるため、RSF ビットがセットされるのを待つ必要がありません。シャドウレジスタは低電力モード（STOP または STANDBY）では更新されないため、低電力モード終了後に特にこのような読出しが有用です。

BYPSHAD ビットが 1 にセットされている場合、レジスタへの 2 回の読出しアクセス間で RTCCLK エッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読出し操作中に RTCCLK エッジが発生した場合、レジスタの 1 つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを 2 回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わり、ソフトウェアはカレンダレジスタの最下位の数値を 2 回比較するだけで構いません。

注： **BYPSHAD = 1 の間、カレンダレジスタの読出し命令が完了するには 1 APB サイクルだけ余計に必要となります。**

35.3.10 RTC のリセット

カレンダシャドウレジスタ（RTC_SSR、RTC_TR、RTC_DR）および RTC ステータスレジスタ（RTC_ICSR）の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタは バックアップ ドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTC の現在のカレンダレジスタ、RTC 制御レジスタ（RTC_CR）、プリスケーラレジスタ（RTC_PRER）、RTC 較正レジスタ（RTC_CALR）、RTC シフトレジスタ（RTC_SHIFTR）、RTC タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、および RTC_TSDR）、ウェイクアップタイマレジスタ（RTC_WUTR）、およびアラーム A およびアラーム B レジスタ（RTC_ALRMASSR/RTC_ALRMAR および RTC_ALRMBSSR/RTC_ALRMBR）。

さらに、LSE クロックによって駆動されている際にリセットソースが バックアップ ドメインリセットと異なる場合、システムリセットがかかっても RTC は動作を続けます（システムリセットの影響を受けない RTC クロックソースの詳細については、RCC を参照してください）。バックアップ ドメインリセットが発生すると、RTC は停止し、すべての RTC レジスタがリセット値にセットされます。

35.3.11 RTC の同期

RTC は、高精度でリモートクロックと同期できます。サブセカンド項目（RTC_SSR または RTC_TSSSR）を読み出すと、リモートクロックによって維持されている時刻と RTC 間の正確なオフセットが計算できます。その後、RTC_SHIFTR を使用してほんの一瞬クロックを「シフト」することによって RTC を調整し、このオフセットを取り除くことができます。

RTC_SSR には、同期プリスケーラのカウンタの値が入っています。これにより、RTC によって維持されている正確な時刻を 1/(PREDIV_S + 1) 秒の分解能まで計算することができます。その結果、同期プリスケーラ値（PREDIV_S[14:0]）を増加させることにより分解能を改善できます。許可されている最大分解能（32768 Hz クロックで 30.52 µs）は、PREDIV_S を 0x7FFF にセットすることにより得られます。

ただし、PREDIV_S を増加させるということは、同期プリスケーラの出力を 1 Hz に維持するため PREDIV_A を減らす必要があることを意味します。このように、非同期プリスケーラの出力周波数が増加すると、RTC の動的消費電力が増加する場合があります。

RTC は、RTC シフト制御レジスタ（RTC_SHIFTR）を使って微調整できます。RTC_SHIFTR に書き込むことにより、1/（PREDIV_S + 1）秒の分解能で、クロックを最大 1 秒だけシフト（遅れ／進み）させることができます。このシフト操作の本質は、同期プリスケーラのカウンタ SS[15:0] に SUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。同時に ADD1S ビットがセットされた場合、1 秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。

注意： **シフト操作を始める前に、ユーザは SS[15] = 0 であることを確認し、オーバーフローが発生しないようにする必要があります。**

RTC_SHIFTR レジスタへの書込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示す SHPF フラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： **この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1 のときにファームウェアから RTC_SHIFTR への書込みはできません。**

35.3.12 RTC リファレンスクロック検出

RTC カレンダの更新は、リファレンスクロックである RTC_REFIN に同期させることができます。通常は商用電源（50 または 60 Hz）です。RTC_REFIN リファレンスクロックには、32.768 kHz LSE クロックより高い精度が必要です。RTC_REFIN 検出を有効にした際（RTC_CR の REFCKON ビットが 1 にセット）、カレンダは引き続き LSE クロックによって駆動され、RTC_REFIN はカレンダ更新周波数（1 Hz）の誤差の補正に使用されます。

各 1 Hz クロックエッジは、一番近い RTC_REFIN クロックエッジ（所与の時間枠内に見つかった場合）と比較されます。ほとんどの場合、2 つのクロックエッジは正しく整列しています。LSE クロックが不正確なために 1 Hz のクロックがずれた場合、RTC は 1 Hz のクロックを少しシフトさせ、その後の 1 Hz のクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTC は、32.768 kHz クォーツから生成される 256 Hz クロック（ck_apre）を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新（1 秒ごと）程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠は ck_apre 7 周期に等しくなります。その後のカレンダ更新では、ck_apre 3 周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_spre クロックを出力する非同期プリスケーラは強制的に再ロードされます。プリスケーラは同時に再ロードされるので、リファレンスクロックおよび 1 Hz のクロックが整列するタイミングには影響しません。クロックが整列していない場合、後の 1 Hz クロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した（ck_apre 3 周期の枠内でリファレンスクロックエッジが発生しない）場合、カレンダは LSE クロックのみを基準にして更新が継続されます。その後 RTC は ck_spre エッジを中心として ck_apre 7 周期という広い検出時間枠でリファレンスクロックを待ちます。

RTC_REFIN検出を有効にした場合、PREDIV_A および PREDIV_S を以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREVID_S = 0x00FF

注： **RTC_REFIN クロック検出は、STANDBY モードでは利用できません。**

35.3.13 RTC の高精度デジタル較正

RTC 周波数の精度は、-487.1 ppm から +488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル的に較正できます。周波数の修正は、一連の微調整（個々の RTCCLK パルスの追加や削除）によって行われます。このような調整は、短い期間で観測された場合でも RTC が十分に較正されるように、かなり広範に分散して行われます。

この高精度デジタル較正は、入力周波数が 32768 Hz の場合、RTCCLK 約 2^{20} パルスのサイクルまたは 32 秒の間で行われます。このサイクルは、RTCCLK によって駆動される 20 ビットカウンタ、cal_cnt[19:0] によって維持されます。

高精度較正レジスタ（RTC_CALR）によって較正サイクル中にマスクされる RTCCLK クロック･サイクル数を指定します。

- CALM[0] ビットを 1 に設定すると較正サイクルの中でちょうど 1 パルスがマスクされます。
- CALM[1] ビットを 1 にセットすると、さらに 2 サイクルがマスクされます。
- CALM[2] ビットを 1 にセットすると、さらに 4 サイクルがマスクされます。
- CALM[8] ビットを 1 にセットするまで続けると、256 クロックがマスクされます。

注： **CALM[8:0]（RTC_CALR）によって較正サイクル中にマスクされる RTCCLK パルス数を指定します。CALM[0] ビットを 1 に設定すると、cal_cnt[19:0] = 0x80000 になった時点で、較正サイクルのちょうど 1 パルスがマスクされます。CALM[1] = 1 では、さらに 2 サイクルがマスクされ（cal_cnt = 0x40000 および 0xC0000）、CALM[2] = 1 では、さらに 4 サイクルがマスクされ（cal_cnt = 0x20000/0x60000/0xA0000/0xE0000）、CALM[8] = 1 まで続けると、256 クロックがマスクされます（cal_cnt = 0xXX800）。**

CALM では、細かい分解能で RTC 周波数を最大 487.1 ppm 負の方向に調整することができ、CALP ビットでは周波数を 488.5 ppm 正の方向に調整することができます。CALP を“1”に設定することにより、実質上は、RTCCLK 2^{11} サイクルごとに、RTCCLK パルスが 1 パルス追加で挿入されます。すなわち、較正サイクルごとに 512 クロックが追加されることとなります。

CALM を CALP と合わせて使用すると較正サイクルの間に -511 から +512 RTCCLK サイクルのオフセットを追加でき、これは約 0.954 ppm の分解能で較正範囲 -487.1 ~ +488.5 ppm に換算されます。

有効較正周波数（FCAL）を入力周波数（FRTCCLK）に対して求める計算式は次のとおりです。

$$F_{CAL} = F_{RTCCLK} \times [1 + (CALP \times 512 - CALM) / (2^{20} + CALM - CALP \times 512)]$$

PREDIV_A < 3 の場合の較正

非同期プリスケーラ値（RTC_PRER レジスタの PREDIV_A ビット）が 3 未満の場合、CALP ビットを 1 にセットすることはできません。CALP がすでに 1 にセットされていて、PREDIV_A ビットが 3 未満の値にセットされた場合、CALP の設定値は無視され、CALP が 0 に設定された場合と同じように較正されます。

PREDIV_A が 3 未満の状態で較正を実施するには、各秒のカウントが 8 RTCCLK クロック･サイクル分早められるように同期プリスケーラ値（PREDIV_S）を小さくする必要があります。これは較正サ

イクル毎に 256 クロック･サイクル追加することに相当します。結果として、CALM ビットのみを使用して較正サイクルの間に実効的に 255 ~ 256 クロック･パルス（243.3 ~ 244.1 ppm の較正範囲に相当）を追加できます。

公称 RTCCLK 周波数が 32768 Hz で、PREDIV_A が 1（分周比 2）の場合、PREDIV_S を 16383 ではなく 16379（4 少ない）にセットする必要があります。また、PREDIV_A が 0 の場合、PREDIV_S を 32767 ではなく 32759（8 少ない）にセットする必要がありますので注意してください。

PREDIV_S をこのように減少させた場合、較正された入力クロックの有効周波数の式は次のようになります。

$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100（CALM 設定範囲の中間値）と等しくなるのが正しい設定です。

RTC 較正値の確認

RTC の精度は、RTCCLK の正確な周波数を測定し、正しい CALM 値および CALP 値を計算することにより保証されます。オプションの 1 Hz 出力が搭載されており、アプリケーションによって RTC 精度の測定と確認を行うことができます。

ある時間間隔で RTC の周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大 2 RTCCLK クロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は 32 秒です。

このモードを使用して正確に 32 秒で 1 Hz 出力の精度を測定すると、その精度は 0.477 ppm（較正分解能の制限により 32 秒で 0.5 RTCCLK サイクル）以内となることが保証されます。

- RTC_CALR レジスタの CALW16 ビットを 1 にセットして、較正サイクル周期を強制的に 16 秒にすることができます。

この場合、RTC 精度は最大誤差 0.954 ppm（16 秒で 0.5 RTCCLK サイクル）で 16 秒間で測定できます。ただし、較正分解能が下がるため、長期的な RTC 精度もまた 0.954 ppm に下がります。CALW16 が 1 にセットされると、CALM[0] ビットは 0 のままとなります。

- RTC_CALR レジスタの CALW8 ビットを 1 にセットして、較正サイクル周期を強制的に 8 秒にすることができます。

この場合、RTC 精度は最大誤差 1.907 ppm（8 秒で 0.5 RTCCLK サイクル）で 8 秒間で測定できます。長期的な RTC 精度もまた 1.907 ppm に下がります。CALW8 が 1 にセットされると、CALM[1:0] ビットは 00 のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ICSR/INITF = 0 の間でも、較正レジスタ（RTC_CALR）を動作中に更新することができます。

1. RTC_ICSR/RECALPF（再較正保留フラグ）をポーリングします。
2. このフラグが 0 にセットされている場合は、必要に応じて新しい値を RTC_CALR に書き込みます。すると、RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書込み動作後 ck_apre 3 サイクル以内に、新しい較正設定が有効になります。

35.3.14 タイムスタンプ機能

タイムスタンプは、RTC_CR レジスタの TSE ビットまたは ITSE ビットを 1 にセットすることにより有効になります。

TSE がセットされている場合：

RTC_TS ピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

TAMPTS がセットされている場合：

TAMP_INx ピンでタンパイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。

ITSE がセットされている場合：

内部タイムスタンプイベントが検出されると、タイムスタンプレジスタ（RTC_TSSSR、RTC_TSTR、RTC_TSDR）にカレンダが保存されます。内部タイムスタンプイベントは、V_{BAT} 供給への切り替えによって生成されます。

タイムスタンプイベントが発生すると、内部または外部イベントによって、RTC_SR レジスタのタイムスタンプフラグビット（TSF）がセットされます。イベントが内部の場合、RTC_SR レジスタの ITSF フラグもセットされます。

RTC_CR レジスタの TSIE ビットをセットすることにより、タイムスタンプイベントが発生したときに割込みが生成されます。

タイムスタンプフラグ（TSF）がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ（TSOVF）がセットされ、タイムスタンプレジスタ（RTC_TSTR および RTC_TSDR）は、その前のイベントの結果を維持します。

注： **同期処理のため、TSF はタイムスタンプイベント発生から ck_apre 2 サイクル後にセットされます。**

一方、TSOVF のセットに遅延はありません。これは、2 つのタイムスタンプイベントの発生したタイミングが近い場合、TSF がまだ“0”であっても TSOVF が“1”と検出される可能性があることを意味します。よって、TSOVF のポーリングは TSF がセットされた後に実施することをお勧めします。

注意： **TSF ビットのクリア処理を行った直後にタイムスタンプイベントが発生した場合、TSF ビットおよび TSOVF ビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、アプリケーションは TSF がすでに 1 と読み出されていない限り、0 クリアの処理を行ってはなりません。**

オプション機能として、タンパイベントによってタイムスタンプイベントを記録することもできます。RTC 制御レジスタ（RTC_CR）の TAMPTS 制御ビットの詳細を参照してください。

35.3.15 較正クロック出力

RTC_CR レジスタで COE ビットが 1 にセットされると、CALIB デバイス出力にリファレンスクロックが供給されます。

RTC_CR レジスタの COSEL ビットがリセットされ、かつ PREDIV_A = 0x7F である場合、CALIB 周波数は $f_{RTCCLK/64}$ です。これは 32.768 kHz の RTCCLK 周波数に対する 512 Hz の較正出力に相当します。立ち下がりエッジには軽いジッタがあるため、CALIB のデューティサイクルは不規則になります。したがって、立ち上がりエッジの使用が推奨されます。

COSEL がセットされ、かつ “PREDIV_S+1”がゼロ以外の 256 の倍数である場合（すなわち、PREDIV_S[7:0] = 0xFF）、CALIB 周波数は fRTCCLK/(256 * (PREDIV_A+1)) となります。これは、RTCCLK 周波数が 32.768 kHz で、プリスケーラデフォルト値（PREDIV_A = Ox7F、PREDIV_S = 0xFF）に対する 1 Hz の較正出力に相当します。

注： **CALIB 出力が選択されると、RTC_OUT1 ピンは自動的に設定されますが、RTC_OUT2 ピンはオルタネート機能として設定する必要があります。**

COSEL がクリアされると、CALIB 出力は非同期プリスケーラの 6 番目のステージの出力になります。

COSEL がセットされると、CALIB 出力は同期プリスケーラの 8 番目のステージの出力になります。

35.3.16 タンパおよびアラーム出力

RTC_CR レジスタの OSEL[1:0] 制御ビットを使用してアラーム出力 TAMPALRM を有効にし、出力となる機能を選択します。これらの機能は、RTC_SR レジスタの該当するフラグの内容を反映します。

RTC_CR レジスタの TAMPOE 制御ビットがセットされると、すべての外部および内部のタンパフラグが論理和され、TAMPALRM 出力に送られます。OSEL = 00 の場合、TAMPALRM 出力はタンパフラグのみ反映します。OSEL ≠ 00 の場合、TAMPALRM の信号でタンパフラグと、アラーム A、B、ウェイクアップフラグのいずれかがともに供給されます。

RTC_CR の POL 制御ビットは、POL が 1 にセットされているときに選択されたフラグビットの逆が出力されるよう、TAMPALRM 出力の極性を決定します。

TAMPALRM 出力

TAMPALRM ピンは、制御ビット TAMPALRM_TYPE（RTC_CR レジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。TAMPALRM_PU（RTC_CR）によって、出力モードで内部プルアップを適用できます。

注： **TAMPALRM 出力が一度有効化されると、それは RTC_OUT1に対して CALIB よりも優先されます。**

TAMPALRM 出力が選択されると、RTC_OUT1 ピンは自動的に設定されますがRTC_OUT2 ピンにオルタネート機能を設定する必要があります。TAMPALRM が RTC でオープンドレインに設定されている場合、RTC_OUT1 GPIO を入力として設定する必要があります。

35.4 RTC 低電力モード

表 334. 低消費電力モードが RTC に与える影響

モード	説明
SLEEP	影響なし。 RTC 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	RTC クロック･ソースが LSE または LSI の場合、RTC はアクティブのままです。RTC 割込みによって、デバイスは STOP モードを終了します。
STANDBY	RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC 割込みによって、デバイスは STANDBY モードを終了します。
SHUTDOWN	RTC クロックソースが LSE の場合、RTC はアクティブのままです。RTC 割込みによって、デバイスは SHUTDOWN モードを終了します。

次の表は、すべてのモードでの RTC ピンと機能の一覧です。

表 335. RTC ピン機能（モード共通）

機能	STANDBY モードおよび SHUTDOWN モード以外のすべての低電力モードで機能するか	STANDBY モードおよび SHUTDOWN モードで機能するか	V_{BAT} モードで使用可能な機能
RTC_TS	はい	はい	はい
RTC_REFIN	はい	いいえ	不可
RTC_OUT1	はい	はい	はい
RTC_OUT2	はい	いいえ	不可

35.5 RTC 割込み

割込みチャネルがマスク済み割込みステータスレジスタにセットされています。また、割込み出力も有効化されます。

表 336. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了	SHUTDOWN モードの終了
RTC	アラーム A	ALRAF	ALRAIE	CALRAF に 1 を書き込む	可能	はい(3)	可能(4)
	アラーム B	ALRBF	ALRBIE	CALRBF に 1 を書き込む	可能	可能 (3)	あり (4)
	タイムスタンプ	TSF	TSIE	CTSF に 1 を書き込む	可能	可能 (3)	あり (4)
	ウェイクアップタイマ割込み	WUTF	WUTIE	CWUTF に 1 を書き込む	可能	可能 (3)	あり (4)

1. イベントフラグは、RTC_SR レジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、RTC_MISR レジスタ内です。
3. STOP モードおよび STANDBY モードからのウェイクアップは、RTC クロック･ソースが LSE または LSI のときのみ可能です。
4. SHUTDOWN モードからのウェイクアップは、RTC クロック･ソースが LSE のときのみ可能です。

35.6 RTC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの72ページのセクション 1.2 を参照してください。

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

35.6.1 RTC 時刻レジスタ（RTC_TR）

RTC_TR は、カレンダ時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。1517 ページのカレンダの初期化と設定 および 1518 ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x00

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

35.6.2 RTC 日付レジスタ（RTC_DR）

RTC_DR は、カレンダ日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。1517 ページのカレンダの初期化と設定 および 1518 ページのカレンダの読出しを参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x04

バックアップ ドメイン･リセット値：0x0000 2101

システムリセット値：0x0000 2101（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **YT[3:0]**：BCD 形式での年の十の位

ビット 19:16 **YU[3:0]**：BCD 形式での年の一の位

ビット 15:13 **WDU[2:0]**：曜日

000：禁止
001：月曜日
.....
111：日曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

注：　カレンダは最大値に達すると停止し、ロールオーバーできません。

35.6.3 RTC サブセカンドレジスタ（RTC_SSR）

アドレスオフセット：0x08

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット値：0x0000 0000（BYPSHAD = 0 の場合。BYPSHAD = 1 の場合は、影響を受けません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS[15:0]**：サブセカンド値

SS[15:0] は、同期プリスケーラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。

秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)

注： SS は、シフト操作後に限り、PREDIV_S より大きな値となる場合があります。この場合、正確な時刻／日付は、RTC_TR/RTC_DR で示される値よりも 1 秒少ない値となります。

35.6.4 RTC 初期化制御およびステータスレジスタ（RTC_ICSR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x0C

バックアップ ドメイン･リセット値：0x0000 0007

システムリセット値：0bxxxx xxxx xxxx xxxx xxxx xxxx 000x xxxx（影響を受けません。ただし、0 にクリアされる INIT、INITF、RSF ビットを除きます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RECALPF
															r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INIT	INITF	RSF	INITS	SHPF	WUTWF	ALRBWF	ALRAWF
								rw	r	rc_w0	r	r	r	r	r

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RECALPF**：再較正保留フラグ

ソフトウェアによって RTC_CALR レジスタに書込みが行われると、RECALPF ステータスフラグが自動的に 1 にセットされ、RTC_CALR レジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは 0 に戻ります。動作中の再較正を参照してください。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **INIT**：初期化モード

0：フリーランニングモード

1：時刻と日付レジスタ（RTC_TR と RTC_DR）、およびプリスケーラレジスタ（RTC_PRER）のプログラムに使用する初期化モードです。INIT がリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット 6 **INITF**：初期化フラグ

このビットが 1 にセットされると、RTC は初期化状態となり、時刻、日付およびプリスケーラレジスタが更新できます。

0：カレンダレジスタを更新できません。

1：カレンダレジスタを更新できます。

ビット 5 **RSF**：レジスタ同期フラグ

このビットは、カレンダ･レジスタがシャドウ･レジスタ（RTC_SSR、RTC_TR、RTC_DR）にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中（SHPF = 1）の初期化モードで、またはバイパス･シャドウ･レジスタ･モード（BYPSHAD = 1）で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0：カレンダシャドウレジスタはまだ同期していません。

1：カレンダシャドウレジスタは同期しています。

ビット 4 **INITS**：初期化ステータス･フラグ

このビットは、カレンダの年の項目が 0 ではないとき（バックアップ ドメインリセット状態）にハードウェアによってセットされます。

0：カレンダは初期化されていません。

1：カレンダは初期化されています。

ビット 3 **SHPF**：シフト操作保留中

このフラグは、RTC_SHIFTR への書込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPF ビットに書き込んでも影響はありません。

0：保留中のシフト操作はありません。

1：保留中のシフト操作があります。

ビット 2 **WUTWF**：ウェイクアップ･タイマ書込みフラグ

このビットは、RTC_CR で WUTE ビットが 0 にセットされた後、WUT 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：初期化モード以外で、ウェイクアップタイマ設定は更新できません。

1：ウェイクアップタイマ設定は更新できます。

ビット 1 **ALRBWF**：アラーム B 書込みフラグ

このビットは、RTC_CR で ALRBE ビットが 0 にセットされた後、アラーム B 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム B は更新できません。

1：アラーム B は更新できます。

ビット 0 **ALRAWF**：アラーム A 書込みフラグ

このビットは、RTC_CR で ALRAE ビットが 0 にセットされた後、アラーム A 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0：アラーム A は更新できません。

1：アラーム A は更新できます。

35.6.5 RTC プリスケーラレジスタ（RTC_PRER）

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2 回の書込みアクセスに分けて行う必要があります。1517 ページのカレンダの初期化と設定 を参照してください。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x10

バックアップ ドメイン･リセット値：0x007F 00FF

システム･リセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PREDIV_S[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **PREDIV_A[6:0]**：非同期プリスケーラ分周比

非同期分周比です。
ck_apre 周波数 = RTCCLK 周波数／(PREDIV_A+1)

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PREDIV_S[14:0]**：同期プリスケーラ分周比

同期分周比です。
ck_spre 周波数 = ck_apre 周波数／(PREDIV_S+1)

35.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ICSR の WUTWF が 1 にセットされているときにのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x14

バックアップ ドメイン･リセット値：0x0000 FFFF

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WUT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTE が 1 にセット）なとき、ck_wut の（WUT[15:0] + 1）サイクルごとに WUTF フラグがセットされます。ck_wut の周期は、RTC_CR レジスタの WUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1 のとき、ウェイクアップタイマは 17 ビットとなり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットである WUT[16] となります。

WUTF の最初のアサートは、WUTE がセットされてから WUT と ck_wut の（WUT + 1）サイクルの間に発生します。WUCKSEL[2:0] = 011 (RTCCLK/2) のときに WUT[15:0] を 0x0000 にセットすることはできません。

35.6.7 RTC 制御レジスタ（RTC_CR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x18

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	Res.	Res.	TAMPOE	TAMPTS	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31 **OUT2EN**：RTC_OUT2 出力は有効です。

このビットをセットして、以下のように RTC_OUT2 の RTC 出力を再配置できます。

OUT2EN=0：RTC 出力 2 無効

OSEL ≠ 00 または TAMPOE = 1 の場合：RTC_OUT1 に TAMPALRM が出力されます。

OSEL = 00 かつ TAMPOE = 0 かつ COE = 1 の場合：RTC_OUT1 に CALIB が出力されます。

OUT2EN=1：RTC 出力 2 有効

（OSEL ≠ 00 または TAMPOE = 1）かつ COE = 0 の場合：RTC_OUT2 に TAMPALRM が出力されます。

OSEL = 00 かつ TAMPOE = 0 かつ COE = 1 の場合：RTC_OUT2 に CALIB が出力されます。

（OSEL ≠ 00 または TAMPOE = 1）かつ COE = 1 の場合：RTC_OUT2 に CALIB が、RTC_OUT1 に TAMPALRM が出力されます。

ビット 30 **TAMPALRM_TYPE**：TAMPALRM 出力形式

0：TAMPALRM はプッシュプル出力になります。

1：TAMPALRM はオープンドレイン出力になります。

ビット 29 **TAMPALRM_PU**：TAMPALRM プルアップイネーブル

0：TAMPALRM 出力でプルアップは適用されません。

1：TAMPALRM 出力でプルアップは適用されます。

ビット 28:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **TAMPOE**：TAMPALRM でのタンパ検出出力イネーブル

0：タンパフラグが TAMPALRM で送られません。

1：タンパフラグが OSEL による信号および POL による極性とともに TAMPALRM で送られます。

ビット 25 **TAMPTS**：タンパ検出イベント時のタイムスタンプのアクティブ化

0：タンパ検出イベントがあっても、RTC タイムスタンプは保存されません。

1：タンパ検出イベント時、RTC タイムスタンプが保存されます。

RTC_CR レジスタで TSE = 0 であっても TAMPTS は有効です。タイムスタンプフラグはタンパフラグの後にセットされるため、TAMPTS と TSIE がセットされている場合、タンパ割込みを無効にして、2 つの割込みを処理しないようにすることを推奨します。

ビット 24 **ITSE**：内部イベントでのタイムスタンプイネーブル

0：内部イベントでのタイムスタンプは無効です。

1：内部イベントでのタイムスタンプは有効です。

ビット 23 **COE**：較正出力イネーブル

このビットは、CALIB 出力を有効にします。

0：較正出力は無効です。

1：較正出力は有効です。

ビット 22:21 **OSEL[1:0]**：出力選択

これらのビットは、TAMPALRM 出力に送られるフラグの選択に使用します。

00：出力は無効です。

01：アラーム A 出力は有効です。

10：アラーム B 出力は有効です。

11：ウェイクアップ出力は有効です。

ビット 20 **POL**：出力極性

このビットは、TAMPALRM 出力の極性の設定に使用します。

0 : ALRAF/ALRBF/WUTF が（OSEL[1:0] に応じて）アサートされた場合、または TAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはハイになります。

1 : ALRAF/ALRBF/WUTF が（OSEL[1:0] に応じて）アサートされた場合、または TAMPxF/ITAMPxF がアサートされた場合（TAMPOE = 1）、このピンはローになります。

ビット 19 **COSEL**：較正出力選択

COE = 1 のとき、このビットによって CALIB に出力される信号を選択します。

0：較正出力は 512 Hz です。

1：較正出力は 1 Hz です。

これらの周波数は、RTCCLK が 32.768 kHz で、プリスケーラがデフォルト値（PREDIV_A = 127 および PREDIV_S = 255）の場合に有効です。セクション 35.3.15 : 較正クロック出力を参照してください。

ビット 18 **BKP**：バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザが書き込むことができます。

ビット 17 **SUB1H**：1 時間差し引き（冬時間変更）

このビットを初期化モード以外のときにセットすると、現在時刻が 0 でない場合にカレンダ時刻から 1 時間を差し引きます。このビットは常に 0 として読み出されます。

現在時間が 0 のときにこのビットをセットしても、影響はありません。

0：影響なし。

1：現在時刻から 1 時間差し引きます。これは、冬時間変更に使用できます。

ビット 16 **ADD1H**：1 時間加算（サマータイム変更）

このビットを初期化モード以外のときにセットすると、カレンダ時刻に 1 時間加算します。このビットは常に 0 として読み出されます。

0：影響なし。

1：現在時刻に 1 時間加算します。これは、サマータイム変更に使用できます。

ビット 15 **TSIE**：タイムスタンプ割込みイネーブル

0：タイムスタンプ割込みは無効です。

1：タイムスタンプ割込みは有効です。

ビット 14 **WUTIE**：ウェイクアップ･タイマ割込みイネーブル

0：ウェイクアップタイマ割込みは無効です。

1：ウェイクアップタイマ割込みは有効です。

ビット 13 **ALRBIE**：アラーム B 割込みイネーブル

0：アラーム B 割込みは無効です。

1：アラーム B 割込みは有効です。

ビット 12 **ALRAIE**：アラーム A 割込みイネーブル

0：アラーム A 割込みは無効です。

1：アラーム A 割込みは有効です。

ビット 11 **TSE**：タイムスタンプイネーブル

0：タイムスタンプは無効です。

1：タイムスタンプは有効です。

ビット 10 **WUTE**：ウェイクアップ･タイマ･イネーブル

0：ウェイクアップタイマは無効です。

1：ウェイクアップタイマは有効です。

注： ウェイクアップタイマが無効の場合は、再度有効にする前に WUTWF=1 になるまで待ちます。

ビット 9 **ALRBE**：アラーム B イネーブル

0：アラーム B は無効です。

1：アラーム B は有効です。

ビット 8 **ALRAE**：アラーム A イネーブル

0：アラーム A は無効です。

1：アラーム A は有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **FMT**：時間形式

0：24 時間／日形式

1：AM／PM 時間形式

ビット 5 **BYPSHAD**：シャドウレジスタをバイパスします。

0：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、シャドウレジスタから取得され、これらは 2 RTCCLK サイクルごとに 1 回更新されます。

1：カレンダ値は（RTC_SSR、RTC_TR、RTC_DR から読み出す場合）、カレンダカウンタから直接取得されます。

注： APB1 クロックの周波数が RTCCLK の 7 倍未満である場合、BYPSHAD は 1 にセットする必要があります。

ビット 4 **REFCKON**：RTC_REFIN リファレンスクロック検出イネーブル（50 または 60 Hz）

0：RTC_REFIN 検出は無効です。

1：RTC_REFIN 検出は有効です。

注： PREDIV_S は 0x00FF である必要があります。

ビット 3 **TSEDGE**：タイムスタンプ･イベント･アクティブ･エッジ

0：RTC_TS 入力の立ち上がりエッジによってタイムスタンプイベントを生成します。

1：RTC_TS 入力の立ち下がりエッジによってタイムスタンプイベントを生成します。

不要な TSF 設定を回避するため、TSEDGE が変化した場合には TSE をリセットする必要があります。

ビット 2:0 **WUCKSEL[2:0]**：ck_wut ウェイクアップクロックの選択

000：RTC/16 クロックが選択されます。

001：RTC/8 クロックが選択されます。

010：RTC/4 クロックが選択されます。

011：RTC/2 クロックが選択されます。

10x：ck_spre（通常は 1 Hz）クロックが選択されます。

11x：ck_spre（通常は 1 Hz）クロックが選択され、2^{16} が WUT カウンタ値に加算されます。

注： **初期化モード（RTC_ICSR/INITF = 1）の場合のみ、このレジスタのビット 6 および 4 が書き込めます。**

WUT = ウェイクアップユニットカウンタ値WUT = (0x0000～0xFFFF) + 0x10000（WUCKSEL[2:1] = 11 の場合追加されます。）

このレジスタのビット 2～0 は、RTC_CR の WUTE ビット = 0 かつ RTC_ICSR の WUTWF ビット = 1 の場合にのみ書き込めます。

カレンダの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1H および SUB1H の変更は、次の秒から有効になります。

35.6.8 RTC 書込み保護レジスタ（RTC_WPR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
								w	w	w	w	w	w	w	w

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY[7:0]**：書込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に 0x00 が返されます。

RTC レジスタの書込み保護解除方法については、RTC レジスタ書込み保護を参照してください。

35.6.9 RTC 較正レジスタ（RTC_CALR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x28

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	CALW8	CALW 16	Res.	Res.	Res.	Res.	CALM[8:0]								
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALP**：RTC 周波数の 488.5 ppm 増加

0：RTCCLK パルスは加えられません。

1：RTCCLK の 2^{11} パルスごとに 1 パルス効果的に挿入されます（周波数が 488.5 ppm 増加）。

この機能は、CALM と共に使用されることを想定しており、カレンダの周波数を高分解能で下げることができます。入力周波数が 32768 Hz の場合、32 秒枠の間に追加される RTCCLK パルスの数は次のように算出されます：(512 * CALP) - CALM

セクション 35.3.13：RTC の高精度デジタル較正を参照してください。

ビット 14 **CALW8**：8 秒較正サイクル周期の使用

CALW8 が 1 にセットされると、8 秒較正サイクル周期が選択されます。

注： CALW8 = 1 の場合、CALM[1:0] は 00 に固定されます。セクション 35.3.13：RTC の高精度デジタル較正を参照してください。

ビット 13 **CALW16**：16 秒較正サイクル周期の使用

CALW16 が 1 にセットされると、16 秒較正サイクル周期が選択されます。CALW8=1 の場合、このビットを 1 にセットすることはできません。

注： CALW16 = 1 の場合、CALM[0] は 0 に固定されます。セクション 35.3.13：RTC の高精度デジタル較正を参照してください。

ビット 12:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **CALM[8:0]**：較正マイナス

RTCCLK 2^{20} パルス（入力周波数が 32768 Hz の場合 32 秒）内の CALM をマスクすることによって、カレンダの周波数が下げられます。この方法により、カレンダの周波数を 0.9537 ppm の分解能で下げることができます。

カレンダの周波数を上げるには、この機能を CALP と共に使用する必要があります。セクション 35.3.13：1521 ページのRTC の高精度デジタル較正を参照してください。

35.6.10 RTC シフト制御レジスタ（RTC_SHIFTR）

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x2C

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS[14:0]														
	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **ADD1S**：1 秒加算

0：影響なし。

1：時計／カレンダに 1 秒加算します。

このビットは書込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ICSR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS（下記説明を参照）と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SUBFS[14:0]**：秒の小数部差し引き

このビットは書込み専用であり、常に 0 として読み出されます。シフト操作が保留中（RTC_ICSR で SHPF = 1）の場合、このビットに書き込んでも影響はありません。

SUBFS に書き込まれた値は、同期プリスケーラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き（遅延）されます。

遅れ（秒）= SUBFS / (PREDIV_S + 1)

ADD1S 機能が SUBFS と共に用いられた場合、秒の小数部を効果的にクロックに加算する（クロックを進める）ことができ、実際のクロックの進みは次の式のとおりとなります。

進み（秒）= (1 - (SUBFS / (PREDIV_S + 1)))

注： SUBFS に書き込むことにより RSF はクリアされます。その後、ソフトウェアが RSF = 1 まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実になります。

35.6.11 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x30

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]			
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

35.6.12 RTC タイムスタンプ日付レジスタ（RTC_TSDR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x34

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT[1:0]		DU[3:0]			
r	r	r	r	r	r	r	r			r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **WDU[2:0]**：曜日

ビット 12 **MT**：BCD 形式での月の十の位

ビット 11:8 **MU[3:0]**：BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]**：BCD 形式での日の十の位

ビット 3:0 **DU[3:0]**：BCD 形式での日の一の位

35.6.13 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_SR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x38

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 SS[15:0]：サブセカンド値

SS[15:0] は、タイムスタンプイベントが発生したときの同期プリスケーラのカウンタの値です。

35.6.14 RTC アラーム A レジスタ（RTC_ALRMAR）

このレジスタは、RTC_ICSR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x40

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム A 日付マスク

0：日付／曜日が一致すると、アラーム A がセットされます。

1：アラーム A の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択
0：DU[3:0] は日付の一の位を表します。
1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム A 時マスク
0：時が一致すると、アラーム A がセットされます。
1：アラーム A の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム A 分マスク
0：分が一致すると、アラーム A がセットされます。
1：アラーム A の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム A 秒マスク
0：秒が一致すると、アラーム A がセットされます。
1：アラーム A の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

35.6.15 RTC アラーム A サブセカンドレジスタ（RTC_ALRMASSR）

このレジスタは、RTC_ICSR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x44

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0：アラーム A に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。

1：アラーム A の比較では SS[14:1] を無視します。SS[0] のみ比較されます。

2：アラーム A の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。

3：アラーム A の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。

.....

12：アラーム A の比較では SS[14:12] を無視します。SS[11:0] が比較されます。

13：アラーム A の比較では SS[14:13] を無視します。SS[12:0] が比較されます。

14：アラーム A の比較では SS[14] を無視します。SS[13:0] が比較されます。

15：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。

同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

注： 同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム A をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

35.6.16 RTC アラーム B レジスタ（RTC_ALRMBR）

このレジスタは、RTC_ICSR の ALRBWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、1517 ページのRTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x48

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSK4	WD SEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **MSK4**：アラーム B 日付マスク

0：日付／曜日が一致すると、アラーム B がセットされます。

1：アラーム B の比較では日付／曜日を無視します。

ビット 30 **WDSEL**：曜日選択

0：DU[3:0] は日付の一の位を表します。

1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位

ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

ビット 23 **MSK3**：アラーム B 時マスク
0：時が一致すると、アラーム B がセットされます。
1：アラーム B の比較では時を無視します。

ビット 22 **PM**：AM/PM 表記
0：AM または 24 時間形式
1：PM

ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位

ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

ビット 15 **MSK2**：アラーム B 分マスク
0：分が一致すると、アラーム B がセットされます。
1：アラーム B の比較では分を無視します。

ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

ビット 7 **MSK1**：アラーム 秒マスク
0：秒が一致すると、アラーム B がセットされます。
1：アラーム B の比較では秒を無視します。

ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位

35.6.17 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CR レジスタの ALRBE がリセットされた場合、または初期化モードの場合にのみ書き込めます。

このレジスタは書込み保護されています。書込みアクセスの手順は、セクション ：RTC レジスタ書込み保護を参照してください。

アドレスオフセット：0x4C

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]**：このビットから始まる最上位ビットのマスク

0x0：アラーム B に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
0x1：アラーム B の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
0x2：アラーム B の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
0x3：アラーム B の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
.....
0xC：アラーム B の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
0xD：アラーム B の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
0xE：アラーム B の比較では SS[14] を無視します。SS[13:0] が比較されます。
0xF：アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。
同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]**：サブセカンド値

この値が同期プリスケーラのカウンタの内容と比較され、アラーム B をアクティブ化するかどうかを決定します。0～MASKSS-1 のビットだけが比較されます。

35.6.18 RTC ステータスレジスタ（RTC_SR）

アドレスオフセット：0x50

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ITSF**：内部タイムスタンプ･フラグ

このフラグは、内部イベントでタイムスタンプが発生したときに、ハードウェアによってセットされます。

ビット 4 **TSOVF**：タイムスタンプ･オーバーフロー･フラグ

このフラグは、TSF が既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。
TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **TSF**：タイムスタンプ･フラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。
ITSF フラグがセットされている場合、ITSF とともに TSF をクリアする必要があります。

ビット 2 **WUTF**：ウェイクアップ･タイマ･フラグ

このフラグは、ウェイクアップ自動再ロードカウンタが 0 に到達したときに、ハードウェアによってセットされます。
このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 1 **ALRBF**：アラーム B フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム B レジスタ（RTC_ALRMBR）と一致したときにハードウェアによってセットされます。

ビット 0 **ALRAF**：アラーム A フラグ

このフラグは、時刻／日付レジスタ（RTC_TR および RTC_DR）がアラーム A レジスタ（RTC_ALRMAR）と一致したときにハードウェアによってセットされます。

注： このレジスタのビットは、RTC_SCR レジスタで対応するクリアビットをセットしてから 2 APB クロックサイクル後にクリアされます。

35.6.19 RTC マスク済み割込みステータスレジスタ（RTC_MISR）

アドレスオフセット：0x54

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
										r	r	r	r	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **ITSMF**：内部タイムスタンプ･マスク済みフラグ

このフラグは、内部イベントでタイムスタンプが発生しかつタイムスタンプ割込みが生じたとき、ハードウェアによってセットされます。

ビット 4 **TSOVMF**：タイムスタンプ･オーバーフロー･マスク済みフラグ

このフラグは、TSMF がすでにセットされている間にタイムスタンプ割込みが発生した場合に、ハードウェアによってセットされます。

TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **TSMF**：タイムスタンプ･マスク済みフラグ

このフラグは、タイムスタンプ割込みが発生したときに、ハードウェアによってセットされます。

ITSF フラグがセットされている場合、ITSF とともに TSF をクリアする必要があります。

ビット 2 **WUTMF**：ウェイクアップ･タイマ･マスク済みフラグ

このフラグは、ウェイクアップタイマ割込みが発生したときに、ハードウェアによってセットされます。

このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 1 **ALRBMF**：アラーム B マスク済みフラグ

このフラグは、アラーム B 割込みが発生したときに、ハードウェアによってセットされます。

ビット 0 **ALRAMF**：アラーム A マスク済みフラグ

このフラグは、アラーム A 割込みが発生したときに、ハードウェアによってセットされます。

35.6.20 RTC ステータス･クリア･レジスタ（RTC_SCR）

アドレスオフセット：0x5C

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
										w	w	w	w	w	w

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CITSF**：内部タイムスタンプ･フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ITSF ビットがクリアされます。

ビット 4 **CTSOVF**：タイムスタンプ･オーバーフロー･フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの TSOVF ビットがクリアされます。

TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。さもないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 3 **CTSF**：タイムスタンプ･フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの TSOVF ビットがクリアされます。

ITSF フラグがセットされている場合、CRSF と CITSF をセットして、ITSF とともに TSF をクリアする必要があります。

ビット 2 **CWUTF**：ウェイクアップ･タイマ･フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの WUTF ビットがクリアされます。

ビット 1 **CALRBF**：アラーム B フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ALRBF ビットがクリアされます。

ビット 0 **CALRAF**：アラーム A フラグのクリア

このビットに 1 を書き込むと、RTC_SR レジスタの ALRAF ビットがクリアされます。

35.6.21 RTC レジスタマップ

表 337. RTC レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	RTC_TR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT [1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x04	RTC_DR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	YT[3:0]				YU[3:0]				WDU[2:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0]			
	リセット値									0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1			0	0	0	0	0	1
0x08	RTC_SSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	RTC_ICSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RECALPF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	INIT	INITF	RSF	INITS	SHPF	WUT WF	ALRBWF	ALRAWF
	リセット値																0									0	0	0	0	0	1	1	1
0x10	RTC_PRER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PREDIV_A[6:0]							PREDIV_S[14:0]															
	リセット値										1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0x14	RTC_WUTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUT[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x18	RTC_CR	OUT2EN	TAMPALRM_TYPE	TAMPALRM_PU	Res.	Res.	TAMPOE	TAMPTS	ITSE	COE	O SEL [1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	Res.	FMT	BYPSHAD	REFCKON	TSEDGE	WUCK SEL[2:0]		
	リセット値	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x24	RTC_WPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x28	RTC_ CALR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CALP	CALW8	CALW16	Res.	Res.	Res.	Res.	CALM[8:0]								
	リセット値																	0	0	0					0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SUBFS[14:0]														
	リセット値	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	RTC_TSTR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PM	HT[1:0]		HU[3:0]				Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x34	RTC_TSDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDU[1:0]			MT	MU[3:0]				Res.	Res.	DT [1:0]		DU[3:0]			
	リセット値																	0	0	0	0	0	0	0	0			0	0	0	0	0	0
0x38	RTC_TSSSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x40	RTC_ALRMAR	MSK4	WDSEL	DT [1:0]		DU[3:0]				MSK3	PM	HT [1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

表 337. RTC レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x44	RTC_ALRMASSR	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	RTC_ALRMBR	MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]				MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	RTC_ALRMBSSR	Res.	Res.	Res.	Res.	MASKSS[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	SS[14:0]														
	リセット値					0	0	0	0										0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	RTC_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSF	TSOVF	TSF	WUTF	ALRBF	ALRAF
	リセット値																											0	0	0	0	0	0
0x54	RTC_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITSMF	TSOVMF	TSMF	WUTMF	ALRBMF	ALRAMF
	リセット値																											0	0	0	0	0	0
0x5C	RTC_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITSF	CTSOVF	CTSF	CWUTF	CALRBF	CALRAF
	リセット値																											0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

36 タンパおよびバックアップレジスタ（TAMP）

36.1 概要

32（カテゴリ 3 およびカテゴリ 4 デバイス）または 16（カテゴリ 2 デバイス）個の 32 ビットバックアップレジスタは、すべての低電力モードと V_{BAT} モードで保持されます。これらのレジスタは、内容がタンパ検出回路によって保護されるため、機密データの格納に使用できます。3 本のタンパピンおよび 4 個の内部タンパが対タンパ検出に利用できます。外部タンパピンは、フィルタの有無にかかわらず、エッジ検出またはレベル検出のどちらにも設定できます。

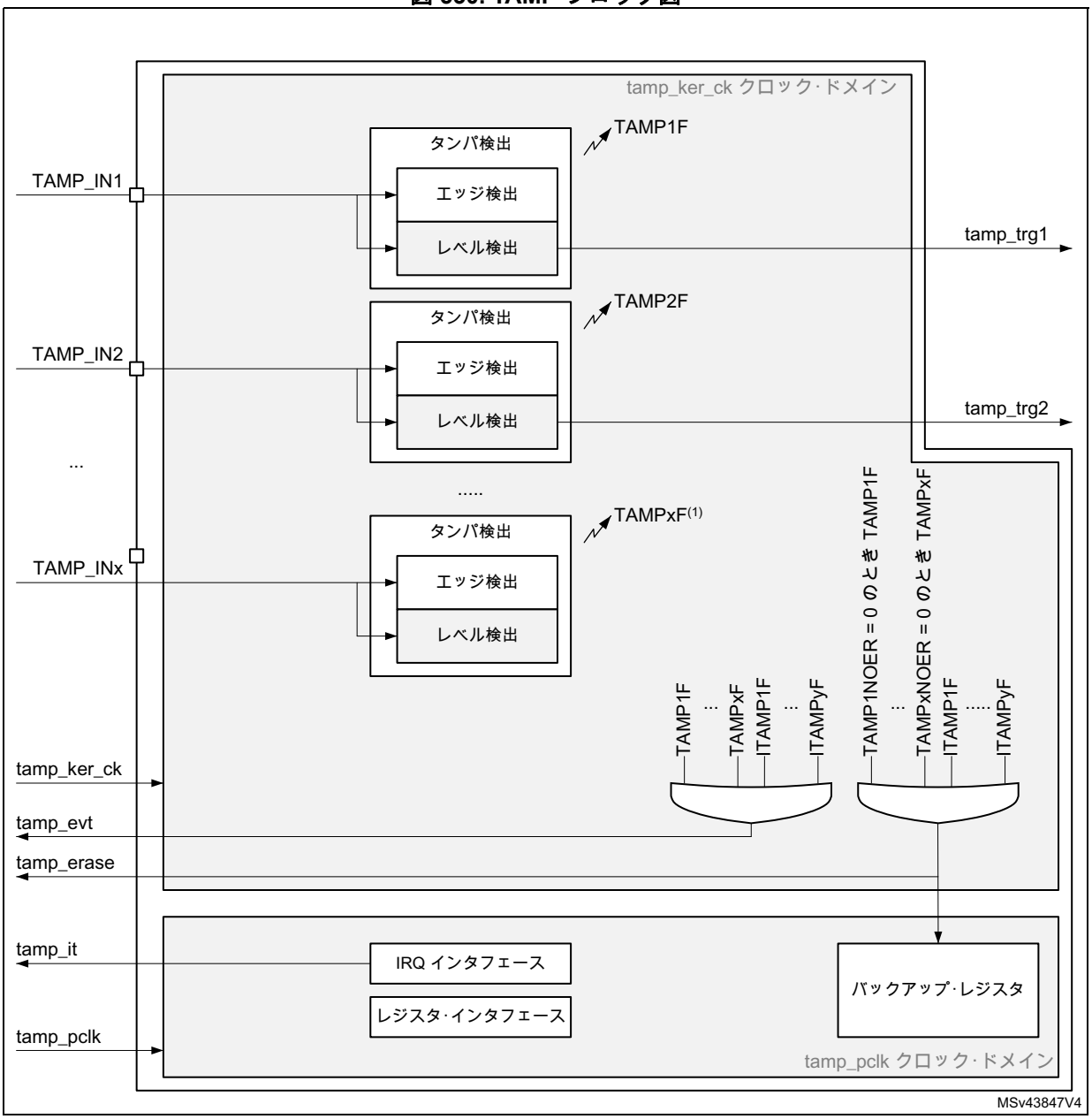
36.2 TAMP の主な機能

- 32（カテゴリ 3 およびカテゴリ 4 デバイス）または 16（カテゴリ 2 デバイス）個のバックアップレジスタ：
  - バックアップレジスタ（TAMP_BKPxR）は、V_{DD} 電源がオフに切り替わったときに V_{BAT} による電源がオンの状態になる RTC ドメインに実装されます。
- 3 個の外部タンパ検出イベント
  - 外部のパッシブタンパは、フィルタおよび内部プルアップが設定可能です。
- 4 個の内部タンパイベント
- あらゆるタンパ検出によって RTC タイムスタンプイベントを生成可能
- あらゆるタンパ検出によって バックアップレジスタ を消去可能

36.3 TAMP の機能詳細

36.3.1 TAMP ブロック図

図 530. TAMP ブロック図



1. 外部および内部タンパの数は製品により異なります。

36.3.2 TAMP ピンおよび内部信号

表 338. TAMP の入出力ピン

ピン名	信号タイプ	説明
TAMP_INx（x = ピンインデックス）	入力	タンパ入力ピン

表 339. TAMP 内部入力／出力信号

内部信号名	信号タイプ	説明
tamp_ker_ck	入力	TAMP カーネルクロック（rtc_ker_ck に接続、本書では RTCCLK とも言います）
tamp_pclk	入力	TAMP APB クロック（rtc_pclk に接続）
tamp_itamp[y] （y = 信号インデックス）	入力	内部タンパイベントソース
tamp_evt	出力	タンパイベント検出（内部または外部） tamp_evt は RTC タイムスタンプイベントを生成するために使用されます。
tamp_erase	出力	タンパイベント検出後のデバイス機密情報消去リクエスト（内部または外部）
tamp_it	出力	TAMP 割込み（詳細については、セクション 36.5 : TAMP 割込みを参照してください）
tamp_trg[x] （x = 信号インデックス）	出力	タンパ検出トリガ

通常、TAMP カーネルクロックは、RCC で他のクロックソースを選択できますが、32.768 kHz での LSE です（詳細は RCC を参照してください）。選択したクロックが LSE ではない場合、一部の低電力モードや V_{BAT} で使用できない検出モードがあります（詳細は、セクション 36.4 : TAMP の低電力モードを参照してください）。

表 340. TAMP 相互接続

信号名	転送元／転送先
tamp_evt	タイムスタンプイベントの生成に使用される rtc_tamp_evt
tamp_erase	tamp_erase 信号は、次にリストされているデバイス機密情報を消去するために使用されます。バックアップレジスタ
tamp_itamp3	LSE 監視
tamp_itamp4	HSE 監視
tamp_itamp5	RTC カレンダ·オーバーフロー（rtc_calovf）
tamp_itamp6	ST 製造者読出し

36.3.3 TAMP レジスタ書込み保護

システム·リセット後、電源制御ペリフェラルの DBP ビットによって、TAMP レジスタ（バックアップ·レジスタを含む）は不要な書込みアクセスから保護されます（PWR 電源制御のセクションを参照）。TAMP レジスタ書込みアクセスを可能にするには、DBP ビットをセットする必要があります。

36.3.4 タンパ検出

タンパ検出は次の目的で設定できます。

- バックアップレジスタ（デフォルト設定）を消去する。
- STOP および STANDBY モードからウェイクアップするための割込みを生成する。
- 低電力タイマのためのハードウェアトリガを生成する。

TAMP バックアップレジスタ

バックアップレジスタ（TAMP_BKPxR）は、システムリセットや STANDBY モードからのウェイクアップではリセットされません。

バックアップレジスタは、タンパ検出イベント発生時にリセットされますが、TAMPxNOER ビットがセットされている場合や TAMP_CR2 レジスタの TAMPxMSK がセットされている場合は例外です。

注： **バックアップレジスタは、Flash の読出し保護がレベル 1 からレベル 0 に変わったときにも消去されます。**

タンパ検出の初期化

各入力は、TAMP_CR レジスタにおいて該当する TAMPxE ビットを 1 にセットすることにより有効にできます。

各 TAMP_INx タンパ検出入力は、TAMP_SR レジスタのフラグ TAMPxF に関連付けられています。

TAMPxMSK がクリアされている場合：

TAMPxF フラグは、ピン上でタンパイベントが発生した後にアサートされます。その際の遅延時間を以下に示します。

- TAMPFLT が 0x0（フィルタありのレベル検出）ではない場合は 3 ck_apre サイクル
- TAMPTS = 1（タンパイベントのタイムスタンプ）の場合は 3 ck_apre サイクル
- TAMPFLT = 0x0（エッジ検出）および TAMPTS = 0 の場合は遅延なし

TAMPxF がセットされている場合に、この周期中に同一ピンで発生した新たなタンパイベントを検出することはできません。

TAMPxMSK がセットされている場合：

同一ピンで発生した新たなタンパイベントは、上述した遅延の間とさらに ck_rtc 2.5 サイクルの間は検出できません。

TAMP_IER レジスタの TAMPxIE ビットをセットすることにより、タンパ検出イベント発生時に割込みが生成されます（TAMPxF がセットされている場合）。該当する TAMPxMSK がセットされている場合、TAMPxIE をセットすることはできません。

タンパイベント時のトリガ出力生成

タンパイベント検出は、低電力タイマによるトリガ入力として使用できます。

TAMP_CR レジスタで TAMPxMSK ビットがセットされている場合、TAMPxF フラグはソフトウェアによってクリアされ、同じピンで新しいタンパ検出ができるようになります。

TAMPxMSK ビットがセットされている場合、TAMPxF フラグはマスクされ、TAMP_SR レジスタでクリアされたままとなります。この設定により、TAMPxF をクリアするためにシステムをウェイクアップする必要もなく、STOP モードのまま低電力タイマを自動的にトリガすることができます。この場合、バックアップレジスタはクリアされません。

この機能は、タンパが タンパ入力でのフィルタを使ったレベル検出（パッシブモード）モードで設定された場合にのみ利用可能です（TAMPFLT ≠ 00、およびアクティブモードが非選択の場合）。

タンパイベント時のタイムスタンプ

RTC_CR で TAMPTS を 1 にセットすると、すべてのタンパイベントがタイムスタンプを発生させるようになります。この場合、通常のタイムスタンプイベント発生時と同様に TSF ビットまたは TSOVF ビットが RTC_SR でセットされます。TSF または TSOVF が RTC_SR でセットされるのと同時に、影響を受けるタンパフラグレジスタ TAMPxF が TAMP_SR でセットされます。

タンパ入力でのエッジ検出（パッシブモード）

TAMPFLT ビットが 00 の場合、該当する TAMPxTRG ビットに応じて、立ち上がりエッジ／ハイレベルまたは立ち下がりエッジ／ローレベルが観測されると TAMP_INx ピンがタンパ検出イベントを生成します。エッジ検出を選択すると、TAMP_INx 入力の内部プルアップ抵抗が無効になります。

注意： **エッジ検出を使用する場合は、タンパ検出の有効化直後のタンパピンのレベルをソフトウェアでチェックすることをお勧めします（GPIO レジスタを読み出す）。また、タンパイベントの検出を有効化する前にアクティブエッジが発生していないかどうかを確認するため、影響を受ける値をバックアップレジスタに書き込む前にソフトウェアでチェックすることをお勧めします。**
TAMPFLT = 00 および TAMPxTRG = 0（立ち上がりエッジ検出）のとき、タンパ入力がタンパ検出を有効にする前にすでにハイレベルになっている場合、タンパイベントがハードウェアによって検出される場合があります。

タンパイベントが検出されクリアされた後に、バックアップレジスタ（TAMP_BKPxR）を再プログラムする場合には、事前に、TAMP_INx を無効にしてから再度有効にする（TAMPxE を 1 にセット）必要があります。これによって、TAMP_INx の入力値がタンパ検出を示している期間中に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、TAMP_INx の入力でのレベル検出に相当します。

注： **タンパ検出は、V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不必要なリセットを避けるには、TAMPx が設定されているピンを外部で適切な信号レベルに接続しておく必要があります。**

タンパ入力でのフィルタを使ったレベル検出（パッシブモード）

フィルタを使ったレベル検出は、TAMPFLT を 0 以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLT に応じて）2、4 または 8 回のいずれかの連続したサンプルが TAMPxTRG ビットで指定するレベルで観測されたときに生成されます。

TAMP_INx 入力は、TAMPPUDIS が 1 にセットされて無効な状態になっていない限り、その状態がサンプリングされる前に I/O の内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間は TAMPPRCH ビットによって決定され、TAMP_INx 入力により大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQ を使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： **プルアップ抵抗の電気的特性については、データシートを参照してください。**

36.4 TAMP の低電力モード

表 341. 低電力モードが TAMP に与える影響

モード	説明
SLEEP	影響はありません。 TAMP 割込みによって、デバイスは SLEEP モードから復帰します。
STOP	クロック･ソースが LSE または LSI のときに限り有効な状態を保つフィルタ･モードによるレベル検出の場合を除き、すべての機能に影響はありません。 タンパイベントによって、デバイスは STOP モードを終了します。
STANDBY	クロックソースが LSE または LSI のときに限り有効なフィルタモードによるレベル検出の場合を除き、すべての機能に影響はありません。タンパイベントよって、デバイスは STANDBY モードから復帰します。
SHUTDOWN	クロックソースが LSE のときに限り有効なフィルタモードによるレベル検出の場合を除き、すべての機能に影響はありません。タンパイベントによって、デバイスは SHUTDOWN モードを終了します。

36.5 TAMP 割込み

割込みチャネルは割込みステータス･レジスタで設定します。また、割込み出力も有効化されます。

表 342. 割込みリクエスト

項目（割込みの略称）	割込みイベント	イベントフラグ(1)	イネーブル制御ビット(2)	割込みのクリア方法	SLEEP モードの終了	STOP および STANDBY モードの終了	SHUTDOWN モードの終了
TAMP	タンパ x(3)	TAMPxF	TAMPxIE	CTAMPxF に 1 を書き込む	可能	はい(4)	可能(5)
	内部タンパ y (3)	ITAMPyF	ITAMPyIE	CITAMPxF に 1 を書き込む	可能	可能 (4)	あり (5)

1. イベントフラグは、TAMP_SR レジスタ内です。
2. 割込みマスク済みフラグ（イベントフラグおよびイネーブル制御ビットからの結果）は、TAMP_MISR レジスタ内です。
3. タンパおよび内部タンパイベントの数は製品によります。
4. フィルタ･パッシブ･タンパ･モードによるレベル検出の場合、STOP モードおよび STANDBY モードからのウェイクアップは、TAMP クロック･ソースが LSE または LSI のときのみ可能です。
5. フィルタ･パッシブ･タンパ･モードによるレベル検出の場合、SHUTDOWN モードからのウェイクアップは、TAMP クロック･ソースが LSE のときのみ可能です。

36.6 TAMP レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの72ページのセクション 1.2 を参照してください。ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスすることができます。

36.6.1 TAMP 制御レジスタ 1（TAMP_CR1）

アドレス·オフセット：0x00

バックアップ ドメイン·リセット値：0xFFFF 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6E	ITAMP5E	ITAMP4E	ITAMP3E	Res.	Res.
										rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3E	TAMP2E	TAMP1E
													rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6E**：内部タンパ 6 イネーブル：ST 製造者読出し

0：内部タンパ 6 は無効です。

1：内部タンパ 6 が有効です：ST 製造者による読出しに対して、タンパが生成されます。

ビット 20 **ITAMP5E**：内部タンパ 5 イネーブル：RTC カレンダオーバーフロー

0：内部タンパ 5 は無効です。

1：内部タンパ 5 は有効です。RTC カレンダが 99 年 12 月 31 日 23:59:59 に最大値に達するとタンパが生成されます。このとき、カレンダは停止し、オーバーフローしません。

ビット 19 **ITAMP4E**：内部タンパ 4 イネーブル：HSE 監視

0：内部タンパ 4 は無効です。

1：内部タンパ 4 は有効です。HSE 周波数が閾値の範囲外になるとタンパが生成されます。

ビット 18 **ITAMP3E**：内部タンパ 3 イネーブル：LSE 監視

0：内部タンパ 3 は無効です。

1：内部タンパ 3 は有効です。LSE 周波数が閾値の範囲外になるとタンパが生成されます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3E**：TAMP_IN3 のタンパ検出イネーブル (1)

0：TAMP_IN3 のタンパ検出は無効です。

1：TAMP_IN3 のタンパ検出は有効です。

ビット 1 **TAMP2E**：TAMP_IN2 のタンパ検出イネーブル (1)

0：TAMP_IN2 のタンパ検出は無効です。

1：TAMP_IN2 のタンパ検出は有効です。

ビット 0 **TAMP1E**：TAMP_IN1 のタンパ検出イネーブル(1)

0：TAMP_IN1 のタンパ検出は無効です。

1：TAMP_IN1 のタンパ検出は有効です。

1. タンパ検出モード（TAMP_FLTCR レジスタと、TAMP_CR2 レジスタの TAMPxTRG ビットで選択）は、タンパ検出を有効にする前に設定する必要があります。

36.6.2 TAMP 制御レジスタ 2（TAMP_CR2）

アドレス･オフセット：0x04

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	TAMP3 TRG	TAMP2 TRG	TAMP1 TRG	Res.	Res.	Res.	Res.	Res.	TAMP3 MSK	TAMP2 MSK	TAMP1 MSK
					rw	rw	rw						rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3 NOER	TAMP2 NOER	TAMP1 NOER
													rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **TAMP3TRG**：タンパ 3 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 3 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 3 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 3 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 3 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 25 **TAMP2TRG**：タンパ 2 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 2 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 2 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 2 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 2 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 24 **TAMP1TRG**：タンパ 1 の入力のアクティブレベル（アクティブモード無効）

0：TAMPFLT ≠ 00 の場合、タンパ 1 の入力がローのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 1 の入力が立ち上がりエッジおよびハイレベルのとき、タンパ検出イベントがトリガされます。

1：TAMPFLT ≠ 00 の場合、タンパ 1 の入力がハイのままのとき、タンパ検出イベントがトリガされます。TAMPFLT = 00 の場合、タンパ 1 の入力が立ち下がりエッジおよびローレベルのとき、タンパ検出イベントがトリガされます。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **TAMP3MSK**：タンパ 3 マスク

0:タンパ 3 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP3F をソフトウェアでクリアする必要があります。

1：タンパ 3 イベントによってトリガイベントが生成されます。TAMP3F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタは消去されません。

TAMP3MSK がセットされている場合、タンパ 3 割込みを有効にできません。

ビット 17 **TAMP2MSK**：タンパ 2 マスク

0:タンパ 2 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP2F をソフトウェアでクリアする必要があります。

1：タンパ 2 イベントによってトリガイベントが生成されます。TAMP2F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタは消去されません。

TAMP2MSK がセットされている場合、タンパ 2 割込みを有効にできません。

ビット 16 **TAMP1MSK**：タンパ 1 マスク

0:タンパ 1 イベントでトリガイベントが生成され、次のタンパイベント検出を許可するために TAMP1F をソフトウェアでクリアする必要があります。

1：タンパ 1 イベントによってトリガイベントが生成されます。TAMP1F はマスクされ、ハードウェアで内部的にクリアされます。バックアップレジスタは消去されません。

TAMP1MSK がセットされている場合、タンパ 1 割込みを有効にできません。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3NOER**：タンパ 3 消去なし

0：タンパ 3 イベントでバックアップレジスタは消去されます。

1：タンパ 3 イベントでバックアップ･レジスタは消去されません。

ビット 1 **TAMP2NOER**：タンパ 2 消去なし

0：タンパ 2 イベントでバックアップレジスタは消去されます。

1：タンパ 2 イベントでバックアップ･レジスタは消去されません。

ビット 0 **TAMP1NOER**：タンパ 1 消去なし

0：タンパ 1 イベントでバックアップレジスタは消去されます。

1：タンパ 1 イベントでバックアップ･レジスタは消去されません。

36.6.3 TAMP フィルタ制御レジスタ（TAMP_FLTCR）

アドレスオフセット：0x0C

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP PUDIS	TAMPPRCH [1:0]		TAMPFLT [1:0]		TAMPFREQ [2:0]		
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TAMPPUDIS**：TAMP_INx プルアップディスエーブル

このビットにより、毎回のサンプリング前に各 TAMPx ピンをプリチャージするかどうか決定します。

0：サンプリング前に TAMP_INx ピンをプリチャージします（内部プルアップを有効化）。

1：TAMP_INx のプリチャージを無効化します。

ビット 6:5 **TAMPPRCH[1:0]**：TAMP_INx プリチャージ持続時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCH は、各 TAMP_INx 入力に対して有効です。

0x0：1 RTCCLK サイクル

0x1：2 RTCCLK サイクル

0x2：4 RTCCLK サイクル

0x3：8 RTCCLK サイクル

ビット 4:3 **TAMPFLT[1:0]**：TAMP_INx フィルタカウント

これらのビットにより、タンパイベントをアクティブにするのに必要な指定のレベル（TAMP*TRG）での連続サンプリングの数を決定します。TAMPFLT は、各 TAMP_INx 入力に対して有効です。

0x0：TAMP_INx 入力がアクティブレベル（TAMP_INx 入力における内部プルアップなし）に変化するときのエッジで、タンパイベントがアクティブになります。

0x1：アクティブレベルでの連続した 2 回のサンプリングの後、タンパイベントがアクティブになります。

0x2：アクティブレベルでの連続した 4 回のサンプリングの後、タンパイベントがアクティブになります。

0x3：アクティブレベルでの連続した 8 回のサンプリングの後、タンパイベントがアクティブになります。

ビット 2:0 **TAMPFREQ[2:0]**：タンパサンプリング周波数

これらのビットにより、各 TAMP_INx 入力がサンプリングされる周波数を決定します。

0x0：RTCCLK / 32768（RTCCLK = 32768 Hz の場合 1 Hz）

0x1：RTCCLK / 16384（RTCCLK = 32768 Hz の場合 2 Hz）

0x2：RTCCLK / 8192（RTCCLK = 32768 Hz の場合 4 Hz）

0x3：RTCCLK / 4096（RTCCLK = 32768 Hz の場合 8 Hz）

0x4：RTCCLK / 2048（RTCCLK = 32768 Hz の場合 16 Hz）

0x5：RTCCLK / 1024（RTCCLK = 32768 Hz の場合 32 Hz）

0x6：RTCCLK / 512（RTCCLK = 32768 Hz の場合 64 Hz）

0x7：RTCCLK / 256（RTCCLK = 32768 Hz の場合 128 Hz）

注： このレジスタは、パッシブモードでのタンパ入力のみ関係します。

36.6.4 TAMP 割込み有効レジスタ（TAMP_IER）

アドレスオフセット：0x2C

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6IE	ITAMP5IE	ITAMP4IE	ITAMP3IE	Res.	Res.
										rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3IE	TAMP2IE	TAMP1IE
													rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6IE**：内部タンパ 6 割込みイネーブル：ST 製造者読出し

0：内部タンパ 6 割込みは無効です。

1：内部タンパ 6 割込みは有効です。

ビット 20 **ITAMP5IE**：内部タンパ 5 割込みイネーブル：RTC カレンダオーバーフロー
0：内部タンパ 5 割込みは無効です。
1：内部タンパ 5 割込みは有効です。

ビット 19 **ITAMP4IE**：内部タンパ 4 割込みイネーブル：HSE 監視
0：内部タンパ 4 割込みは無効です。
1：内部タンパ 4 割込みは有効です。

ビット 18 **ITAMP3IE**：内部タンパ 3 割込みイネーブル：LSE 監視
0：内部タンパ 3 割込みは無効です。
1：内部タンパ 3 割込みは有効です。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3IE**：タンパ 3 割込みイネーブル
0：タンパ 3 割込みは無効です。
1：タンパ 3 割込みは有効です。

ビット 1 **TAMP2IE**：タンパ 2 割込みイネーブル
0：タンパ 2 割込みは無効です。
1：タンパ 2 割込みは有効です。

ビット 0 **TAMP1IE**：タンパ 1 割込みイネーブル
0：タンパ 1 割込みは無効です。
1：タンパ 1 割込みは有効です。

36.6.5 TAMP ステータスレジスタ （TAMP_SR）

アドレスオフセット：0x30

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6F	ITAMP5F	ITAMP4F	ITAMP3F	Res.	Res.
										r	r	r	r		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3F	TAMP2F	TAMP1F
													r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6F**：ST 製造者による読出しタンパ検出フラグ
このフラグは、内部タンパ 6 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 20 **ITAMP5F**：RTC カレンダオーバーフロータンパ検出フラグ
このフラグは、内部タンパ 5 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 19 **ITAMP4F**：HSE 監視タンパ検出フラグ

このフラグは、内部タンパ 4 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 18 **ITAMP3F**：LSE 監視タンパ検出フラグ

このフラグは、内部タンパ 3 にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3F**：TAMP3 検出フラグ

このフラグは、TAMP3 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 1 **TAMP2F**：TAMP2 検出フラグ

このフラグは、TAMP2 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

ビット 0 **TAMP1F**：TAMP1 検出フラグ

このフラグは、TAMP1 入力にタンパ検出イベントが検出されたときに、ハードウェアによってセットされます。

36.6.6 TAMP マスク済み割込みステータスレジスタ（TAMP_MISR）

アドレスオフセット：0x34

バックアップ ドメイン･リセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6MF	ITAMP5MF	ITAMP4MF	ITAMP3MF	Res.	Res.
										r	r	r	r		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP2MF	TAMP1MF
													r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **ITAMP6MF**：ST 製造者による読出しタンパの割込みマスク済みフラグ

このフラグは、内部タンパ 6 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 20 **ITAMP5MF**：RTC カレンダオーバーフロータンパの割込みマスク済みフラグ

このフラグは、内部タンパ 5 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 19 **ITAMP4MF**：HSE 監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 4 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 18 **ITAMP3MF**：LSE 監視タンパの割込みマスク済みフラグ

このフラグは、内部タンパ 3 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **TAMP3MF**：TAMP3 割込みマスク済みフラグ
このフラグは、タンパ 3 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 1 **TAMP2MF**：TAMP2 割込みマスク済みフラグ
このフラグは、タンパ 2 に割込みが行われたときに、ハードウェアによってセットされます。

ビット 0 **TAMP1MF**：TAMP1 割込みマスク済みフラグ
このフラグは、タンパ 1 に割込みが行われたときに、ハードウェアによってセットされます。

36.6.7 TAMP ステータスクリアレジスタ（TAMP_SCR）

アドレスオフセット：0x3C

システム･リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITAMP6F	CITAMP5F	CITAMP4F	CITAMP3F	Res.	Res.
										w	w	w	w		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP3F	CTAMP2F	CTAMP1F
													w	w	w

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CITAMP6F**：ITAMP6 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP6F ビットがクリアされます。

ビット 20 **CITAMP5F**：ITAMP5 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP5F ビットがクリアされます。

ビット 19 **CITAMP4F**：ITAMP4 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP4F ビットがクリアされます。

ビット 18 **CITAMP3F**：ITAMP3 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの ITAMP3F ビットがクリアされます。

ビット 17 予約済みであり、リセット値に保持する必要があります。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CTAMP3F**：TAMP3 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP3F ビットがクリアされます。

ビット 1 **CTAMP2F**：TAMP2 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP2F ビットがクリアされます。

ビット 0 **CTAMP1F**：TAMP1 検出フラグのクリア
このビットに 1 を書き込むと、TAMP_SR レジスタの TAMP1F ビットがクリアされます。

36.6.8 TAMP バックアップ x レジスタ（TAMP_BKPxR）

アドレスオフセット：0x100 + 0x04 * x、（x = 0 から 31）

バックアップドメインリセット値：0x0000 0000

システムリセット：影響なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKP[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BKP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw

ビット 31:0 **BKP[31:0]**

アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。

これらのレジスタは、V_{DD} がオフになった場合、V_{BAT} によって電源が供給されるため、システムリセットによりリセットされず、デバイスが低電力モードで動作する場合、レジスタの内容は有効なまま保持されます。

デフォルト設定で、このレジスタはタンパ検出イベントでリセットされます。少なくとも 1 つの内部または外部タンパイベントフラグがセットされると、値をリセットするよう強制されます。このレジスタは、読出し保護（RDP）が無効である場合もリセットされます。

36.6.9 TAMP レジスタマップ

表 343. TAMP レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	TAMP_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6E	ITAMP5E	ITAMP4E	ITAMP3E	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3E	TAMP2E	TAMP1E
	リセット値											1	1	1	1																0	0	0
0x04	TAMP_CR2	Res.	Res.	Res.	Res.	Res.	TAMP3TRG	TAMP2TRG	TAMP1TRG	Res.	Res.	Res.	Res.	Res.	TAMP3MSK	TAMP2MSK	TAMP1MSK	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3NOER	TAMP2NOER	TAMP1NOER
	リセット値						0	0	0						0	0	0													0	0	0
0x0C	TAMP_FLTCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]		
	リセット値																									0	0	0	0	0	0	0	0
0x2C	TAMP_IER	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6IE	ITAMP5IE	ITAMP4IE	ITAMP3IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3IE	TAMP2IE	TAMP1IE
	リセット値											0	0	0	0																0	0	0
0x30	TAMP_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6F	ITAMP5F	ITAMP4F	ITAMP3F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3F	TAMP2F	TAMP1F
	リセット値											0	0	0	0																0	0	0
0x34	TAMP_MISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ITAMP6MF	ITAMP5MF	ITAMP4MF	ITAMP3MF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TAMP3MF	TAMP2MF	TAMP1MF
	リセット値											0	0	0	0																0	0	0
0x3C	TAMP_SCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CITAMP6F	CITAMP5F	CITAMP4F	CITAMP3F	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTAMP3F	CTAMP2F	CTAMP1F
	リセット値											0	0	0	0																0	0	0
0x100 + 0x04*x、（x = 0～31）	TAMP_BKPxR	BKP[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

37 USART/UART（Universal synchronous / asynchronous receiver transmitter）

このセクションでは、USART（Universal synchronous asynchronous receiver transmitter）について説明します。

37.1 USART の概要

USART を使用すると、業界標準の NRZ 非同期シリアルデータフォーマットを必要とする外部機器と柔軟性の高い全二重データ交換を行うことができます。フラクショナルボーレートジェネレータによって、非常に広範囲のボーレートが達成できます。

USART は同期式単方向通信と半二重単線通信の両方のほか、LIN（Local Interconnection Network）、スマートカードプロトコル、IrDA（infrared data association）SIR ENDEC 仕様、およびモデム動作（CTS/RTS）もサポートします。マルチプロセッサ通信もサポートされています。

マルチバッファ設定で DMA（直接メモリアクセス）を使用することによって、ハイスピードデータ通信が可能です。

37.2 USART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- 16 倍または 8 倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の最適な調整を達成
- ボーレートジェネレータシステム
- データ送信および受信用の 2 つの内部 FIFO
  各 FIFO はソフトウェアで有効／無効にすることができ、ステータスフラグを装備
- 共通にプログラム可能な送信および受信ボーレート
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- 自動ボーレート検出
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 同期通信のための同期マスタ／スレーブモードとクロック出力／入力
- SPI スレーブ送信アンダーランエラーフラグ
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 通信制御／エラー検出フラグ
- パリティ制御：
  – パリティビットの送信
  – 受信したデータバイトのパリティ検査
- フラグ付き割込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ
- STOP モードからのウェイクアップ

37.3 USART の拡張機能

- LIN マスタの同期ブレーク送信機能と LIN スレーブのブレーク検出機能
  - USART が LIN 用にハードウェア設定されている場合、13 ビットのブレーク生成と 10/11 ビットのブレーク検出
- 通常モードで 3/16 ビット期間をサポートする IrDA SIR エンコーダデコーダ
- スマートカードモード
  - ISO/IEC 7816-3 規格で定義されているスマートカードの T = 0 および T = 1 非同期プロトコルをサポート
  - スマートカード動作用に 0.5 個および 1.5 個のストップ･ビット
- Modbus 通信のサポート
  - タイムアウト機能
  - CR/LF キャラクタ認識

37.4 USART の実装

下表に、STM32G4 シリーズ デバイスの USART 実装の機能を示します。比較のため、LPUART の機能も示します。

表 344. USART/LPUART の機能

USART/LPUART のモード／機能(1)	USART1/2/3	UART4/5	LPUART1
モデムのハードウェア･フロー制御	X	X	X
DMA を使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDEC ブロック	X	X	-
LIN モード	X	X	-
デュアルクロックドメインと低電力モードからのウェイクアップ	X	X	X
レシーバタイムアウト割込み	X	X	-
Modbus 通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USART データ長	7、8、および 9 ビット		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFO サイズ	8		

1. X：サポートされています。

37.5 USART の機能詳細

37.5.1 USART ブロック図

図 531. USART ブロック図

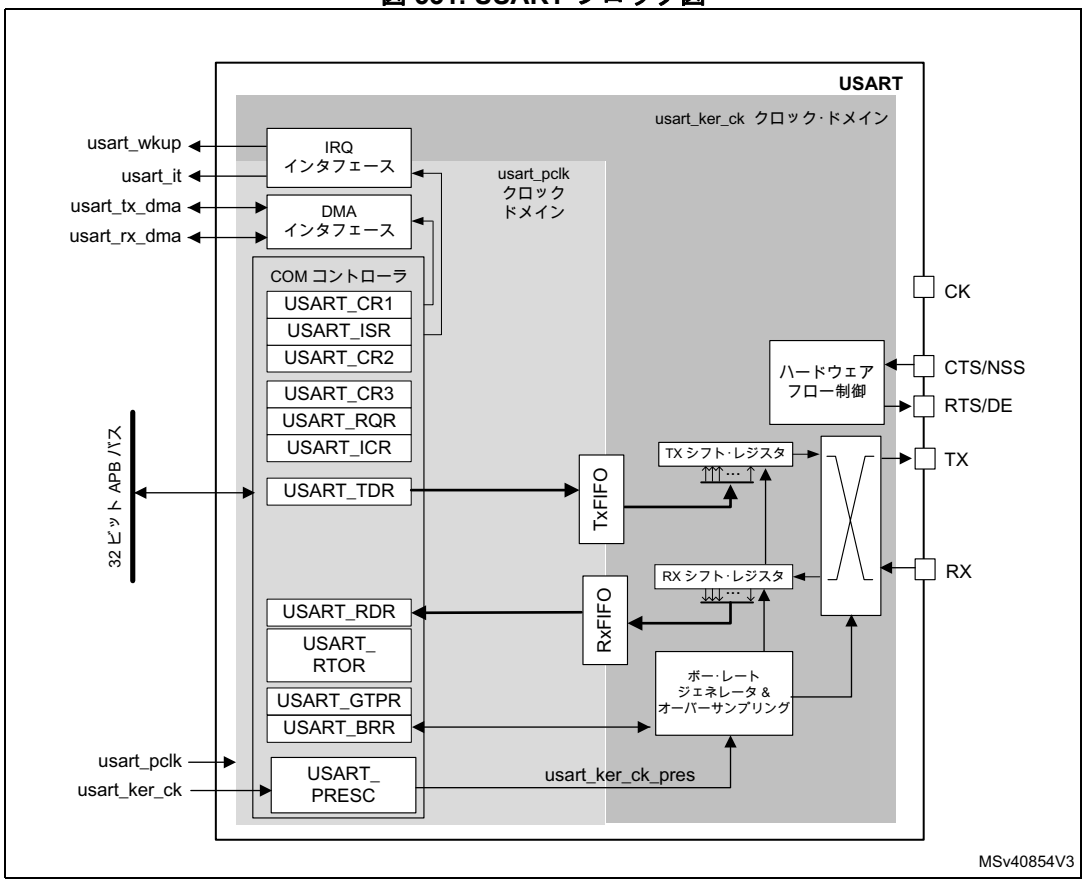


図 531 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロック·ドメインが示されています。

- **usart_pclk** クロックドメイン

  Usart_pclkクロック信号はペリフェラルバスインタフェースにクロック供給します。USART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **usart_ker_ck** カーネルクロックドメイン

  usart_ker_ck は USART クロックソースです。**usart_pclk** とは独立で、RCC によって供給されます。したがって、**usart_ker_ck** クロックが停止しているときでも、USART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**usart_ker_ck** クロックは **usart_pclk** クロックと同じになります。

usart_pclk と **usart_ker_ck** の間には何の制約もありません。**usart_ker_ck** は **usart_pclk** より速くすることも遅くすることもできます。唯一の限界は、十分に速く通信を管理できるソフトウェアの能力です。

USART が SPI スレーブモードで動作するときは、USART は、外部マスタ SPI デバイスによって提供される外部 CK 信号から導き出されるシリアルインタフェースクロックを使用してデータフロー

を処理します。**usart_ker_ck** クロックは CK 入力のクロックより少なくとも 3 倍は速くなければなりません。

37.5.2 USART 信号

USART 双方向通信

USART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）

  RX はシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用されています。それによって、有効な入力データとノイズを区別しています。

- **TX**（送信データ出力）

  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、データを送信する必要がないとき、TX ピンはハイになります。単線およびスマートカードモードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）

  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。

- **RTS**（Request To Send）

  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェア制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（Driver Enable）

  この信号は、外部トランシーバの送信モードを有効にします。

注： **DE と RTS は同じピンを共有します。**

同期マスタ／スレーブモードおよびスマートカードモード

同期マスタ／スレーブモードおよびスマートカードモードでは、以下のピンが必要です。

- CK

  このピンは同期マスタ･モードおよびスマートカード･モードで、クロック出力として機能します。

  同期スレーブモードでは、クロック入力として機能します。

  同期マスタモードでは、このピンは、SPI マスタモードに対応する同期送信用のトランスミッタデータクロックを出力します（スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します）。これと並行して、RX ピンでデータを同期受信できます。このメカニズムを使用して、シフトレジスタを持つペリフェラル（例：LCD ドライバ）を制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。

  スマートカードモードでは、CK 出力はスマートカードにクロックを供給します。

- NSS

  このピンは、同期スレーブ･モードでスレーブ選択入力として機能します。

注： **NSS と CTS は同じピンを共有します。**

37.5.3 USART キャラクタの説明

ワード長は、USART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 ビットに設定できます（図 532 を参照）。

- 7 ビットのキャラクタ長：M[1:0] = “10”
- 8 ビットのキャラクタ長：M[1:0] = “00”
- 9 ビットのキャラクタ長：M[1:0] = “01”

注： **7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート（0x7F および 0x55 フレーム検出）はサポートされません。**

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

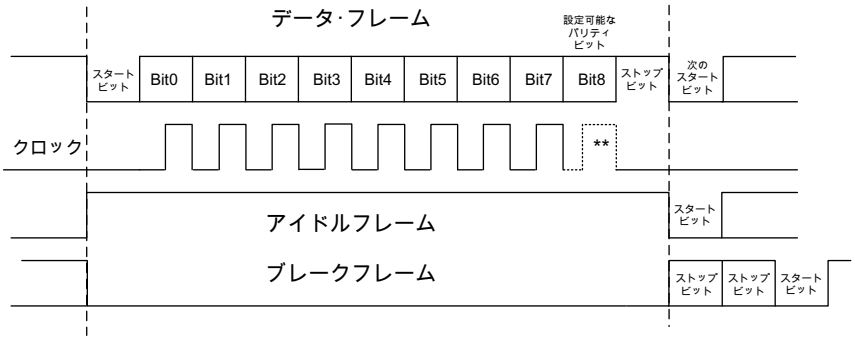
ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

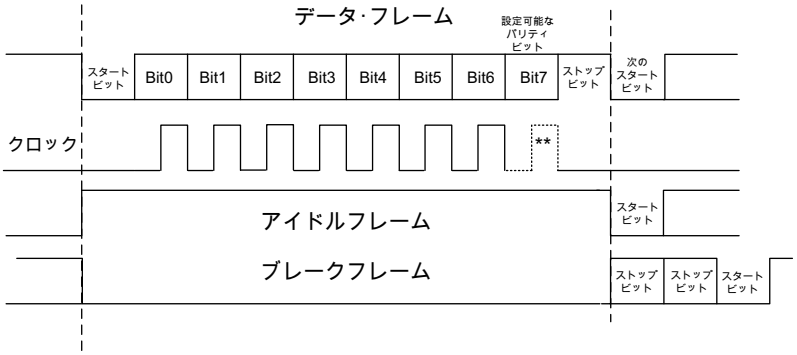
各ブロックの詳細を次に示します。

図 532. ワード長のプログラミング

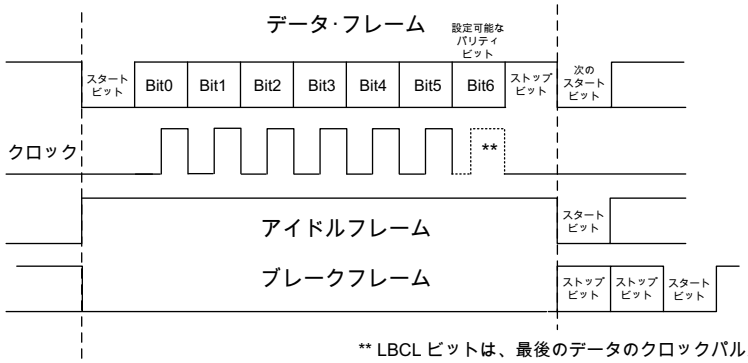
9 ビットワード長（M = 01）、1 個のストップビット



8 ビットワード長（M = 00）、1 個のストップビット



7 ビットワード長（M = 10）、1 個のストップビット



** LBCL ビットは、最後のデータのクロックパルスを制御します。

MS33194V2

37.5.4 USART の FIFO と閾値

USART は FIFO モードで動作できます。

USART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、USART_CR1 レジスタの FIFOEN（ビット 29）をセットすることによって有効になります。このモードは、UART、SPI、およびスマートカードモードでのみ、サポートされます。

最大のデータワード長が 9 ビットなので、TXFIFO は 9 ビット幅です。しかし、RXFIFO のデフォルト幅は 12 ビットです。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。**

ステータスフラグは、USART_ISR レジスタで入手可能です。

Tx および RX 割込みがトリガされる TXFIFO および RXFIFO のレベルを設定することができます。これらの閾値は、USART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、USART_ISR レジスタの RXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、RXFIFO 内のデータ数が、プログラムされた閾値と等しくなるまで RXFIFO が満たされることを意味します。

  RXFTCFG 分のデータが受信されると、1 データが USART_RDR にあり、（RXFTCFG - 1）データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された（FIFO サイズ -1 のデータが RXFIFO に入り、1 データが USART_RDR に入った）時に、RXFT フラグがセットされます。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、USART_ISR レジスタの TXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、TXFIFO の空き場所の数が、プログラムされた閾値と等しくなるまで TXFIFO が空にされることを意味します。

37.5.5 USART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは TX ピンで出力され、対応するクロックパルスは CK ピンで出力されます。

キャラクタ送信

USART 送信時、データは LSB ファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、USART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます。

FIFO モードが有効なとき、送信データレジスタ（USART_TDR）に書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 ビット周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 0.5、1、1.5、または 2 に設定できます。

注： **送信データを USART_TDR に書き込む前に、TE ビットをセットする必要があります。**

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

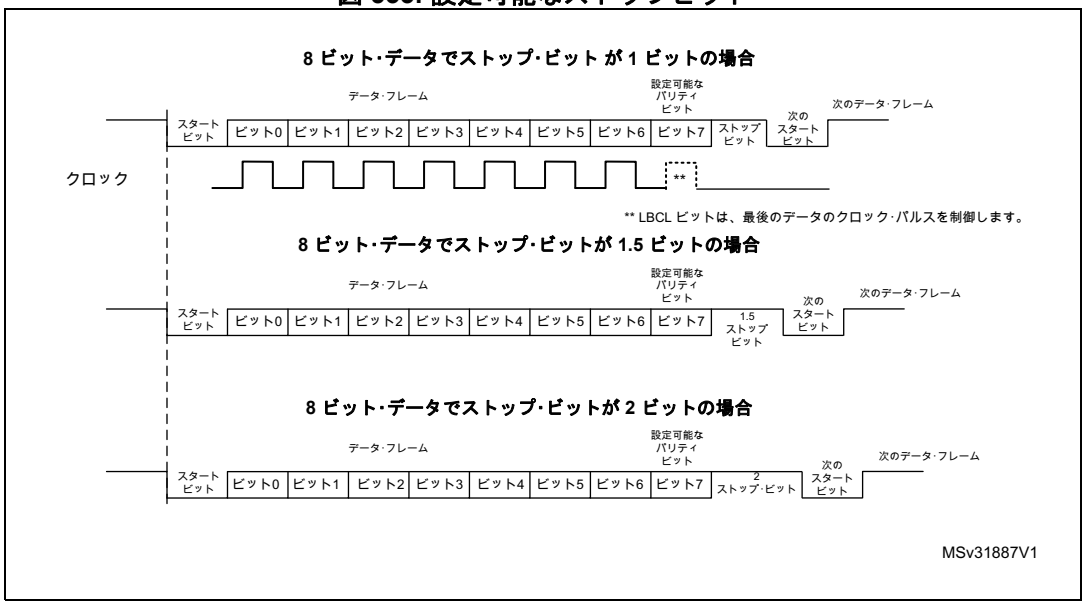
各キャラクタとともに送信されるストップビットの数は、USART_CR2 のビット 13、12 でプログラミングできます。

- **1 個のストップビット**：ストップビット数のデフォルト値です。
- **2 個のストップビット**：通常の USART モード、単線モード、およびモデムモードでサポートされます。
- **1.5 個のストップビット**：スマートカードモードで使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信では、M[1:0] = “00”の場合は 10 個のロー･ビットの後、M[1:0] = “01”の場合は 11個のロー･ビットの後、M[1:0] = “10”の場合は 9個のロー･ビットの後に 2個のストップ･ビットが続きます（図 533 を参照）。長いブレーク（9個、10個、または 11 個のロー･ビットを超える長さのブレーク）を送信することはできません。

図 533. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. USART_BRR レジスタを使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAT）を選択します。セクション 37.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを USART_TDR レジスタに書き込みます。シングルバッファの場合、送信される各データにこれを繰り返します。
   - FIFO モードが無効になっている場合は、USART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。
   - FIFO モードが有効になっている場合は、USART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、USART_TDR への書込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。
8. USART_TDR レジスタに最後のデータを書き込んだら、TC = 1 になるまで待ちます。
   - FIFO モードが無効になっている場合、これは最後のフレームの送信が完了したことを示します。
   - FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

   このチェックは、USART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされます。同フラグは以下を示します。
  - データは USART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  - USART_TDR レジスタは空です。
  - 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、このフラグは割込みを生成します。

  送信が行われているとき、USART_TDR レジスタへの書込み命令によってデータが TDR バッファに格納されます。そして、データは現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、USART_TDR レジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。
- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  - TXFIFO はフルではありません。
  - USART_TDR レジスタは空です。

– 次のデータを、前のデータに上書きせずに、USART_TDR レジスタに書き込めます。送信が行われているとき、USART_TDR レジスタへの書込み命令によってデータが TXFIFO に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

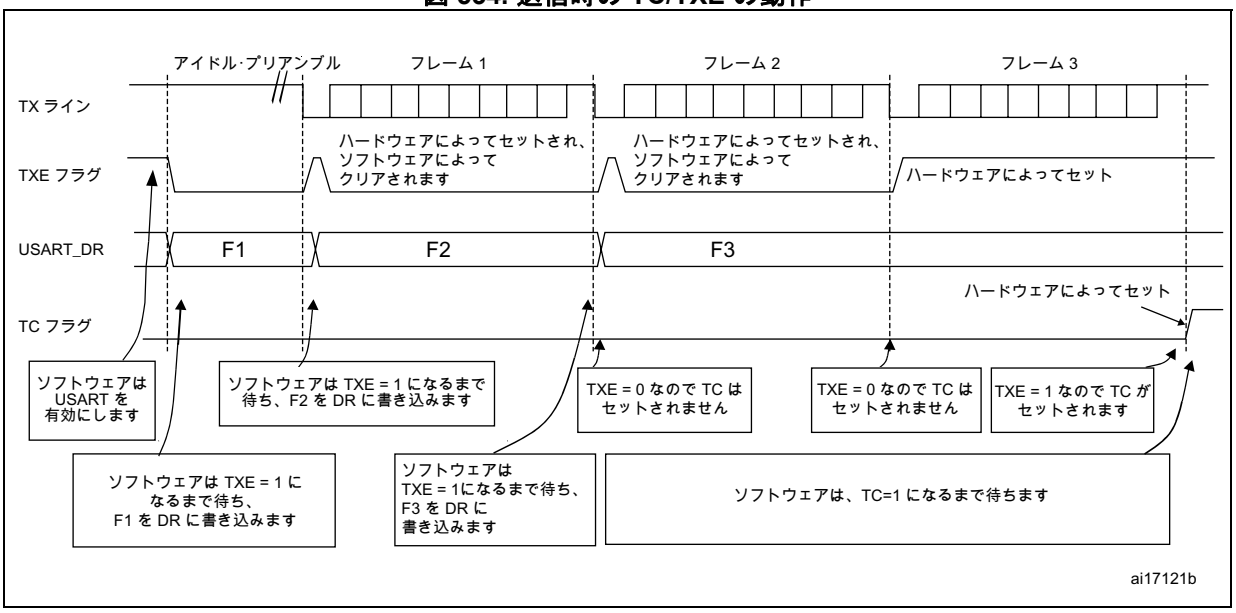
TXFIFO がフルではない場合、USART_TDR レジスタへの書込み動作が終わった後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNFIE ビットがセットされている場合、このフラグは割込みを生成します。

あるいは、TXFIFO 閾値に達した時、割込みが生成され、データを FIFO に書き込むことができます。この場合、CPU は、プログラムされたトリガレベルによって定義されたデータのブロックを書き込むことができます。

フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC フラグはハイレベルになります。USART_CR1 レジスタの TCIE ビットがセットされると、割込みが生成されます。

USART_TDR レジスタに最後のデータを書き込んだ後は、USART の無効化やデバイスの低電力モード移行の前に TC がセットされるまで待つ必要があります（図 534:送信時の TC/TXE の動作を参照）。

図 534. 送信時の TC/TXE の動作



注： **FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（図 532 を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。USART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（ストップ）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、USART は、最初のデータフレームの前にアイドルフレームを送信します。

37.5.6 USART レシーバ

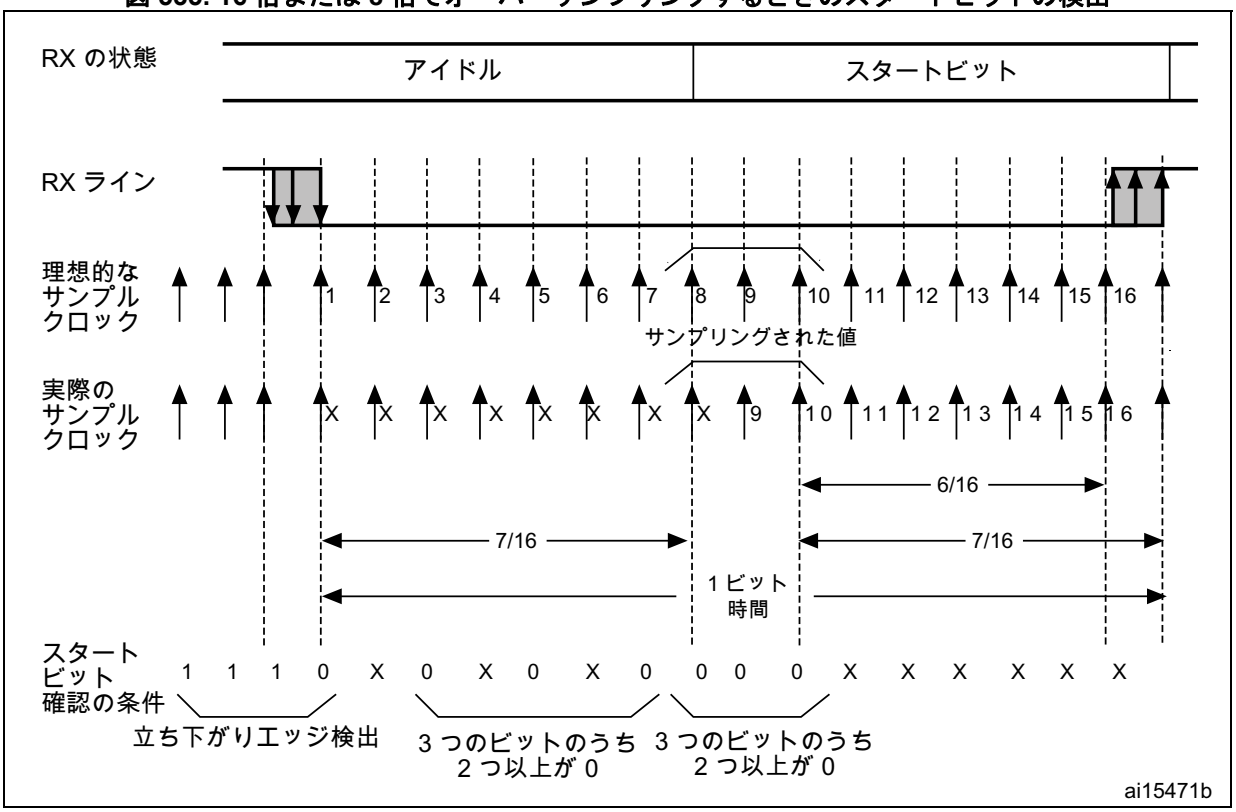
USART は、USART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

スタートビット検出シーケンスは、16 倍または 8 倍でオーバーサンプリングするときと同じです。

USART では、サンプルの特定シーケンスが認識されると、スタートビットが検出されます。このシーケンスは、1 1 1 0 X 0 X 0X 0X 0 X 0X 0 です。

図 535. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出



注： **シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。**

サンプリングされた 3 つのビットが“0”の場合（第 3、第 5、および第 7 ビットでの最初のサンプリングで 3 つのビットが“0”であり、第 8、第 9、および第 10 ビットでの 2 回目のサンプリングでも 3 つのビットが“0”）、スタート･ビットが確認されます（RXNE フラグがセットされ、RXNEIE = “1”の場合は割込みが生成されるか、または FIFO モードが有効な場合は、RXFNE フラグがセットされ、RXFNEIE = “1”の場合は割込みが生成されます）。

次の場合、スタート･ビットは確認されますが、NE ノイズ･フラグがセットされます。

a) 両方のサンプリングについて、サンプリングされた 3 ビットのうちの 2 ビットが“0”の場合（第 3、第 5、および第 7 ビットのサンプリングと、第 8、第 9、および第 10 ビットのサンプリング）。

または

b) いずれか一方のサンプリングで（第 3、第 5、および第 7 ビットでのサンプリングまたは第 8、第 9、および第 10 ビットでのサンプリング）、3 つのビットのうち 2 つが“0”の場合。

上記のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

キャラクタの受信

USART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルトの設定）でシフトアウトされます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. USART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ USART_BRR を使用して、希望するボーレートを選択します。
3. USART_CR2 レジスタでストップビットの数をプログラミングします。
4. USART_CR1 レジスタの UE ビットに“1”を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA 有効（DMAR）を選択します。セクション 37.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. USART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効の場合、RXNE ビットは、シフトレジスタの内容が RDR レジスタに転送されたことを示すためにセットされます。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示すために RXFNE ビットがセットされます。USART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードが有効な場合は RXFNEIE）ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズ、パリティ、またはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効な場合は、RXNE フラグはバイトを受信するたびにセットされます。DMA が受信データレジスタを読み出すと、同フラグはクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる USART_RDR レジスタからの読出しによって行われます。RXNE フラグは、USART_RQR レジスタの RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。オーバー

ランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE フラグをクリアする必要があります。

– FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE がセットされます。USART_RDR からの読出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、USART_RQR の RXFRQ ビットに“1”をプログラムすることによってクリアすることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割込みを生成します。あるいは、RXFIFO 閾値に達した時、割込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていない状態でキャラクタを受信すると、オーバーラン·エラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。前回のデータは USART_RDR レジスタを読み出すことによって、入手可能です。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

- FIFO モードが有効な場合

  シフト·レジスタを転送する準備ができていて、受信 FIFO がフルのとき、オーバーラン·エラーが発生します。

  RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから USART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

  RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RXFIFO の最初のエントリは失われません。それは、USART_RDR レジスタを読み出すことによって入手可能です。

– シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
– RXFNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

ORE ビットは、USART_ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： **ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。**

FIFO モードが無効になっているとき、2 つの可能性があります。

- **RXNE = 1 の場合、有効な最後のデータは受信レジスタ（RDR）に格納されていて、読出しが可能です。**
- **RXNE= 0 の場合、最後の有効なデータは読出し済みなので、RDR レジスタには読み出すデータが残っていません。有効な最後のデータが RDR レジスタで読み出されると同時に新しいデータが受信されて失われると、この状況が発生します。**

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（**リセットおよびクロック制御（RCC）**のセクションを参照）。クロックソースは、UE ビットのセットによって USART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

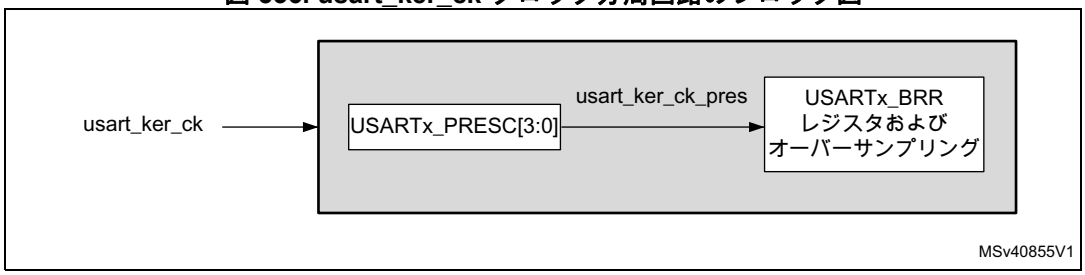
- USART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、usart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、usart_ker_ck クロックソースは RCC で設定できます（**リセットおよびクロック制御（RCC）**のセクションを参照）。そうしない場合、usart_ker_ck クロックは usart_pclk と同じになります。

usart_ker_ck クロックは、USART_PRESC レジスタで定義されたプログラム可能な係数によって分周できます。

図 536. usart_ker_ck クロック分周回路のブロック図



usart_ker_ck ソースによっては、USART は MCU が低電力モードのときにデータを受信することができます。受信データと選択されたウェイクアップモードに応じて、USART より必要なときに MCU をウェイクアップし、ソフトウェアによる USART_RDR レジスタの読出しによって、または DMA によって受信データを転送できます。

他のクロックソースの場合、USART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています（同期モードの場合を除く）。これにより、最大通信速度とノイズ／クロック精度の耐性との間の最適なトレードオフが図られます。

オーバーサンプリング方式は、USART_CR1 レジスタの OVER8 ビットをボーレートクロックの 16 倍または 8 倍にプログラムすることによって選択できます（図 537 および 図 538 を参照）。

アプリケーションに応じて、

- より高速な通信（最大 usart_ker_ck_pres/8）を実現するには 8 倍（OVER8 = 1）のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます（1581 ページのセクション 37.5.8 : クロック偏差に対する USART レシーバの許容誤差を参照）。
- クロック偏差に対するレシーバの許容誤差を大きくするには、16 倍（OVER8 = “0”）のオーバーサンプリングを選択します。この場合、最大速度は usart_ker_ck_pres/16 に制限されます（この usart_ker_ck_pres は、USART 入力クロックをプリスケーラで分周したクロックです）。

論理レベルの評価方法を選択するには、USART_CR3 レジスタの ONEBIT ビットをプログラミングします。次の 2 つのオプションが利用できます。

- 受信されたビットの中央にある 3 つのサンプルの多数決。この場合、多数決に使用された 3 つのサンプルが等しくないとき、NE ビットがセットされます。
- 受信されたビットの中央にある 1 つのサンプル。

  アプリケーションに応じて、

  – ノイズの多い環境での動作では 3 つのサンプルの多数決方式（ONEBIT = 0）を選択します。そしてノイズが検出された場合（図 345 を参照）は、サンプリング中にグリッチが発生していることになるので、該当のデータを除去します。

  – ラインにノイズがない場合、1 つのサンプルによる方式（ONEBIT = 1）を選択し、クロック偏差に対するレシーバの許容誤差を大きくします（1581 ページのセクション 37.5.8 : クロック偏差に対する USART レシーバの許容誤差を参照）。この場合、NE ビットはセットされません。

フレーム内でノイズが検出された場合：

- RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）の立ち上がりエッジで、NE ビットがセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3レジスタの EIE ビットがセットされている場合に割込みが発行されます。

NE ビットは、USART_ICR レジスタの NECF ビットをセットすることによってリセットされます。

注： **ノイズエラーは SPI モードではサポートされていません。**

8 倍のオーバーサンプリングは、スマートカード、IrDA、および LIN の各モードでは使用できません。これらのモードでは、OVER8 ビットはハードウェアによって“0”に固定されています。

図 537. データサンプリング（16 倍のオーバーサンプリング）

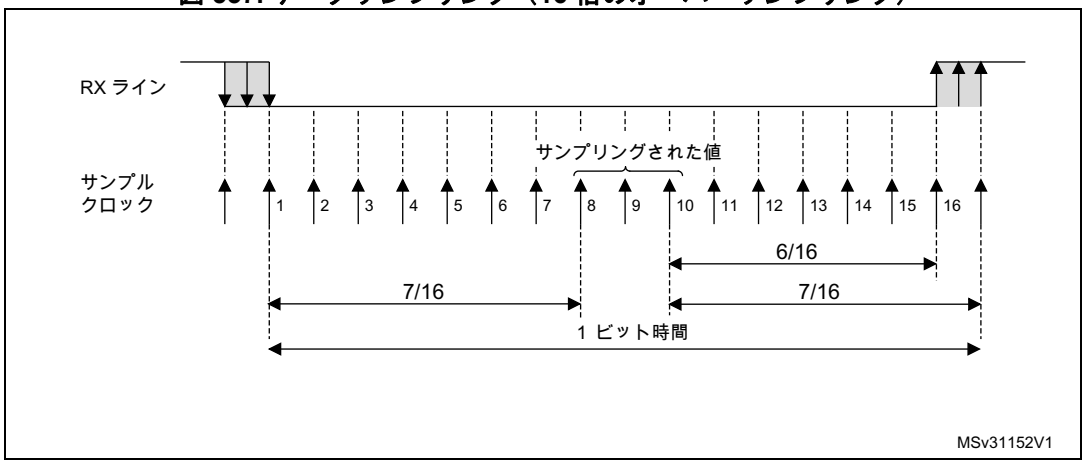


図 538. データサンプリング（8 倍のオーバーサンプリング）

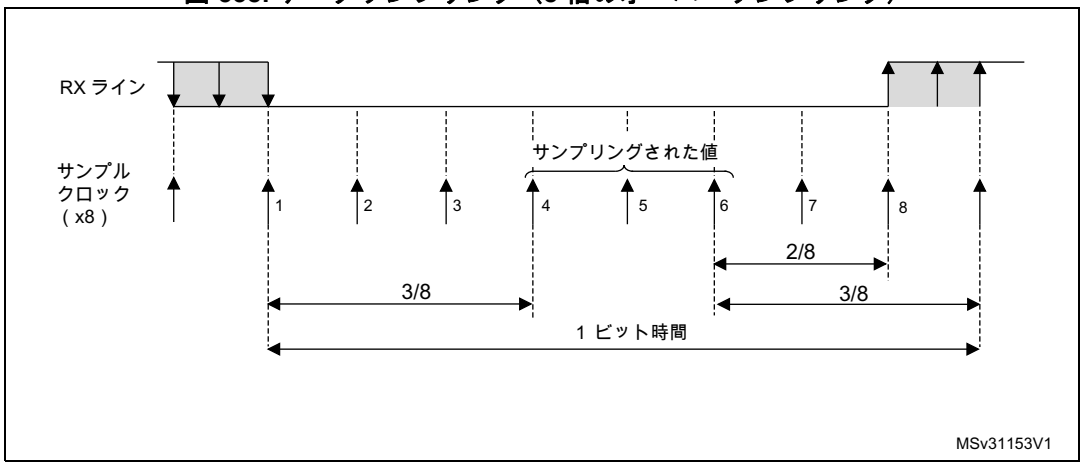


表 345. サンプリングされたデータからのノイズ検出

サンプリングされた値	NE ステータス	受信ビットの値
000	0	0
001	1	0
010	1	0
011	1	1
100	1	0
101	1	1
110	1	1
111	0	1

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USART_RDR レジスタ（FIFO モードが有効な場合は RXFIFO）へ転送されます。
- 1 バイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビット（FIFO モードが有効な場合は RXFNE ビット）と同時に立ち上がります。マルチバッファ通信の場合、USART_CR3 レジスタの EIE ビットがセットされている場合に割込みが発行されます。

USART_ICR レジスタの FECF に“1”を書き込むことによって、FE ビットがリセットされます。

注： **フレーミングエラーは SPI モードではサポートされていません。**

受信時の設定可能なストップビット

受信するストップビット数は、USART_CR の制御ビットを通じて設定でき、通常モードでは 1 または 2、スマートカードモードでは 0.5 または 1.5 にできます。

- **0.5 個のストップビット（スマートカードモードでの受信）**：0.5 個のストップビットでは、サンプリングは行われません。したがって、0.5 個のストップビットが選択されている場合、フレーミングエラーやブレークフレームは検出されません。
- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- 1.5 個のストップビット（スマートカードモード）

  スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバ･ブロックを有効にし（USART_CR1 の RE = 1）、ストップ･ビットをチェックして、スマートカードがパリティ･エラーを検出したかどうかをテストする必要があります。

  パリティ･エラーが発生している場合、スマートカードはサンプリング時のデータ信号を強制的にロー･レベルにします（これは、フレーミング･エラーとしてフラグ指定される NACK 信号です）。その後、1.5 個のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）を通して FE フラグがセットされます。ストップビット 1.5 個のサンプリングは、16 番目、17 番目、および 18 番目のサンプルで（ストップビットの開始から1 ボークロック周期後に）行われます。1.5 個のストップビットは、2 つの部分に分解できます。すなわち、何も起こらない 0.5 ボークロック周期と、途中でサンプリングが行われる通常の 1 ストップビット周期です（詳細は 1593 ページのセクション 37.5.16：USART レシーバタイムアウトを参照）。
- 2 個のストップビット

  ストップビット 2 個のサンプリングは、最初のストップビットの 8 番目、9 番目、および 10 番目のサンプルで行われます。

  最初のストップビット中にフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。

  2 番目のストップビットでは、フレーミングエラーの検査は行われません。最初のストップビットの最後に、RXNE フラグ（FIFO モードが有効な場合は RXFNE フラグ）がセットされます。

37.5.7 USART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、USART_BRR レジスタでプログラムされた値に設定されます。

式 1：標準 USART のボー･レート（SPI モードを含む）（OVER8 = “0”または“1”）

16 倍のオーバーサンプリングの場合、ボー･レートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

8 倍のオーバーサンプリングの場合、ボーレートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{2 \times \text{usart_ker_ckpres}}{\text{USARTDIV}}$$

式 2：スマートカード、LIN、および IrDA モードのボー･レート（OVER8 = 0）

ボー･レートは次の計算式で与えられます。

$$\text{Tx/Rx ボー} = \frac{\text{usart_ker_ckpres}}{\text{USARTDIV}}$$

USARTDIV は、符号なしの固定小数点数であり、USART_BRR レジスタでコード化されます。

- OVER8 = 0 のとき、BRR = USARTDIV です。
- OVER8 = 1 のとき、
  - BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。
  - BRR[3] は、クリアされたままにする必要があります。
  - BRR[15:4] = USARTDIV[15:4]

注： **ボーカウンタは、USART_BRR への書込み後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。**

16 倍および 8 倍のオーバーサンプリングの場合、USARTDIV は 16 以上である必要があります。

USART_BRR レジスタの値から USARTDIV を得る方法

例 1

usart_ker_ck_pres = 8 MHz で 9600 ボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 8 000 000/9600
  BRR = USARTDIV = 0d833 = 0x0341
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 8 000 000/9600
  USARTDIV = 1666,66 (0d1667 = 0x683)
  BRR[3:0] = 0x3 >> 1 = 0x1
  BRR = 0x681

例 2

usart_ker_ck_pres = 48 MHz で 921.6 キロ･ボーを得るには、

- 16 倍のオーバーサンプリングの場合：
  USARTDIV = 48 000 000/921 600
  BRR = USARTDIV = 0d52 = 0x34
- 8 倍のオーバーサンプリングの場合：
  USARTDIV = 2 * 48 000 000/921 600
  USARTDIV = 104 (0d104 = 0x68)
  BRR[3:0] = USARTDIV[3:0] >> 1 = 0x8 >> 1 = 0x4
  BRR = 0x64

37.5.8 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバは、クロックシステムの合計偏差が USART レシーバの許容誤差未満の場合のみ、正しく動作します。

合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART レシーバの許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

M[1:0] = 01 の場合：

$$DWU = \frac{t_{WUUSART}}{11 \times Tbit}$$

M[1:0] = 00 の場合：

$$DWU = \frac{t_{WUUSART}}{10 \times Tbit}$$

M[1:0] = 10 の場合：

$$DWU = \frac{t_{WUUSART}}{9 \times Tbit}$$

$t_{WUUSART}$ は、スタート･ビットの立ち下がりエッジが検出されてから、（ペリフェラルからリクエストされた）クロックが動作可能状態になってペリフェラルに到達し、レギュレータが動作可能状態になるまでの時間です。

USART レシーバは、次の設定に応じて、表 346 および表 347 で指定された最大許容偏差まで、データを正しく受信できます。

- USART_CR1 レジスタの M ビットによって定義された 9、10 または 11 ビットのキャラクタ長
- USART_CR1 レジスタの OVER8 ビットによって定義された 8 倍または 16 倍のオーバーサンプリング
- USART_BRR レジスタのビット BRR[3:0] が 0000 に等しいかどうか。
- データのサンプリングに 1 ビットを使用するか 3 ビットを使用するか（USART_CR3 レジスタの ONEBIT ビットの値に依存）。

表 346. BRR [3:0] = 0000 のときの USART レシーバの許容誤差

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.75%	4.375%	2.50%	3.75%
01	3.41%	3.97%	2.27%	3.41%
10	4.16%	4.86%	2.77%	4.16%

表 347. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT = 0	ONEBIT = 1	ONEBIT = 0	ONEBIT = 1
00	3.33%	3.88%	2%	3%
01	3.03%	3.53%	1.82%	2.73%
10	3.7%	4.31%	2.22%	3.33%

注： **表 346 および表 347 で指定されたデータは、M ビット = 00 で受信フレームに正確に 10 ビット時間（M = 01 の場合は 11 ビット時間、M = 10 の場合は 9 ビット時間）のアイドル・フレームがいくつか含まれる特殊なケースでは若干異なることがあります。**

37.5.9 USART 自動ボー・レート検出

USART は、1 キャラクタの受信に基づいて、USART_BRR レジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2 つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません。

- 16 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/16 までの範囲内です。
- 8 倍のオーバーサンプリングのとき、ボーレートは usart_ker_ck_pres/65535 から usart_ker_ck_pres/8 までの範囲内です。

自動ボーレート検出を有効にする前に、USART_CR2 レジスタの ABRMOD[1:0] フィールドによって自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、4 つのモードがあります。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

これらのモードは以下のとおりです。

- **モード 0** : “1”のビットで始まるキャラクタ。

  この場合、USART はスタートビットの時間を測定します（立ち下がりエッジから立ち上がりエッジまで）。

- **モード 1**：10xx ビットパターンで始まるキャラクタ。

  この場合、USART はスタートと最初のデータビットの時間を測定します。低速な信号スロープの場合の精度を高めるために、測定は立ち下がりエッジから立ち下がりエッジまでで行われます。

- **モード 2**: 0x7F キャラクタフレーム（LSB ファーストモードでは 0x7F キャラクタ、または MSB ファーストモードでは 0xFE）。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 6 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR6）。ビット 0 からビット 6 までが BRs でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。

- **モード 3** : 0x55 キャラクタフレーム。

  この場合、ボーレートは、最初はスタートビット（BRs）の終了時に更新され、次にビット 0 の終了時に更新されます（立ち下がりエッジから立ち下がりエッジまで行われた測定に基づいて、BR0）、最後にビット 6（BR6）の終了時に更新されます。ビット 0 は BRs でサンプリングされ、ビット 1 からビット 6 までは BR0 でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。並行して、RX ラインの中間遷移ごとに別のチェックが行われます。RX の遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット 0 で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USART_BRR レジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USART_CR2 レジスタの ABREN ビットをセットします。USART は、RX ラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USART_ISR レジスタの ABRF フラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR 値が破損して、ABRE エラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が 16 から 65536 までのクロック周期でなく（16 倍のオーバーサンプリング）、8 から 65536 までのクロック周期でない（8 倍のオーバーサンプリング））。

その後、ABRF フラグをリセットすることによって（“0”を書き込むことによって）、自動ボーレート検出を再開できます。

FIFO マネージメントが無効になっていて、自動ボーレートエラーが発生した場合、RXNE および FE ビットを通して ABRE フラグがセットされます。

FIFO マネージメントが有効になっていて、自動ボーレートエラーが発生した場合、RXFNE および FE ビットを通して ABRE フラグがセットされます。

FIFO モードが有効な場合、自動ボーレート検出は、RXFIFO の最初の場所のデータを使用して行うべきです。したがって、自動ボーレート検出を開始する前に、USART_ISR レジスタの RXFNE フラグをチェックして RXFIFO が空であることを確認してください。

注 : **自動ボー・レート動作中に USART を無効にすると（UE = 0）、BRR 値が破損することがあります。**

37.5.10 USART マルチプロセッサ通信

USART のマルチプロセッサ通信が可能です（ネットワーク内で複数の USART を接続して）。たとえば、1 つの USART をマスタとして、その TX 出力を別の USART の RX 入力に接続することができます。一方、他の USART は、それぞれの TX 出力の論理和をとってマスタの RX 入力に接続して、スレーブとなります。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する USART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USART_CR1 レジスタの MME ビットをセットする必要があります。

注： **FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 usart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- USART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、USART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

USART は、USART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

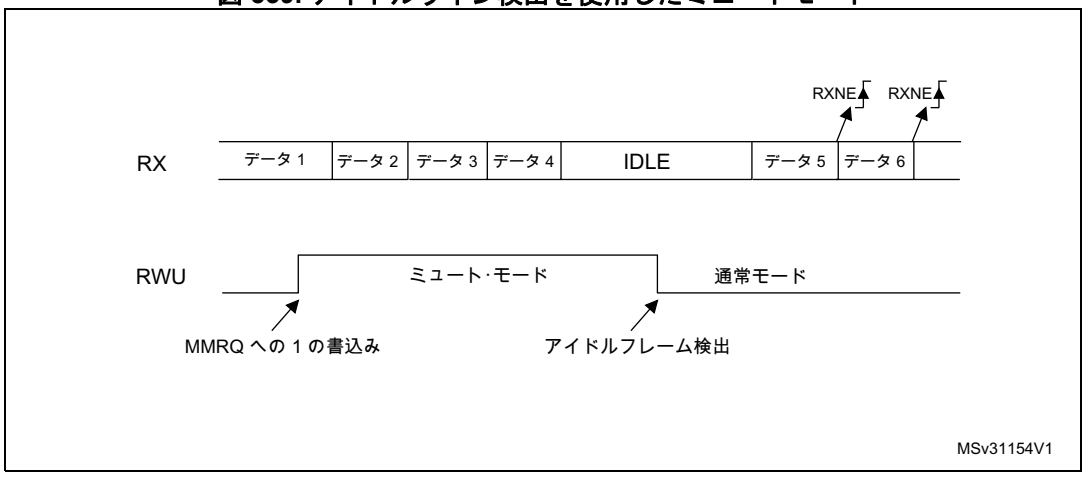
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドル･ライン検出（WAKE = 0）

MMRQ ビットに“1”が書き込まれ、RWU が自動的にセットされると、USART はミュート･モードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、USART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図 539 に示します。

図 539. アイドルライン検出を使用したミュートモード



注： **IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。**

ラインが IDLE のときに USART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビット･アドレス･マーク検出（WAKE = 1）

このモードでは、MSB が“1”の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、USART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割込みも DMA リクエストも発行されません。FIFO 管理が有効になっている場合、ミュートモードに入る前に、RXFIFO に少なくとも 1 つの空き場所があることをソフトウェアによって確認するべきです。

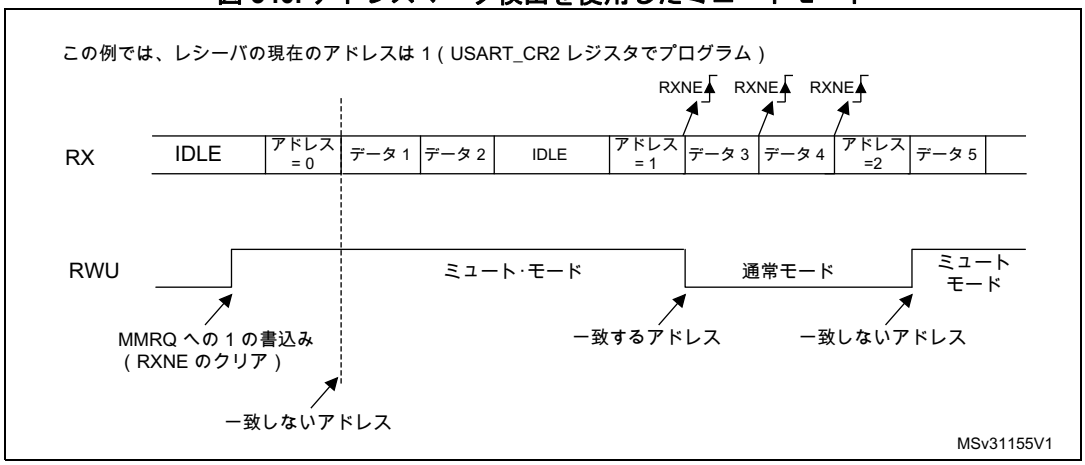
MMRQ ビットに 1 が書き込まれたときにも、USART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ がセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を図 540 に示します。

図 540. アドレスマーク検出を使用したミュートモード



37.5.11 USART Modbus 通信

USART は、Modbus/RTU および Modbus/ASCII プロトコルの実装に対する基本的サポートを備えています。Modbus/RTU は、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USART は、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1 つのブロックの終了は 2 キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割込みは、USART_CR2 レジスタの RTOEN ビットと、USART_CR1 レジスタの RTOIE を通じて有効にする必要があります。2 キャラクタ時間に対応する値（たとえば、22 x ビット時間）を RTO レジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USART は、キャラクター致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドで LF ASCII コードをプログラムし、キャラクター致割込みを有効にすることによって（CMIE = 1）、LF が受信されたときにソフトウェアに通知し、ソフトウェアは DMA バッファの CR/LF をチェックできます。

37.5.12 USART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、USART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な USART フレームフォーマットを表 348 に示します。

表 348. USART フレームのフォーマット

M ビット	PCE ビット	USART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの 値に応じて）とパリティビットから構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 のように 4 つのビットがセットされた場合、偶数パリティを選択していれば（USART_CR1 の PS ビット = 0）、パリティ･ビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 のように 4 つのビットがセットされた場合、奇数パリティを選択していれば（USART_CR1 の PS ビット = 1）、パリティ･ビットは 1 になります。

受信中のパリティ･チェック

パリティチェックに失敗した場合、USART_ISR レジスタの PE フラグがセットされ、USART_CR1 レジスタの PEIE ビットがセットされている場合は割込みが生成されます。PE フラグは、USART_ICR レジスタの PECF に 1 を書き込むことによってクリアされます。

送信中のパリティ生成

USART_CR1 の PCE ビットがセットされている場合、データ･レジスタに書き込まれたデータの MSB ビットは送信されますが、パリティ･ビットによって変更され、偶数パリティを選択している場合（PS = 0）は「1」が偶数個、奇数パリティを選択している場合（PS = 1）は「1」が奇数個になります。

37.5.13 USART LIN（Local Interconnection Network）モード

このセクションは、LIN モードがサポートされるときにのみ適用されます。1564 ページのセクション 37.4：USART の実装を参照してください。

LIN モードを選択するには、USART_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの CLKEN
- USART_CR3 レジスタの STOP[1:0]、SCEN、HDSEL、および IREN

LIN 送信

LIN マスタ送信の場合、セクション 37.5.4：に記述されている手順を適用する必要があります。通常の USART 送信と同じですが、次のような違いがあります。

- 8 ビットのワード長を設定するには M ビットをクリアします。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13 個の “0” ビットがブレークキャラクタとして送信されます。その後、値 “1” の 2bit が送信され、次の START 検出が可能になります。

LIN 受信

LIN モードが有効になると、ブレーク検出回路が有効になります。この検出は、通常の USART レシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生のたびにブレークが検出できます。

レシーバが有効になると（USART_CR1 レジスタの RE = 1）、RX 入力の START 信号を探します。スタート･ビットの検出方法は、ブレーク･キャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます（8、9、および 10番目のサンプル）。10 個（USART_CR2 レジスタの LBDL = 0）または 11 個（USART_CR2 レジスタの LBDL = 1）の連続したビットが“0”として検出され、その後にデリミタ･キャラクタが続く場合、USART_ISR レジスタの LBDF フラグがセットされます。LBDIE ビット = 1 の場合、割込みが生成されます。ブレークを確認する前に、RX ラインがハイ･レベルに戻ったことを知らせるデリミタが検査されます。

この 10 または 11 が発生する前に“1”がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LIN モードが無効にされた場合（LINEN = 0）、レシーバは、ブレーク検出を考慮することなく、通常の USART として機能しつづけます。

LIN モードが有効にされた場合（LINEN = 1）、フレーミング･エラーが発生（つまり、ブレーク･フレームの場合と同様に、“0” の位置でストップ･ビットを検出）すると、レシーバは停止し、ブレーク検出回路が “1”（ブレーク･ワードが完全でなかった場合）またはデリミタ･キャラクタ（ブレークが検出された場合）を受信するまで停止状態を維持します。

ブレーク検出回路ステート･マシンの動作とブレーク･フラグを 図 541：1588 ページのLIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）に示します。

ブレークフレームの例を図 542:1589 ページのLIN モードでのブレーク検出とフレーミングエラー検出 に示します。

図 541. LIN モードでのブレーク検出（11 ビットブレーク長、LBDL=1）

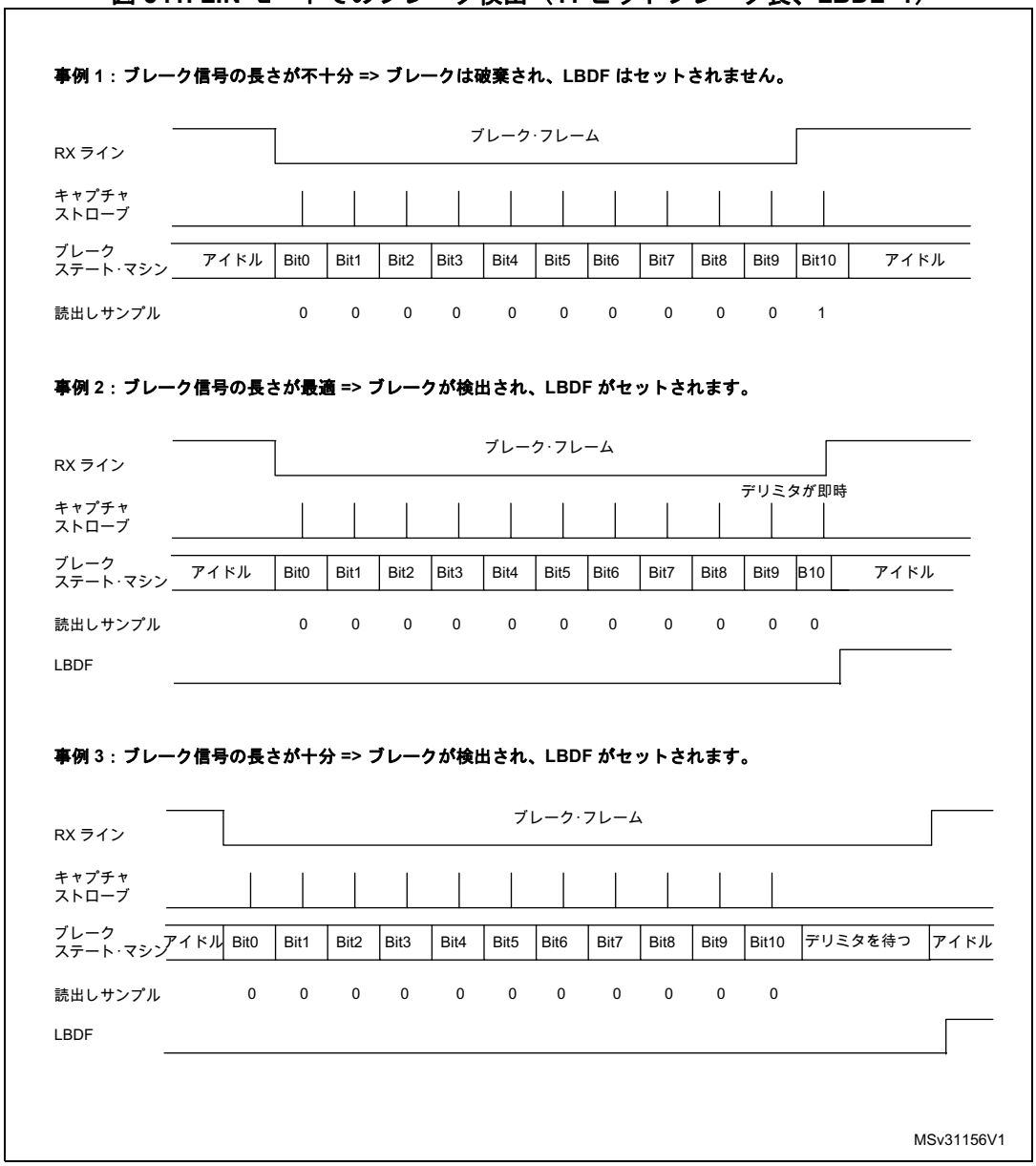
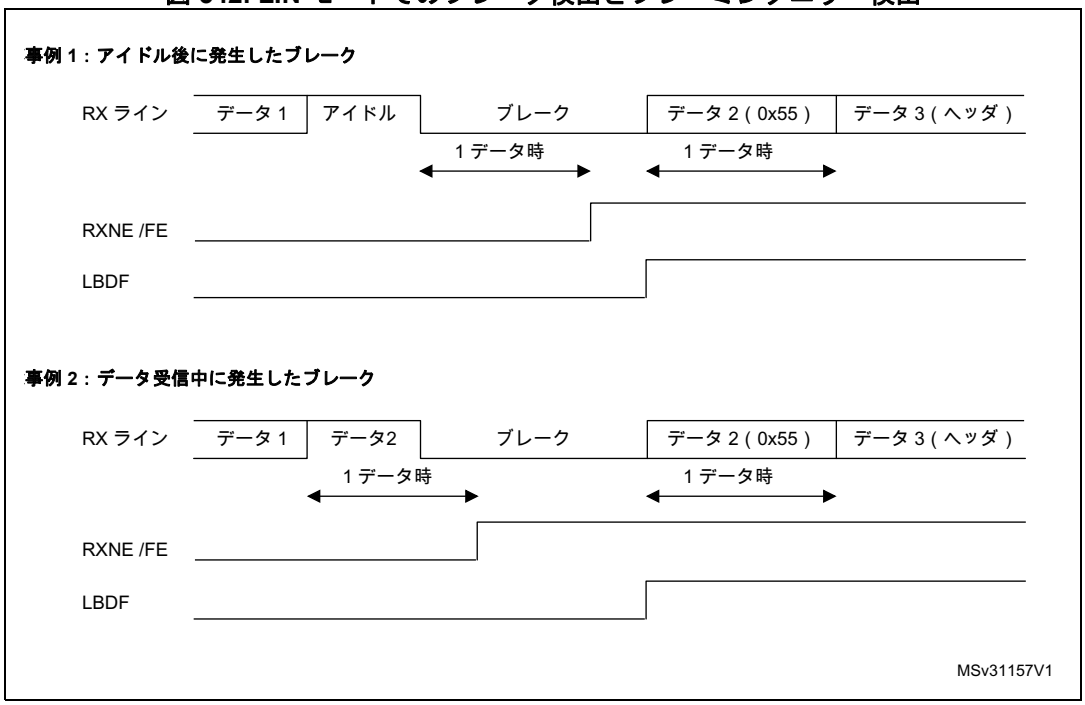


図 542. LIN モードでのブレーク検出とフレーミングエラー検出



37.5.14 USART 同期モード

マスタモード

同期マスタモードを選択するには、USART_CR2 レジスタの CLKEN ビットを“1”にプログラムします。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をマスタモードで制御できます。CK ピンは USART トランスミッタクロックの出力です。スタートビットとストップビットの処理中には、CK ピンにクロックパルスは送信されません。USART_CR2 レジスタの LBCL ビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USART_CR2 レジスタの CPOL ビットは、クロック極性を選択するために使用され、USART_CR2 レジスタの CPHA ビットは、外部クロックの位相を選択するために使用されます（図 543、図 544、および図 545 を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、外部 CK クロックは起動されません。

同期マスタモードでは、USART トランスミッタは非同期モードの場合とまったく同じように動作します。しかし、CPOL と CPHA に基づいて CK が TX と同期するので、TX 上のデータが同期します。

同期マスタモードでは、USART レシーバは非同期モードの場合とは異なる動作をします。RE が 1 にセットされた場合、データはオーバーサンプリングなしで、CK（CPOL と CPHA に応じて立ち上がりまたは立ち下がりエッジ）でサンプリングされます。ボーレート（1/16 ビット時間）に応じたセットアップ時間とホールド時間に従う必要があります。

注： マスタ･モードでは、CK ピンは TX ピンと連携して動作します。したがって、クロックが供給されるのは、トランスミッタが有効であり（TE = 1）、データが送信中（USART_TDR データ･レジスタ への書込み）の場合に限られます。つまり、データを送信せずに同期データを受信することはできません。

図 543. USART の同期マスタ送信の例

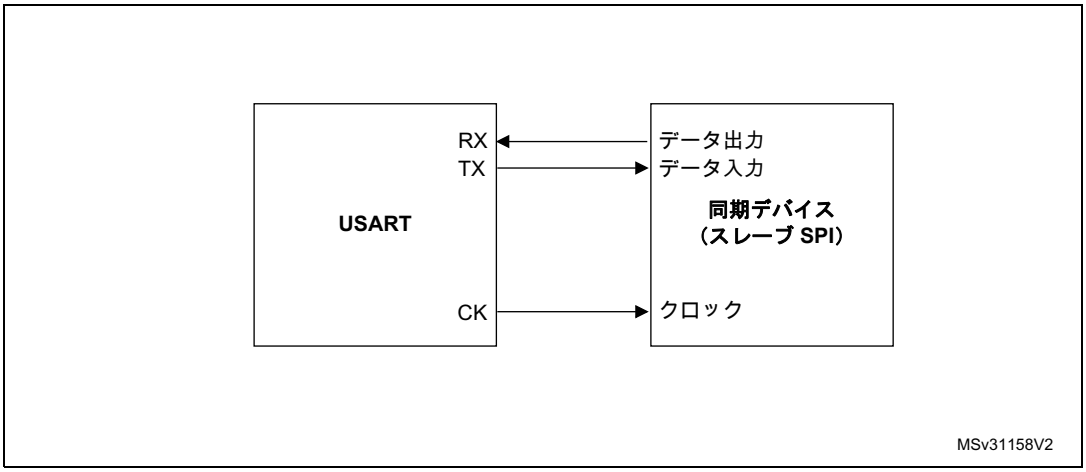


図 544. 同期マスタモードでの USART データクロックタイミング図（M ビット = 00）

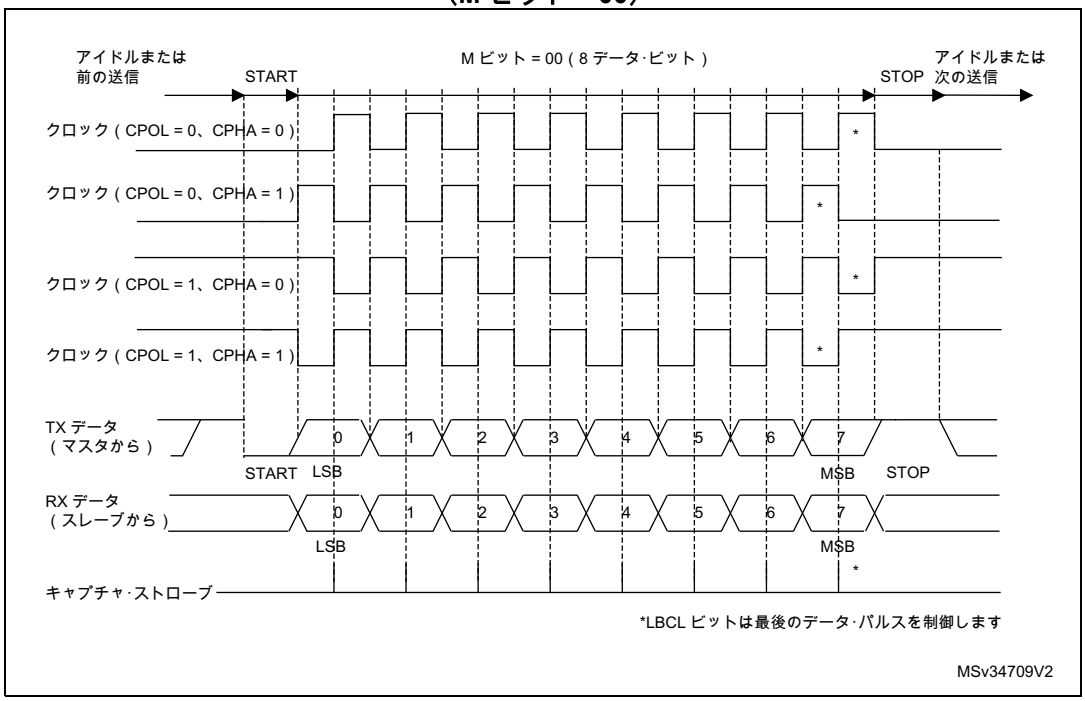
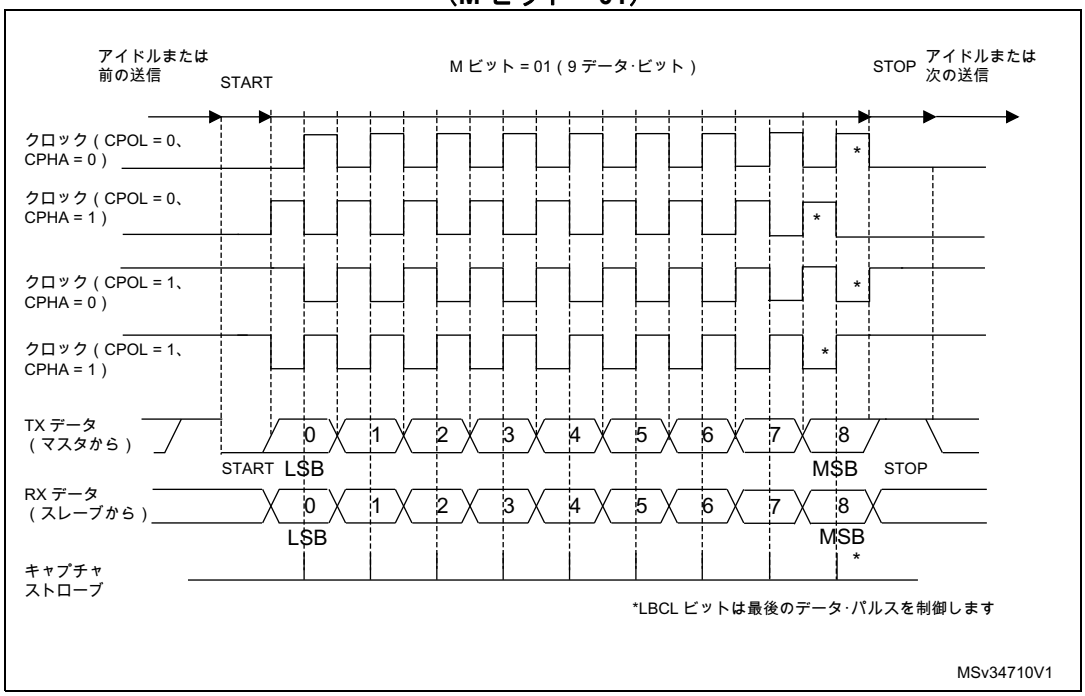


図 545. 同期マスタ･モードでの USART データ･クロック･タイミング図（M ビット = 01）



スレーブモード

同期スレーブモードを選択するには、USART_CR2 レジスタの SLVEN ビットを“1”にプログラムします。同期スレーブモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をスレーブモードで制御できます。CK ピンは、スレーブモードの USART の入力です。

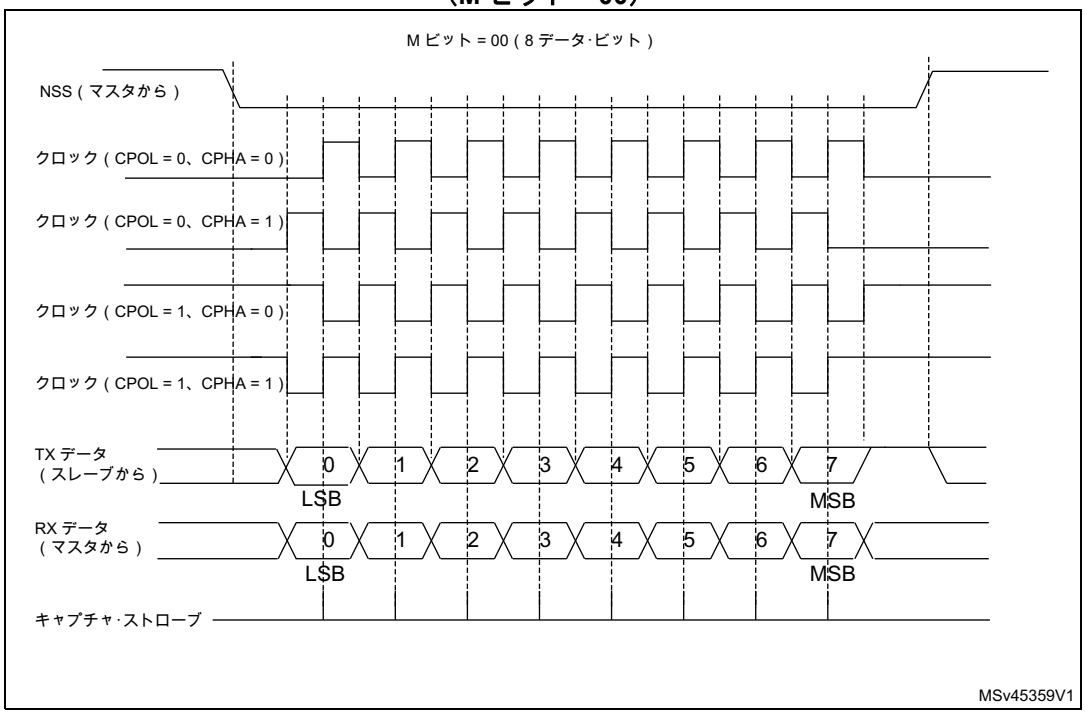
注： **ペリフェラルが SPI のスレーブモードで使用されるとき、ペリフェラルのクロックソースの周波数（usart_ker_ck_pres）は CK 入力周波数の 3 倍より大きくなければなりません。**

USART_CR2 レジスタの CPOL ビットと CPHA ビットは、それぞれクロック極性と外部クロックの位相を選択するために使用されます（図 546 を参照）。

アンダーランエラーフラグはスレーブ送信モードで使用できます。ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。

スレーブはハードウェアおよびソフトウェアの NSS 管理をサポートしています。

図 546. 同期スレーブモードでの USART データクロックタイミング図（M ビット = 00）



スレーブ選択（NSS）ピンの管理

ハードウェアまたはソフトウェアのスレーブ選択管理は、USART_CR2 レジスタの DIS_NSS ビットを使用して設定することができます。

- ソフトウェア NSS 管理（DIS_NSS = 1）

  SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

  外部NSS ピンは他のアプリケーションで使用できます。

- ハードウェア NSS 管理（DIS_NSS = 0）

  SPI スレーブ選択は NSS 入力ピンに依存します。NSS がローレベルのときにスレーブが選択され、NSS がハイレベルのときには選択されません。

注：

USART が無効にされたときには（UE = 0）、クロック·パルスが正常に機能するように、LBCL（SPI マスタ·モードでのみ使用）、CPOL、および CPHA ビットを選択する必要があります。

SPI スレーブ·モードでは、マスタ通信を開始する前に（またはクロックが安定しているときのフレーム間で）、USART を有効にする必要があります。そうしないと、マスタがフレームの中央に位置しているときに USART スレーブが有効化されると、スレーブはマスタとの同期がとれなくなります。スレーブのデータレジスタは、通信クロックの最初のエッジまたは現在の通信の終了より前に、準備ができている必要があります。そうしないと、SPI スレーブはゼロを送信します。

SPI スレーブのアンダーランエラー

アンダーランエラーが起きると、USART_ISR レジスタの UDR フラグがセットされ、ソフトウェアによってアンダーランエラーフラグがクリアされるまで SPI スレーブは最後のデータを送信し続けます。

アンダーランフラグはフレームの開始時にセットされます。USART_CR3 レジスタの EIE ビットがセットされている場合、アンダーランエラー割込みがトリガされます。

アンダーランエラーフラグは、USART_ICR レジスタのビット UDRCF をセットすることによってクリアされます。

アンダーランエラーの場合、TDR レジスタに書き込むことは可能です。アンダーランエラーをクリアすると、新しいデータを送信できます。

アンダーランエラーが起きて、TDR に書き込まれる新しいデータがない場合は、フレームの終りに TC フラグがセットされます。

注： **データが USART_TDR に書き込まれた時点が CK の最初の送信エッジに近すぎると、アンダーランエラーが起きることがあります。このアンダーランエラーを避けるために、USART_TDR への書込みは、最初の CK エッジより usart_ker_ck の 3 サイクル前に行うべきです。**

37.5.15 USART 単線半二重通信

単線半二重モードを選択するには、USART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN および CLKEN ビット
- USART_CR3 レジスタの SCEN および IREN ビット

USART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の USART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

37.5.16 USART レシーバタイムアウト

レシーバタイムアウト機能を有効にするには、USART_CR2 制御レジスタの RTOEN ビットをセットします。

タイムアウト経過時間は USART_RTOR レジスタの RTO ビットフィールドを使用してプログラムします。

レシーバタイムアウトカウンタは次の時点からカウントを開始します。

- STOP = “00” または STOP = “11” の場合、ストップビットの最後から
- STOP = “10” の場合、2 番目のストップビットの最後から。
- STOP = “01” の場合、ストップビットの最初から。

タイムアウト経過時間が過ぎると、USART_ISR レジスタの RTOF フラグがセットされます。USART_CR1レジスタの RTOIE ビットがセットされている場合、タイムアウトが生成されます。

37.5.17 USART スマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。1564 ページのセクション 37.4：USART の実装を参照してください。

スマートカードモードを選択するには、USART_CR3 レジスタの SCEN ビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL および IREN ビット

スマートカードにクロックを供給するために CLKEN ビットもセットすることができます。

スマートカードインタフェースは、ISO 7816-3 標準で定義された非同期スマートカードプロトコルをサポートするように設計されています。T = 0（キャラクタ･モード）と T = 1（ブロック･モード）の両方がサポートされます。

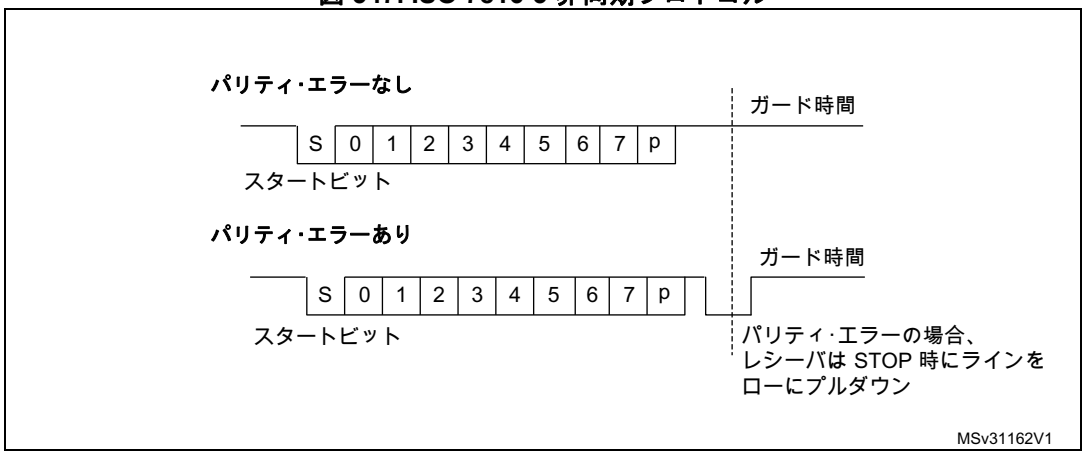
USART は次のように設定します。

- 8 ビット＋パリティ：USART_CR1 レジスタの M = 1 および PCE = 1
- データ送受信ではストップ･ビット 1.5 個：USART_CR2 レジスタの STOP = “11”。受信ではストップ･ビット 0.5 個を選択することもできます。

T = 0（キャラクタ）モードでは、パリティ･エラーはガード･タイム中の各キャラクタの終わりに示されます。

図 547 に、パリティ･エラーの有無によるデータ･ラインの状況の変化の例を示します。

図 547. ISO 7816-3 非同期プロトコル



スマートカードと接続されると、USART の TX 出力は、やはりスマートカードによって駆動される双方向ラインを駆動します。TX ピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

- 送信シフトレジスタからのデータの送信は、少なくとも 1/2 ボークロックの遅れが保証されます。通常動作では、フルの送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された 1/2 ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって（NACK）、この条件を USART に知らせます。この NACK 信号（1 ボークロックの間、送信ラインをローに引き下げ）は、1.5 個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USART は、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNT ビットフィールドでプログラムされます。プログ

ラムされた再試行回数後も USART が NACK を受信し続けた場合は、送信を停止して、エラーをフレーミングエラーとして通知します。TXE ビット（FIFO モードが有効な場合は TXFNF ビット）は、USART_RQR レジスタの TXFRQ ビットを使用してセットできます。

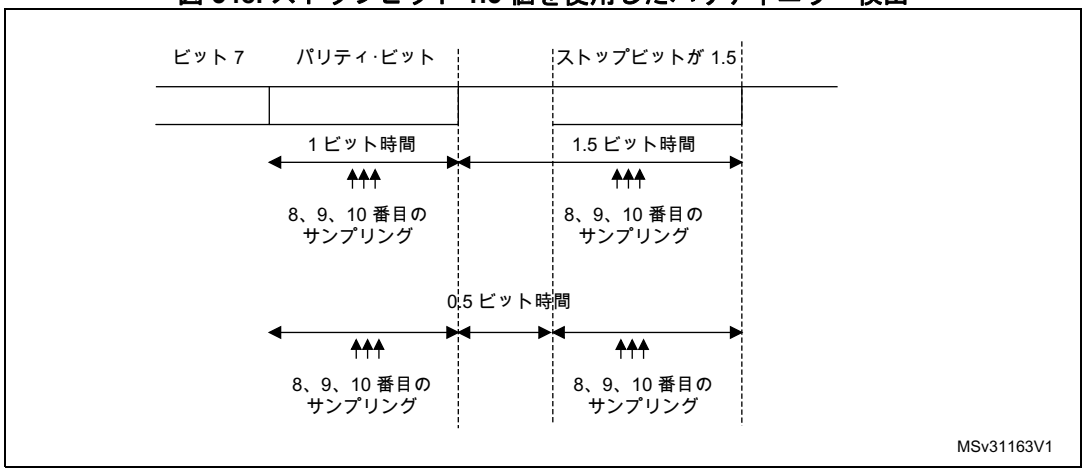
- 送信時のスマートカード自動再試行：USART による NACK の検出と反復キャラクタのスタートビットの間に 2.5 ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちに TC ビットがセットされます（ガードタイムはありません）。ソフトウェアで再び繰り返したい場合は、規格によって指定されている 2 ボー周期以上を確保する必要があります。
- 1.5 個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後 1 ボークロック周期間、送信ラインがローに引き下げられます。これは、USART に送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK 制御ビットがセットされている場合、パリティ・エラーはレシーバによって “NACK” され、そうでない場合、NACK は送信されません（T = 1 モードで使用されます）。受信したキャラクタにエラーがあった場合、RXNE（FIFO モードが有効な場合は RXFNE）／受信 DMA リクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNT ビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USART は NACK の送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行：USART がカードを NACK したが、カードがキャラクタを繰り返さなかった場合、BUSY フラグはセットされたままです。
- 送信時、USART は 2 つの連続するキャラクタの間にガードタイム（ガードタイムレジスタでプログラム）を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的の CGT（7816-3 仕様で定義されている Character Guard Time）から 12（1 キャラクタの時間）を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。TCBGT フラグは、ガード時間完了を待たずにデータ転送の終わりを検出するために使用できます。このフラグは、フレーム送信終了後およびカードから NACK を受信しなかった場合にのみセットされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの NACK によってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、この NACK をスタートビットとして検出しません。ISO プロトコルによれば、受信される NACK の期間は 1 または 2 ボークロック周期です。
- レシーバ側では、パリティエラーが検出されて NACK が送信された場合、レシーバはこの NACK をスタートビットとして検出しません。

注： **スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ 0x00 は、ブレークではなくデータとして処理されます。**

TE ビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISO プロトコルでは定義されていません。

図 548 に、USART による NACK 信号のサンプリング方法を示します。この例では、USART はデータを送信中であり、ストップビットが 1.5 個組み込まれています。データと NACK 信号の整合性を検査するために、USART のレシーバ部が有効にされます。

図 548. ストップビット 1.5 個を使用したパリティエラー検出



USART は、CK 出力を通じてスマートカードにクロックを供給できます。スマートカード·モードでは、CK は通信に関係せず、5 ビットのプリスケーラを経て内部のペリフェラル入力クロックから派生されます。この分周比は、USART_GTPR レジスタで設定されます。CK の周波数は usart_ker_ck_pres/2 から usart_ker_ck_pres/62 の間でプログラミングできます。ここで、usart_ker_ck_pres は、プログラムされたプリスケーラで分周されたペリフェラル入力クロックです。

ブロック·モード（T = 1）

T = 1（ブロック）モードでは、パリティ·エラー送信は、USART_CR3 レジスタの NACK ビットをクリアすることによって無効化できます。

ブロック·モードでスマートカードからの読出しをリクエストするときには、ソフトウェアは RTOR レジスタを BWT（ブロック·ウェイト·タイム）- 11 の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、タイムアウト割込みが生成されます。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE/RXFNE 割込みによって通知されます。

注： **ブロックモードのスマートカードからの読出しに USART を DMA モードで使用するときでも、RXNE/RXFNE 割込みを有効にする必要があります。並行して、DMA は最初の受信バイトの後でのみ有効にする必要があります。**

2つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後（RXNE/RXFNE 割込み）、RTO レジスタを CWT（キャラクタウェイトタイム -11）の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT 未満の時間内に新しいキャラクタを送信しなかった場合、USART は RTOF フラグと割込み（RTOIE ビットがセットされているとき）によって、これをソフトウェアに通知します。

注： **スマートカードプロトコルの定義にあるように、BWT/CWT 値は最後のキャラクタの開始（スタートビット）から定義されるべきです。RTO レジスタは、最後のキャラクタ自体の長さを考慮して、それぞれ BWT -11 または CWT -11 にプログラムする必要があります。**

ブロック長カウンタは、USART が受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USART の送信時にリセットされます。ブロックの長さは、スマートカードによってブロックの 3 番目のバイト（プロローグフィールド）で伝えられます。この値を USART_RTOR レジスタの BLEN フィールドでプログラムする必要があります。DMA モードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値（0x0）にプログラムする必要があります。この値では、4 番目の受信キャラクタの後に割込みが生成されます。ソフトウェアは LEN フィールド（3 番目のバイト）を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割込み駆動受信モードでは、ブロックの長さはソフトウェアによって、または BLEN 値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLEN の最大値（0xFF）をプログラムすることができます。実際の値は、3番目のキャラクタの受信後にプログラムされます。

ブロックが LRC 水平冗長検査（1 エピローグ･バイト）を使用している場合は、BLEN = LEN です。ブロックが CRC メカニズム（2 エピローグ･バイト）を使用している場合は、BLEN = LEN + 1 をプログラムする必要があります。合計ブロック長（プロローグ、エピローグ、および情報フィールドを含む）は、BLEN + 4 に等しくなります。ブロックの終わりは EOBF フラグと割込み（EOBIE ビットがセットされているとき）によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりは RTO 割込みによって通知されます（キャラクタウェイトタイムオーバーフロー）。

注：　エラーチェックコード（LRC/CRC）は、ソフトウェアによって計算／確認されなければなりません。

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの 2 つコンベンションを定義しています。

ダイレクトコンベンションは、LSB ファースト、論理ビットの値 1 がラインの H 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST = 0、DATAINV = 0（デフォルト値）をプログラムする必要があります。

インバース･コンベンションは、MSB ファースト、論理ビットの値 1 が単線の L 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST = 1、DATAINV = 1 をプログラムする必要があります。

注：　論理データ値が反転されると（0 = H、1 = L）、パリティ･ビットも同じように反転されます。

カードのコンベンションが認識されるように、カードは最初のキャラクタ TS を ATR（Answer To Reset）フレームの最初のキャラクタとして送信します。TS には、LHHL LLL LLH と LHHL HHH LLH の 2 つのパターンが使えます。

- (H) LHHL LLL LLH は、インバースコンベンションをセットアップします。状態 L が値 1 にエンコードされ、モーメント 2 は最上位ビットを含みます（MSB ファースト）。インバースコンベンションによってデコードされると、送受信されたバイトは 3F に等しくなります。
- (H) LHHL HHH LLH は、ダイレクトコンベンションをセットアップします。状態 H が値 1 にエンコードされ、モーメント 2 は最下位ビットを含みます（LSB ファースト）。ダイレクトコンベンションによってデコードされると、送受信されたバイトは 3B に等しくなります。

キャラクタパリティは、2 から 10 までの 9 個のモーメントに 1 にセットされた偶数個のビットがあったときに正しいとみなされます。

USART はカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USART がダイレクト･コンベンション（デフォルト）で設定され、カードがインバース･コンベンションで応答することを前提とすると、TS = LHHL LLL LLH の場合、USART が受信したキャラクタは 03 になり、パリティは奇数になります。

したがって、TS パターン認識には次の 2 つの方式を使用できます。

方式 1

USART は、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TS パターンの受信によってパリティエラー割込みと、カードに対するエラー信号が生成されます。

- パリティエラー割込みは、カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USART をインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割込みに応答して、ソフトウェアは USART を再プログラムし、カードに対して新しいリセットコマンドを生成してから、TS を再び待つこともできます。

方式 2

USART は 9 ビット／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように 2 つの TS パターンのいずれかを受信します。

(H) LHHL LLL LLH = 0x103：インバース･コンベンションを選択

(H) LHHL HHH LLH = 0x13B：ダイレクト･コンベンションを選択

ソフトウェアは受信されたキャラクタをこの 2 つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えて USART をプログラムします。

2 つのうちのどちらも認識されなかった場合、ネゴシエーションを再開始するためにカードリセットが生成されます。

37.5.18 USART IrDA SIR ENDEC ブロック

このセクションは、IrDA モードがサポートされるときにのみ適用されます。1564 ページのセクション 37.4：USART の実装を参照してください。

IrDA モードを選択するには、USART_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN、STOP、および CLKEN ビット
- USART_CR3 レジスタの SCEN および HDSEL ビット

IrDA SIR 物理層は、ロジック 0 を赤外光パルスとして表現する RZI（Return to Zero, Inverted）変調方式の使用を指定します（図 549 を参照）。

SIR 送信エンコーダは、USART からの NRZ（Non Return to Zero）送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線 LED に送信されます。SIR ENDEC の場合、USART によるサポートは最大で 115.2 キロ･ボーのビット･レートに限られます。通常モードでは、送信されるパルス幅は、ビット周期の 3/16 として指定されます。

SIR 受信デコーダは、赤外線検出回路からの RZ（Return to Zero）ビットストリームを復調し、受信した NRZ シリアルビットストリームを USART に出力します。デコーダの入力は、アイドル状態のノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDA は半二重通信プロトコルです。トランスミッタがビジーである場合（USART が IrDA エンコーダにデータを送信しているとき）、IrDA デコーダは IrDA 受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USART がデコードされたデータを受信している）場合、IrDA は USART から IrDA への TX 上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。

- “0”はハイパルスとして送信され、“1”は“0”として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の 3/16 と規定されます（図 550 を参照）。
- SIR デコーダは、IrDA 準拠の受信信号を USART 用のビットストリームに変換します。
- SIR 受信ロジックは、ハイ状態を論理値 1 とみなし、ローパルスを論理値 0 とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR 出力は、アイドル時にロー状態になります。
- IrDA 仕様では、1.41 us より大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ端のグリッチ検出回路は、PSC 2 周期（PSC は USART_GTPR でプログラムされたプリスケーラ値）より小さな幅のパルスをフィルタします。PSC 1周期より小さな幅のパルスは常に拒否されますが、1周期以上 2周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2周期より大きな幅のパルスは、パルスとして受け入れられます。PSC = 0 のとき、IrDA エンコーダと IrDA デコーダは機能しません。
- レシーバは、低電力トランスミッタと通信できます。
- IrDA モードでは、USART_CR2 レジスタのストップビットを“1 ストップビット”に設定する必要があります。

IrDA 低電力モード

- トランスミッタ

  低電力モードでは、パルス幅はビット周期の 3/16 に維持されません。代わりに、パルス幅は低電力ボーレート（最小で 1.42 MHz）の 3 倍となります。一般的に、この値は 1.8432 MHz（1.42 MHz < PSC < 2.12 MHz）です。低電力モードのプログラム可能な分周器は、この値を得るためにシステム･クロックを分周します。

- レシーバ

  低電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USART は 1/PSC よりも短いパルスを破棄する必要があります。有効なローレベルは、その期間が IrDA 低電力ボークロック（USART_GTPR の PSC 値）の 2 周期分を超える場合にのみ受け入れられます。

注： **PSC 2 周期未満 1 周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。**

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA 物理層仕様では、送信と受信の間に最小 10 ms の遅延を指定しています（IrDA は半二重プロトコルです）。

図 549. IrDA SIR ENDEC ブロック図

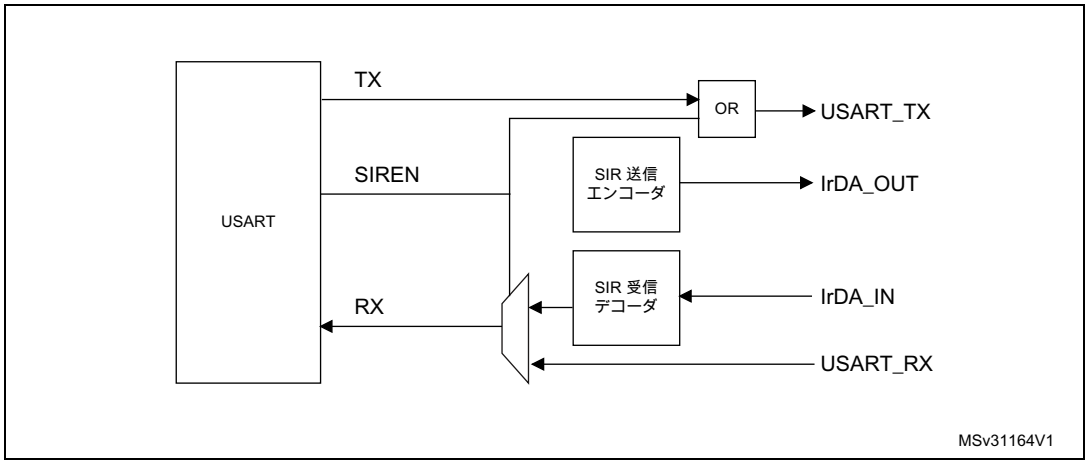
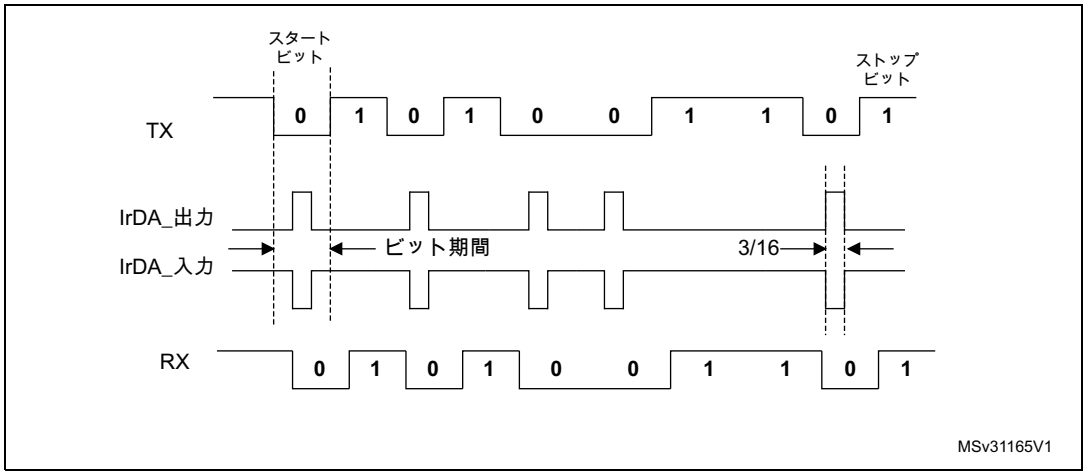


図 550. IrDA データ変調（3/16）- 通常モード



37.5.19 USART および DMA を使用した連続通信

USART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： **DMA モードがサポートされるかどうかについては、1564 ページのセクション 37.4 : USART の実装を参照してください。DMA がサポートされない場合は、セクション 37.5.6 : の説明に従って USART を使用してください。FIFO が無効の場合、連続通信を行うには、USART_ISR レジスタの TXE/ RXNE フラグをクリアします。**

DMA を使用した送信

DMA モードでの送信を有効にするには、USART_CR3 レジスタの DMAT ビットをセットします。TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされるたびに、データは、DMA ペリフェラル（対応するダイレクトメモリアクセスコントローラのセクションを参照）を使用して設定された SRAM 領域から USART_TDR レジスタにロードされます。DMA チャネルを USART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

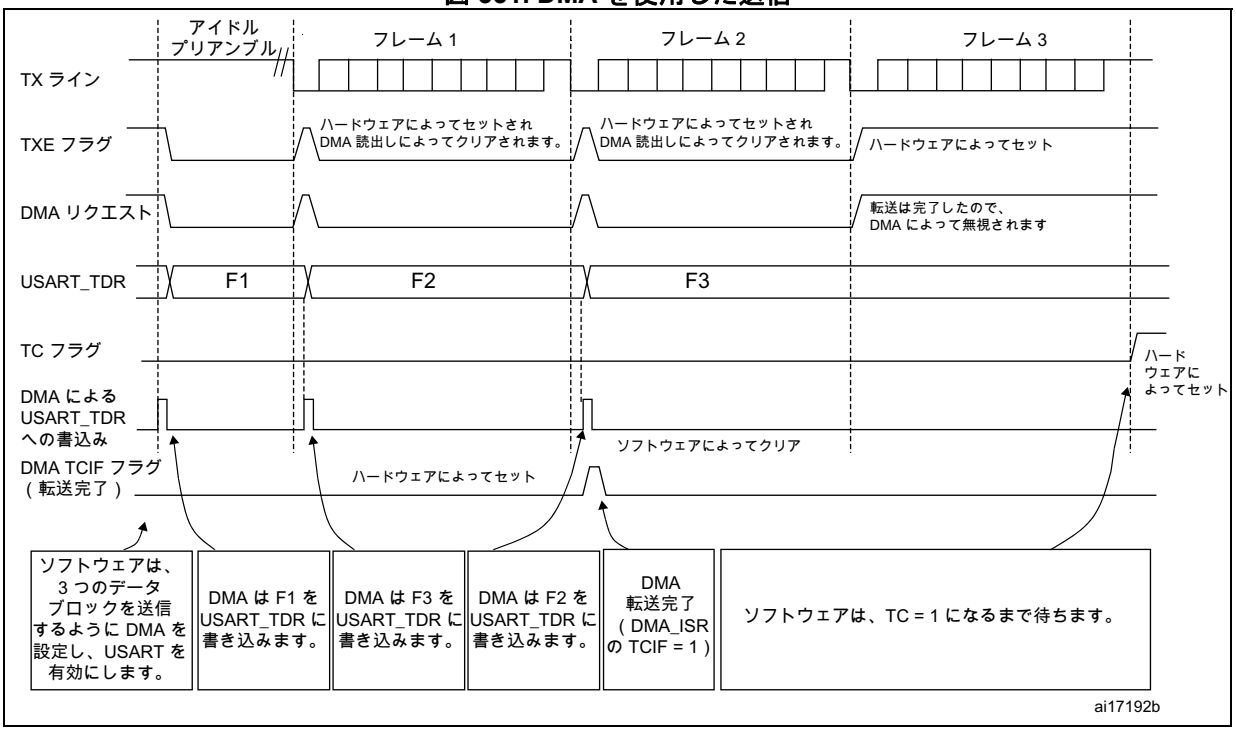
1. DMA 制御レジスタに USART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から USART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割込み生成を設定します。
6. USART_ICR レジスタの TCCF ビットをセットすることによって、USART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して USART 通信の完了を確認することができます。これは、USART を無効にする前に、またはペリフェラルクロックが無効のときシステムが低電力モードに入

る前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC = 1 になるまで待つ必要があります。データの転送中、TC フラグはクリアされた状態を維持し、最後のフレームの送信が終了するとハードウェアによってセットされます。

図 551. DMA を使用した送信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = 1）によってトリガされます。**

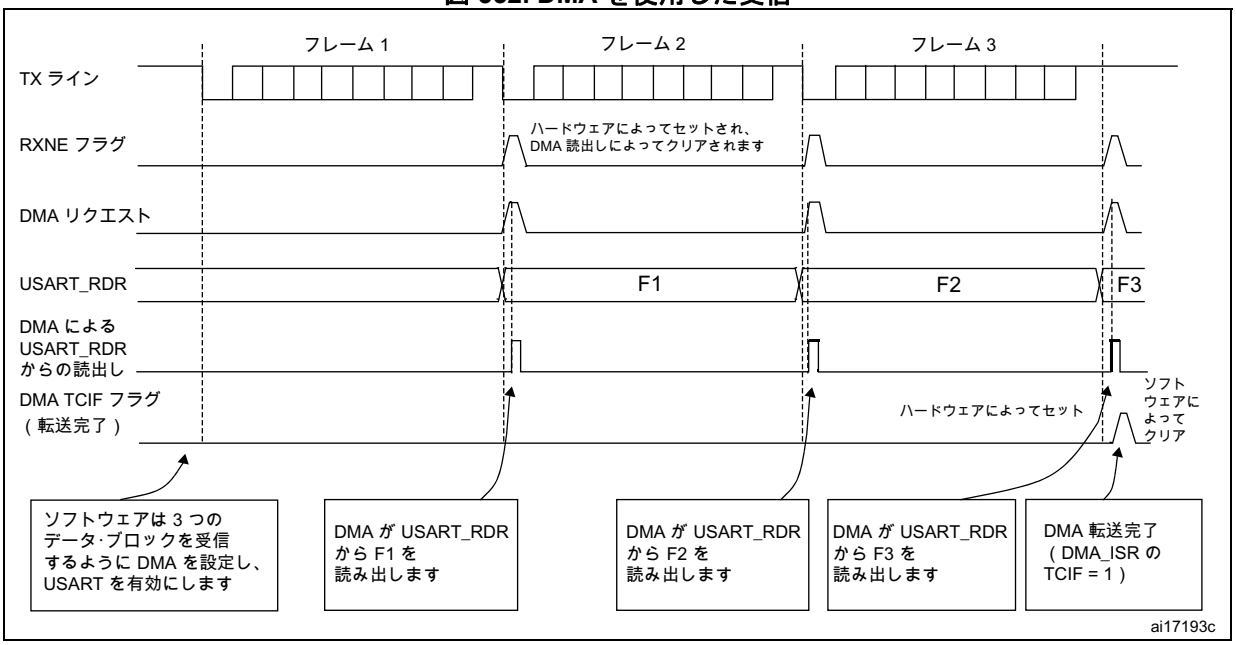
DMA を使用した受信

DMA モードでの受信を有効にするには、USART_CR3 レジスタの DMAR ビットをセットします。データバイトが受信されるたびに、データは、USART_RDR レジスタから DMA ペリフェラル（対応するダイレクトメモリアクセスコントローラのセクションを参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを USART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに USART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、USART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

図 552. DMA を使用した受信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。**

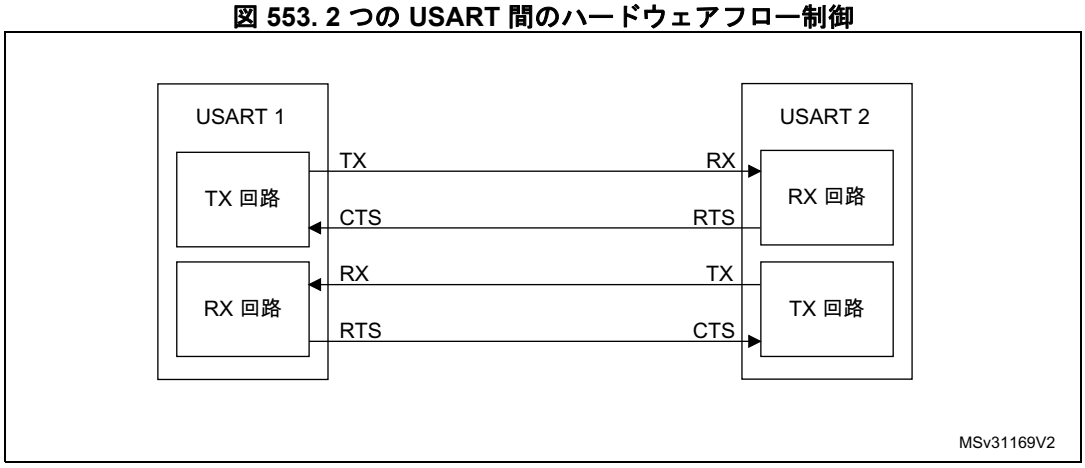
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（USART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

37.5.20 RS232 ハードウェアフロー制御および RS485 ドライバ有効

CTS 入力と RTS 出力を使用すると、2 つのデバイス間のシリアル·データ·フローを制御できます。図 553 に、このモードで 2 つのデバイスを接続する方法を示します。

図 553. 2 つの USART 間のハードウェアフロー制御

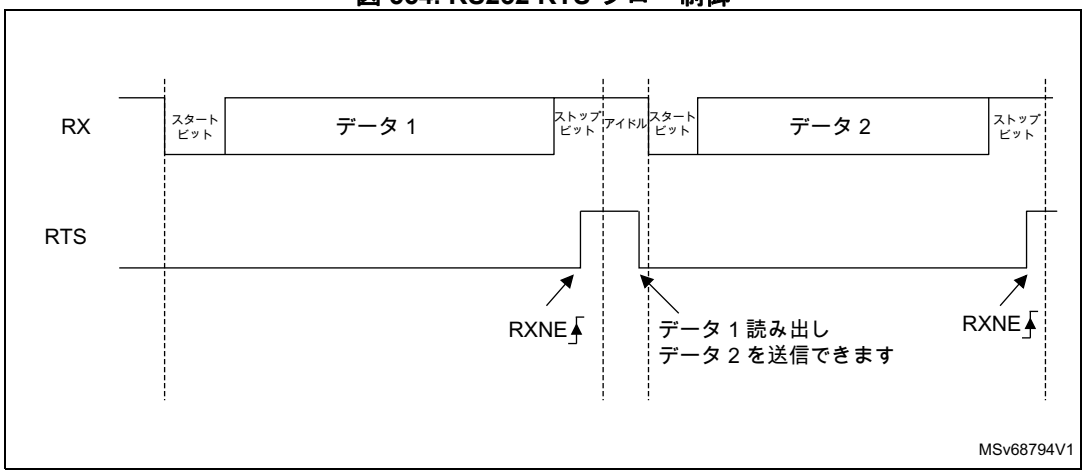


RS232 RTS と CTS のフロー制御は、USART_CR3 レジスタの RTSE ビットと CTSE ビットに“1”を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE = 1）、USART レシーバが新しいデータを受信可能である限り、RTS はネゲートされます（ローに固定）。受信レジスタがフルになると RTS がアサートされ、現在のフレームが終わると送信が停止する予定であることを示します。図 554 に、RTS フロー制御が有効な場合の通信例を示します。

図 554. RS232 RTS フロー制御



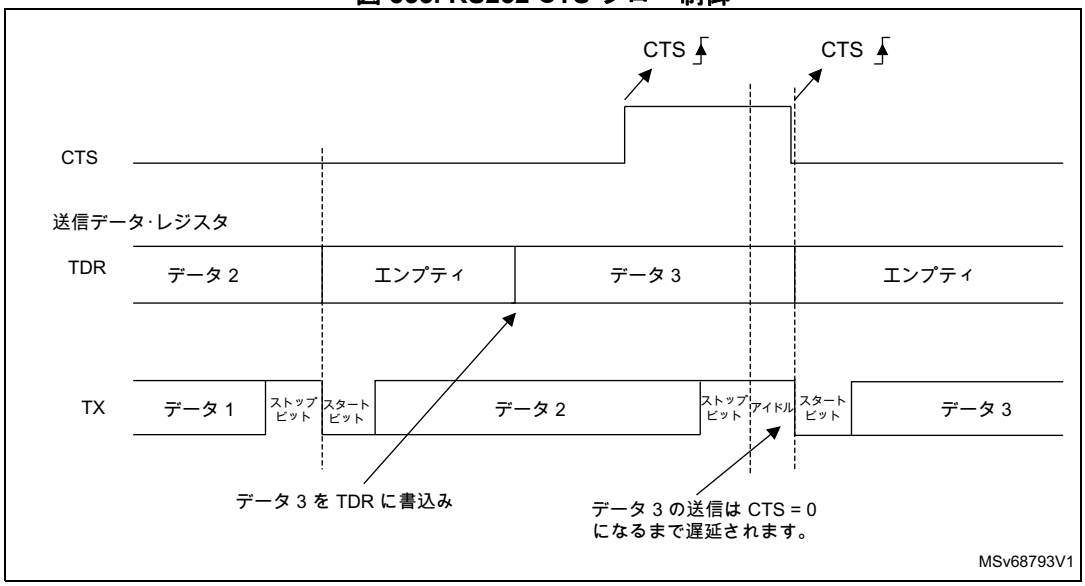
注： **FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、RTS がアサートされます。**

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE = 1）、トランスミッタは、CTS 入力をチェックしてから次のフレームを送信します。CTS がネゲートされている場合（ローに固定）、次のデータが送信されます（データが送信される状態であること、つまり TXE/TXFE = 0 であるとします）。そうでない場合、データは送信されません。送信中に CTS がアサートされると、現在の送信が完了してからトランスミッタが停止します。

CTSE = 1 の場合、CTS 入力がトグルすると、CTSIF ステータス･ビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。USART_CR3 レジスタの CTSIE ビットがセットされている場合、割込みが生成されます。図 555 に、CTS フロー制御が有効な場合の通信例を示します。

図 555. RS232 CTS フロー制御



注： **正しい動作とするには、現在のキャラクタが終了する少なくとも 3 USART クロック·ソース周期前に CTS をデアサートする必要があります。また、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。**

RS485 ドライバ有効

ドライバ有効機能を有効にするには、USART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。USART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。USART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、USART_CR3 制御レジスタの DEP ビットを使用して設定できます。

USART では、DEAT および DEDT はサンプル時間単位（オーバーサンプリングレートに応じて 1/8 または 1/16 ビット時間）で表されます。

37.5.21 USART 低電力管理

USART には、高度な低電力モード機能があり、usart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

USART は、UESM ビットがセットされているとき、MCU を低電力モードからウェイクアップできます。

Usart_pclk がゲートされているとき、**usart_pclk** クロックの有効化を必要とする特定の動作が必要になった場合、USART はウェイクアップ割込み（**usart_wkup**）を生成します。

- FIFO モードが無効の場合

  USART データレジスタを空にするために Usart_pclk クロックを有効にする必要があります。

  この場合、usart_wkup 割込みのソースは“1”にセットされた RXNE です。低電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために Usart_pclk クロックを有効にする必要があります。

  – TXFIFO を満たすため
  – または RXFIFO を空にするため

  この場合、usart_wkup 割込みのソースになる可能性のあるものは以下の通りです。

  – RXFIFO ノット･エンプティ。この場合、低電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  – RXFIFO フル。この場合、低電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  – TXFIFO エンプティ。この場合、低電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、usart_wkup 割込みソースになり得るのは次のイベントのうちの 1 つです。

  – TXFIFO 閾値に達した。この場合、低電力モードに入る前に TXFTIE ビットをセットする必要があります。
  – RXFIFO 閾値に達した。この場合、低電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信するのに必要な時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFO ノットエンプティ、および RXFIFO/TXFIFO 閾値割込みを使用すれば、低電力モード中にできるだけ多くの USART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の **usart_wkup** 割込みを選択することもできます。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は **usart_wkup** 割込みが生成されます。

注： **低電力モードに移行する前に、USART 転送が進行中ではないことを確認します。BUSY フラグをチェックすることでは、データ受信中に低電力モードに入らないことを保証できません。**

WUF フラグは、MCU が 低電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低電力モードに入るときには、REACK ビットをチェックして、USART が有効であることを確認する必要があります。

受信に DMA が使用されるときには、低電力モードに入る前に無効化し、低電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低電力モードでのミュートモードの使用

低電力モードに入る前に USART がミュートモードになった場合は、

- アイドル検出は低電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： **FIFO 管理が有効なとき、ミュートモードは何の制約もなく低電力モードからのウェイクアップとともに使用できます（すなわち、ミュートおよび低電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。**

低電力モードで USART カーネルクロック（usart_ker_ck）がオフのときの低電力モードからのウェイクアップ

低電力モード中、usart_ker_ck クロックがオフになっている場合、USART 受信ラインの立ち下がりエッジが検出されると、usart_ker_ck_req 信号によって USART インタフェースが usart_ker_ck クロックをオンにするようリクエストします。その後、usart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップイベントが確認されない場合、usart_ker_ck クロックが再度オフになり、MCU がウェイクアップせずに低電力モードに留まり、カーネルクロックリクエストが解除されます。

以下の例は、ウェイクアップイベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 556 に、ウェイクアップイベントが確認された時の USART の動作を示します。

図 556. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

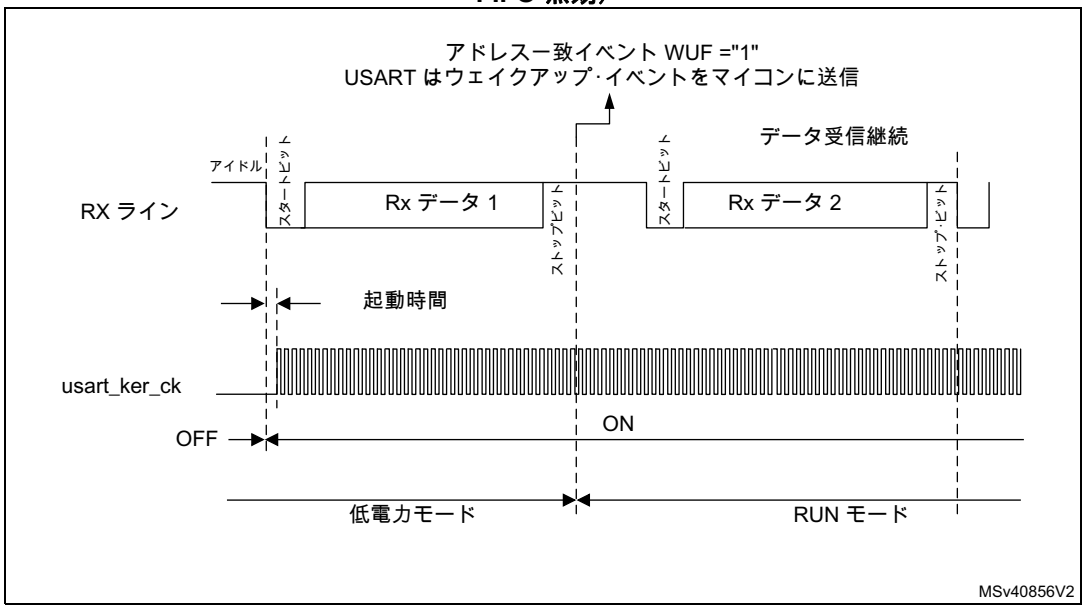
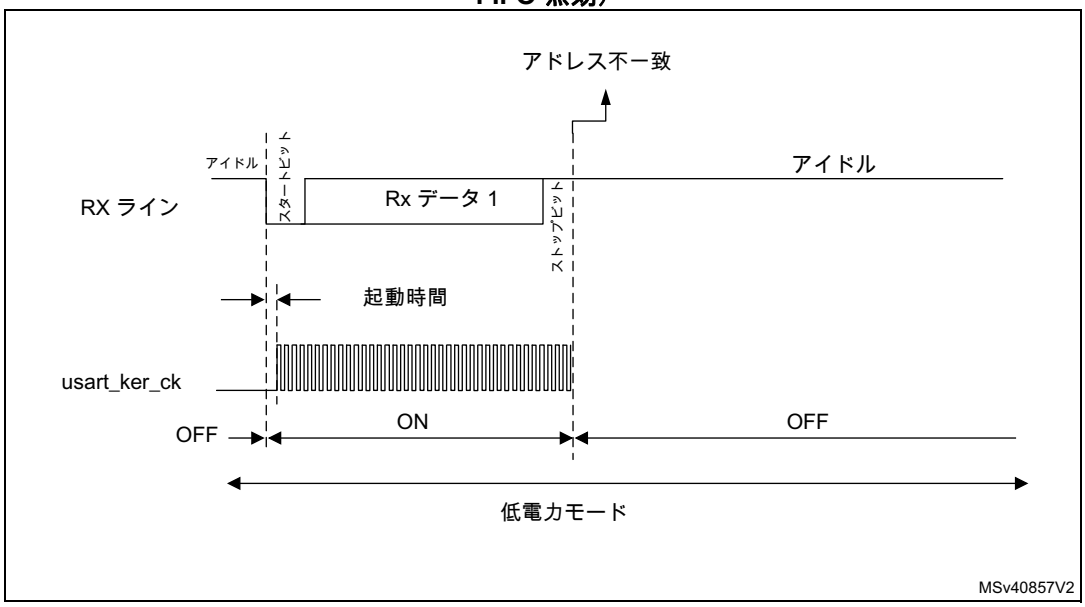


図 557 に、ウェイクアップイベントが確認されない時の USART の動作を示します。

図 557. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： **上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、USART はスタートビットの終端にウェイクアップイベントを MCU に送ります。**

デバイスの低電力モードからの正常なウェイクアップを可能にする最大 USART ボー・レートの決定

デバイスの低電力モードからの正常なウェイクアップを可能にする最大 USART ボー・レートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）と USART レシーバの許容誤差（セクション 37.5.8 : クロック偏差に対する USART レシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、M ビット = "01"、ONEBIT = 0、BRR [3:0] = 0000 とします。

この条件では、表 346 : BRR [3:0] = 0000 のときの USART レシーバの許容誤差によると、USART レシーバの許容誤差は 3.41% です。

DTRA + DQUANT + DREC + DTCL + DWU < USART レシーバの許容誤差

$D_{WUmax} = t_{WUUSART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min} = t_{WUUSART} / (11 \times D_{WUmax})$

ここでは、$t_{WUUSART}$ が低電力モードからのウェイクアップ時間です。

パラメータ DTRA、DQUANT、DREC、および DTCL が 0% であるときに理想的なケースを考えた場合、DWU の最大値は 3.41% です。実際には、最低 usart_ker_ck 精度を考慮する必要があります。

たとえば HSI16 が usart_ker_ck として使用され、HSI16 精度が 1% の場合、以下を得ることができます。

$t_{WUUSART}$ = 3 µs（この値は例にすぎません。正確な値については、デバイスのデータシートを参照してください）

D_{WUmax} = 3.41% - 1% = 2.41%

$T_{bit\ min}$ = 3 µs/ (11 x 2.41%) = 11.32 µs

その結果、低電力モードからの正しいウェイクアップを可能にする最大ボーレートは、次のようになります。1/11.32 µs = 88.36 K ボー

37.6 低電力モードでの USART

表 349. 低電力モードが USART に与える影響

モード	説明
SLEEP	影響はありません。USART 割込みによって、デバイスは SLEEP モードを終了します。
STOP(1)	USART レジスタの内容は保持されます。 STOP モードでも動作しているオシレータで USART のクロックが駆動されていれば、USART でマイクロコントローラを STOP モードからウェイクアップできます。
STANDBY	USART ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

1. 特定のペリフェラル・インスタンスで STOP モードからのウェイクアップがサポートされているかどうかを確認するには セクション 37.4 : USART の実装を参照してください。特定の STOP モードで動作しないインスタンスは、その STOP モードに移行する前に無効化する必要があります。

37.7 USART 割込み

すべての USART 割込みリクエストの詳細な説明については、表 350 を参照してください。

表 350. USART 割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP(1) モードの終了	STANDBY モードの終了
USART または UART	送信データ・レジスタエンプティ	TXE	TXEIE	TDR に書き込む	可能	不可	不可
	送信 FIFO ノット・フル	TXFNF	TXFNFIE	TXFIFO フル		不可	
	送信 FIFO エンプティ	TXFE	TXFEIE	TDR に書き込むか TXFRQ に 1 を書き込む		可能	
	送信 FIFO 閾値到達	TXFT	TXFTIE	TDR に書き込む		可能	
	CTS 割込み	CTSIF	CTSIE	CTSCF に 1 を書き込む		不可	
	送信完了	TC	TCIE	TDR に書き込むか TCCF に 1 を書き込む		不可	
	ガード時間前に送信完了	TCBGT	TCBGTIE	TDR に書き込むか TCBGT に 1 を書き込む		不可	

表 350. USART 割込みリクエスト （続き）

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP(1) モードの終了	STANDBY モードの終了
USART または UART	受信データ·レジスタノット·エンプティ（データの読出し可能）	RXNE	RXNEIE	RDR を読み出すか、RXFRQ に 1 を書き込む	可能	可能	不可
	受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFIFO がエンプティになるまで RDR を読み出すか、RXFRQ に 1 を書き込む		可能	
	受信 FIFO フル	RXFF(2)	RXFFIE	RDR を読み出す		可能	
	受信 FIFO 閾値到達	RXFT	RXFTIE	RDR を読み出す		可能	
	オーバーラン·エラー検出	ORE	RXNEIE/RXFNEIE	ORECF に 1 を書き込む		不可	
	アイドル·ライン検出	IDLE	IDLEIE	IDLECF に 1 を書き込む		不可	
	パリティ·エラー	PE	PEIE	PECF に 1 を書き込む		不可	
	LIN ブレーク	LBDF	LBDIE	LBDCF に 1 を書き込む		不可	
	マルチバッファ通信のノイズ·エラー	NE	EIE	NFCF に 1 を書き込む		不可	
	マルチバッファ通信のオーバーラン·エラー	ORE(3)		ORECF に 1 を書き込む		不可	
	マルチバッファ通信のフレーミング·エラー	FE		FECF に 1 を書き込む		不可	
	キャラクター致	CMF	CMIE	CMCF に 1 を書き込む		不可	
	レシーバ·タイムアウト	RTOF	RTOFIE	RTOCCF に 1 を書き込む		不可	
	ブロックの終了	EOBF	EOBIE	EOBCF に 1 を書き込む		不可	
	低電力モードからのウェイクアップ	WUF	WUFIE	WUC に 1 を書き込む		可能	
	SPI スレーブのアンダーラン·エラー	UDR	EIE	UDRCF に 1 を書き込む		不可	

1. ペリフェラル·インスタンスが STOP モードからのウェイクアップ機能をサポートしている場合にのみ、USART はデバイスを STOP モードからウェイクアップすることができます。サポート対象の STOP モードのリストについては、セクション 37.4 : USART の実装を参照してください。
2. RXFF フラグは、USART が RXFIFO に n 個のデータを置き、USART_RDR に 1 個のデータを置くように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。STOP モードでは、USART_RDR はクロック供給されません。その結果として、このレジスタは書き込まれず、n 個のデータが受信されて RXFIFO に書き込まれた後に、RXFF 割込みがアサートされます（RXFF フラグはセットされません）。
3. OVRDIS = 0 の場合

37.8 USART レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル・レジスタには、ワード（32 ビット）単位でアクセスする必要があります。

37.8.1 USART 制御レジスタ 1[オルタネート]（USART_CR1）

アドレス・オフセット：0x00

リセット値：0x0000 0000

FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも、同じレジスタを使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの RXFF が 1 になると、USART 割込みが生成されます。

ビット 30 **TXFEIE**：TXFIFO エンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TXFE が 1 になると、USART 割込みが生成されます。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： FIFO モードは、SPI マスタ／スレーブ・モードおよびスマートカード・モードでのみ、標準 UART 通信で使用できます。IrDA および LIN のモードでは有効にしてはいけません。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビット・データ長モードでは、スマートカード・モード、LIN マスタ・モード、および自動ボー・レート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 26 **RTOIE**：レシーバ･タイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの RTOF ビットがセットされると、USART 割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4:USART の実装を参照してください。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカード･モードでは、このビットは常にクリア状態に保つ必要があります。

ビット 14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割込みが生成されます。

ビット 13 **MME**：ミュート･モード有効

このビットは USART ミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバ･ウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M = 1 の場合は 9 番目のビット、M = 0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの PE が 1 になると USART 割込みが生成されます。

ビット 7 **TXFNFIE**：TXFIFO ノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TXFNF=1 のときには、USART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TC が 1 になると USART 割込みが生成されます。

ビット 5 **RXFNEIE**：RXFIFO ノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの ORE が 1 または RXFNE が 1 になると USART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの IDLE が 1 になると USART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に“1”を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：低電力モードの USART 有効

このビットがクリアされると、USART は MCU を 低電力モードからウェイクアップできません。

このビットがセットされると、USART は MCU を 低電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は低電力モードから MCU をウェイクアップできません。

1：USART は低電力モードから MCU をウェイクアップできます。

注： 低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ラインにエラーが発生することなく低電力モードに移行するには、TE ビットを事前にリセットし、USART_ISR の TC ビットがセットされるのを待ってからソフトウェアで UE ビットをリセットする必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

スマートカードモードでは（SCEN = 1）、CK は UE ビット値にかかわらず、CLKEN = 1 の場合に常に使用可能です。

37.8.2 USART 制御レジスタ 1[オルタネート]（USART_CR1）

アドレス･オフセット：0x00

リセット値：0x0000 0000

FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも、同じレジスタを使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]				
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： FIFO モードは、SPI マスタ／スレーブ･モードおよびスマートカード･モードでのみ、標準 UART 通信で使用できます。IrDA および LIN のモードでは有効にしてはいけません。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビット･データ長モードでは、スマートカード･モード、LIN マスタ･モード、および自動ボー･レート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27 **EOBIE**：ブロック終了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの EOBF フラグがセットされると、USART 割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 26 **RTOIE**：レシーバ･タイムアウト割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの RTOF ビットがセットされると、USART 割込みが生成されます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4:USART の実装を参照してください。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。サンプル時間単位（オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間）で表されます。

DEDT 時間中に USART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 15 **OVER8**：オーバーサンプリングモード

0：16 倍のオーバーサンプリング

1：8 倍のオーバーサンプリング

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： LIN、IrDA、およびスマートカード･モードでは、このビットは常にクリア状態に保つ必要があります。

ビット 14 **CMIE**：キャラクター致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの CMF ビットがセットされると、USART 割込みが生成されます。

ビット 13 **MME**：ミュート･モード有効

このビットは USART ミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバ･ウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M = 1 の場合は 9 番目のビット、M = 0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの PE が 1 になると USART 割込みが生成されます。

ビット 7 **TXEIE**：送信データ･レジスタ･エンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TXE=1 のときには、USART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの TC が 1 になると USART 割込みが生成されます。

ビット 5 **RXNEIE**：受信データ･レジスタ･ノット･エンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの ORE が 1 または RXNE が 1 になると USART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの IDLE が 1 になると USART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： スマートカードモードの場合を除いて、送信中に TE ビットにローパルスを与える（“0”に続けて“1”を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に“1”を書き込まないでください。必要な時間を確保するために、ソフトウェアは USART_ISR レジスタの TEACK ビットをポーリングできます。

スマートカードモードでは、TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：低電力モードの USART 有効

このビットがクリアされると、USART は MCU を 低電力モードからウェイクアップできません。

このビットがセットされると、USART は MCU を 低電力モードからウェイクアップできます。

このビットは、ソフトウェアによってセット／クリアされます。

0：USART は低電力モードから MCU をウェイクアップできません。

1：USART は低電力モードから MCU をウェイクアップできます。

注： 低電力モードに入る直前に UESM ビットをセットし、低電力モードの終了時にクリアすることが推奨されます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 0 **UE**：USART 有効

このビットがクリアされると、USART プリスケーラと出力はただちに停止され、現在のすべての動作は破棄されます。USART の設定は保たれますが、USART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：USART プリスケーラと出力は無効であり、低電力モードです。

1：USART は有効です。

注： ラインにエラーが発生することなく低電力モードに移行するには、TE ビットを事前にリセットし、USART_ISR の TC ビットがセットされるのを待ってからソフトウェアで UE ビットをリセットする必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

スマートカード･モードでは（SCEN = 1）、UE ビット値にかかわらず、CLKEN = 1 であればいつでも CK ピンを使用できます。

37.8.3 USART 制御レジスタ 2（USART_CR2）

アドレス･オフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw			rw

ビット 31:24 **ADD[7:0]**：USART ノードのアドレス

これらのビットは、ミュート･モードでは USART ノードのアドレス、低電力モードと RUN モードでは認識対象となるキャラクタ･コードを表します。

- ミュート･モード：マルチプロセッサ通信で、4 ビットまたは7 ビットのアドレス･マーク検出によってミュート･モードからウェイクアップするときに使用します。トランスミッタで送信するキャラクタの MSB は 1 であることが必要です。4 ビットのアドレス･マーク検出では、ADD[3:0] ビットのみを使用します。
- 低電力モード：キャラクター致によって低電力モードからウェイクアップするときに使用します。WUS[1:0] を 0b00 にプログラムしている場合（アドレス一致時に WUF をアクティブにする設定）、ADD[6:0] または ADD[3:0] ビットフィールド（どちらであるかは ADDM7 ビットで指定）にプログラムしたキャラクタと受信キャラクタが一致していて、WUFIE ビットのセットによって WUF 割込みが有効になっていれば、低電力モードからのウェイクアップが実行されます。トランスミッタで送信するキャラクタの MSB は 1 であることが必要です。
- ミュート･モードを無効にした RUN モード（ModBus プロトコルでのブロック終端検出など）：受信キャラクタ全体（8 ビット）を ADD[7:0] と比較し、一致していれば CMF フラグがセットされます。CMIE ビットがセットされている場合、割込みが生成されます。

これらのビットフィールドは、受信が無効のとき（RE = 0）または USART が無効のとき（UE = 0）にのみ書き込むことができます。

ビット 23 **RTOEN**：レシーバ･タイムアウト有効

このビットは、ソフトウェアによってセット／クリアされます。

0：レシーバタイムアウト機能は無効です。

1：レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR（レシーバタイムアウトレジスタ）でプログラムされた時間にわたって RX ラインがアイドル（受信なし）であった場合、USART_ISR レジスタの RTOF フラグがセットされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 22:21 **ABRMOD[1:0]**：自動ボーレートモード

これらのビットは、ソフトウェアによってセット／クリアされます。

00：スタートビットの測定がボーレートの検出に使用されます。

01：立ち下がりエッジから立ち下がりエッジまでの測定（受信されたフレームはシングル･ビット = 1 で始まらなければならず、その場合、フレーム = Start10xxxxxx）

10：0x7F フレーム検出

11：0x55 フレーム検出

このビットフィールドは、ABREN = 0 または USART が無効（UE = 0）のときにのみ書き込むことができます。

注： DATAINV = 1 および／または MSBFIRST = 1 の場合、パターンはライン上で同じである必要があります（たとえば、MSBFIRST の場合は 0xAA）。

USART が自動ボー･レート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 20 **ABREN**：自動ボー･レート有効

このビットは、ソフトウェアによってセット／クリアされます。

0：自動ボーレート検出は無効です。

1：自動ボーレート検出は有効です。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装参照してください。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1 = H、0 = L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1 = L、0 = H）パリティ･ビットも反転されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピン･アクティブ･レベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジック･レベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：TX ピン信号の値を反転します（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピン･アクティブ･レベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジック･レベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：RX ピン信号の値を反転します（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の UART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 14 **LINEN**：LIN モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：LIN モードを無効化します。

1：LIN モードを有効化します。

LIN モードでは、USART_CR1 レジスタの SBKRQ ビットを使用して LIN 同期ブレーク（13 のロー･ビット）を送信でき、また LIN 同期ブレークを検出できます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：0.5 個のストップビット

10：2 個のストップビット

11：1.5 個のストップビット

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **CLKEN**：クロック有効

このビットによって、CK ピンを有効にできます。

0：CK ピンは無効です。

1：CK ピンは有効です。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 同期モードとスマートカード･モードのどちらもサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

スマートカードモードでは、スマートカードに CK クロックを正しく供給するには、次のステップを順守する必要があります。
UE = 0
SCEN = 1
GTPR 設定
CLKEN= 1
UE = 1

ビット 10 **CPOL**：クロック極性

このビットによって、同期モードにおける CK ピンのクロック出力の極性を選択できます。CPHA ビットと連携して動作し、希望するクロック／データ関係になるようにします。

0：送信ウィンドウの外で、CK ピンはロー･レベルを維持します。

1：送信ウィンドウの外で、CK ピンはハイ･レベルを維持します。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 9 **CPHA**：クロック位相

このビットは、同期モードでの CK ピンのクロック出力の位相を選択するために使用されます。CPOL ビットと連携して動作し、希望するクロック／データ関係になるようにします（図 537 および図 538 を参照）。

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 8 **LBCL**：最終ビットのクロック･パルス

このビットは、同期モードで送信する最終データ･ビット（MSB）に対応するクロック･パルスを CK ピンに出力する必要があるかどうかを選択するために使用します。

0：最終データ･ビットのクロック･パルスは CK ピンに出力されません。

1：最終データ･ビットのクロック･パルスが CK ピンに出力されます。

注意： 最終ビットは、USART_CR1 レジスタの M ビットによって選択された 7 または 8 または 9 ビットフォーマットに応じて送信された 7 番目または 8 番目または 9 番目のデータビットです。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LBDIE**：LIN ブレーク検出割込み有効

ブレーク割込みマスクです（ブレークデリミタを使用したブレーク検出）。

0：割込みは禁止されています。

1：USART_ISR レジスタの LBDF が 1 になると割込みが生成されます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 5 **LBDL**：LIN ブレーク検出長

このビットでは、10 ビットと 11 ビットのブレーク検出を選択します。

0：10 ビットブレーク検出

1：11 ビットブレーク検出

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビットおよび 9 ビット･データ･モードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットのアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3 **DIS_NSS**：

DIS_NSS ビットがセットされているとき、NSS ピンの入力は無視されます。

0：SPI スレーブ選択は NSS 入力ピンに依存します。

1：SPI スレーブが常に選択され、NSS 入力ピンは無視されます。

注： SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SLVEN**：同期スレーブ･モード有効

SLVEN ビットがセットされると、同期スレーブモードが有効になります。

0：スレーブモードは無効です。

1：スレーブモードは有効です。

注： SPI スレーブモードがサポートされていないときは、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

注： トランスミッタが有効なときには、CPOL、CPHA、LBCL のビットに書き込まないでください。

37.8.4 USART 制御レジスタ 3（USART_CR3）

アドレス･オフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXF TIE	RXFTCFG[2:0]			TCBG TIE	TXFTIE	WUFIE	WUS[1:0]		SCARCNT[2:0]			Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVR DIS	ONE BIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HD SEL	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 **TXFTCFG[2:0]**：TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
010：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済みです。

ビット 28 **RXFTIE**：RXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、USART 割込みが生成されます。

ビット 27:25 **RXFTCFG[2:0]**：受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
010：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済みです。

ビット 24 **TCBGTIE**：ガード時間前送信完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：USART_ISR レジスタの TCBGT=1 のときには、USART 割込みが生成されます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 23 **TXFTIE**：TXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、USART 割込みが生成されます。

ビット 22 **WUFIE**：低電力モードからのウェイクアップ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：USART_ISR レジスタの WUF が 1 になると USART 割込みが生成されます。

注： WUFIE は、低電力モードに入る前にセットする必要があります。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 21:20 **WUS[1:0]**：低電力モードからのウェイクアップ割込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低電力モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み。

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE/RXFNE 時に有効になります。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 19:17 **SCARCNT[2:0]**：スマートカード自動再試行カウント

このビットフィールドは、スマートカードモードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します（FE ビットをセット）。

受信モードでは、受信エラーが生成される（RXNE/RXFNE および PE ビットのセット）までの受信の試行エラー回数を指定します。

このビットフィールドは、USART が無効（UE = 0）のときのみプログラムする必要があります。

USART が有効の場合（UE = 1）、このビットフィールドには再送信を停止する目的で 0x0 のみを書き込むことができます。

0x0：再送信無効 - 送信モードでの自動再送信禁止

0x1 から 0x7：自動再送信試行回数（信号エラーの生成前）

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ドライバ有効機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます（スマートカードモードで使用される）。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアはまず最初にDMA リクエストを無効にするか（DMAR = 0）、FIFO モードが有効な場合、RXNE/RXFNEをクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USART が無効（UE=0）のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが USART_RDR レジスタの前の内容に上書きされます。FIFO モードが有効なとき、RXFIFO はバイパスされ、データは直接、USART_RDR レジスタに書き込まれます。FIFO 管理が有効になっているときでも、RXNE フラグが使用されます。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 **ONEBIT**：1 サンプル･ビット方式有効

このビットによって、サンプル方式を選択できます。1 サンプルビット方式が選択されると、ノイズ検出フラグ（NE）が無効になります。

0：3 サンプルビット方式

1：1 サンプルビット方式

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **CTSIE**：CTS 割込み有効

0：割込みは禁止されています。

1：USART_ISR レジスタの CTSIF が 1 になると割込みが生成されます。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 9 **CTSE**：CTS 有効化

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは CTS 入力がデアサート（0 固定）されている場合にのみ転送されます。データの送信中に CTS 入力がアサートされた場合、送信が完了してから停止します。CTS がアサートされている間にデータがデータ･レジスタに書き込まれた場合、CTS がデアサートされるまで送信は延期されます。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 8 **RTSE**：RTS 有効化

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS 出力がデアサートされます（0 にプルされます）。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5 **SCEN**：スマートカード･モード有効

このビットはスマートカードモードを有効にするために使用します。

0：スマートカードモードが無効です。

1：スマートカードモードが有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART がスマートカード･モードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 4 **NACK**：スマートカード NACK 有効

0：パリティエラーの際の NACK 転送が無効です。

1：パリティエラー時の NACK 転送が有効です。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： USART がスマートカード･モードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

ビット 2 **IRLP**：IrDA 低電力

このビットは、通常と低電力の IrDA モードの選択に使用されます。

0：通常モード

1：低電力モード

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 1 **IREN**：IrDA モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：IrDA は無効です。

1：IrDA は有効です。

このビットは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： IrDA モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 0 **EIE**：エラー割込み有効化

エラー割込み有効化ビットは、フレーミング･エラー、オーバーラン･エラー、ノイズ･フラグ、または SPI スレーブ･アンダーラン･エラー（USART_ISR レジスタで FE = 1、ORE = 1、NE = 1、または UDR = 1）の場合に、割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：SPI スレーブ･モードで USART_ISR レジスタの FE、ORE、NE、または UDR が 1 になると割込みが生成されます。

37.8.5 USART ボー･レート･レジスタ（USART_BRR）

このレジスタは、USART が無効（UE = 0）のときのみ書き込むことができます。自動ボー･レート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **BRR[15:0]**：USART ボーレート

BRR[15:4]

BRR[15:4] = USARTDIV[15:4]

BRR[3:0]

OVER8 = 0 のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = 1 のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。

BRR[3] は、クリアされたままにする必要があります。

37.8.6 USART ガード時間およびプリスケーラレジスタ（USART_GTPR）

アドレスオフセット：0x10

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								PSC[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **GT[7:0]**：ガード時間値

このビットフィールドは、ガード時間値をボークロック周期数でプログラムするために使用します。
これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。
このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： スマートカード･モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 7:0 **PSC[7:0]**：プリスケーラ値

IrDA 低電力および IrDA 通常モード：

PSC[7:0] = IrDA 通常および低電力ボーレート

PSC[7:0] は、USART ソースクロックを分周して低電力周波数を得るためのプリスケーラのプログラミングに使用します。ソースクロックは、レジスタで指定された値（上位 8bit）で分周されます。

スマートカードモード：

PSC[4:0] = プリスケーラ値

PSC[4:0] は、USART ソースクロックを分周してスマートカードのクロックを提供するプリスケーラのプログラミングに使用します。レジスタで指定された値（上位 5 ビット）を 2 倍して、クロックソース周波数の分周比を求めます。

00000：予約済み - この値はプログラミングしないでください。
00001：ソース･クロックは 1 分周（IrDA モード）または 2 分周（スマートカード･モード）されます。
00010：ソース･クロックは 2 分周（IrDA モード）または 4 分周（スマートカード･モード）されます。
00011：ソース･クロックは 3 分周（IrDA モード）または 6 分周（スマートカード･モード）されます。
.....
11111：ソース･クロックは 31 分周（IrDA モード）または 62 分周（スマートカード･モード）されます。
0010 0000：ソース･クロックは 32 分周（IrDA モード）されます。
.....
1111 1111：ソース･クロックは 255 分周（IrDA モード）されます。

このビットフィールドは、USART が無効（UE = 0）のときのみ書き込むことができます。

注： スマートカード･モードを使用する場合、ビット [7:5] はクリアしたままにする必要があります。
スマートカードモードや IrDA モードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に“0”に設定されます。1564 ページのセクション 37.4：USART の実装を参照してください。

37.8.7 USART レシーバタイムアウトレジスタ（USART_RTOR）

アドレスオフセット：0x14

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLEN[7:0]								RTO[23:16]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTO[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 **BLEN[7:0]**：ブロック長

このビットフィールドは、スマートカード T=1 受信のブロック長を指定します。この値は、情報文字の数 + エピローグ･フィールドの長さ（1-LEC/2-CRC）- 1 と等しくなります。

例：

BLEN = 0：0 情報文字 + LEC

BLEN = 1：0 情報文字 + CRC

BLEN = 255：254 情報文字 + CRC（合計 256 文字）

スマートカード･モードでは、TXE = 0（FIFO モードが有効な場合は TXFE = 0）のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。その場合、RE = 0（レシーバ無効）のとき、および/または EOBCF ビットが 1 に書き込まれたときに、ブロック長カウンタがリセットされます。

注： この値は、ブロック受信の開始後にプログラムできます（プロローグ･フィールドの LEN 文字のデータを使用）。必ず受信したブロックにつき一度だけプログラムするようにしてください。

ビット 23:0 **RTO[23:0]**：レシーバタイムアウト値

このビットフィールドは、RX ラインでアクティビティがない間のビット数でレシーバタイムアウト値を指定します。

標準モードでは、最後の受信キャラクタの後、RTO 値を超える間、新しいスタートビットが検出されなかった場合、RTOF フラグがセットされます。

スマートカードモードでは、この値は CWT および BWT を実装するために使用されます。詳細については、スマートカードの章を参照してください。この標準では、CWT/BWT 測定は最後の受信キャラクタのスタートビットから開始して行われます。

注： この値は、受信キャラクタごとにプログラムされる必要があります。

注： RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって 0x00000000 に強制的に設定されます。1564 ページのセクション 37.4：USART の実装を参照してください。

37.8.8 USART リクエストレジスタ（USART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
											w	w	w	w	w

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

FIFO モードが無効になっている場合は、このビットに“1”を書き込むと、TXE フラグがセットされます。これにより、送信データを破棄できます。このビットは、エラー（NACK）によりデータが送信されなかった場合、および USART_ISR レジスタで FE フラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。

FIFO モードが有効な場合、FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（送信 FIFO エンプティ、USART_ISR レジスタのビット 23）がセットされます。送信 FIFO の一掃は、UART モードおよびスマートカードモードの両方でサポートされています。

注： FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、受信 FIFO 全体を空にします（すなわち、RXFNE ビットをクリアします）。これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュート･モード･リクエスト

このビットに 1 を書き込むと、USART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 **ABRRQ**：自動ボー･レート･リクエスト

このビットに 1 を書き込むと、USART_ISR の ABRF と ABRE フラグがリセットされ、次の受信データ･フレームでの自動ボー･レート測定をリクエストします。

注： USART が自動ボー･レート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4 : USART の実装を参照してください。

37.8.9 USART 割込みおよびステータスレジスタ [オルタネート]（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0X80 00C0

X = 2（FIFO／スマートカードモードが有効の場合）

X = 0（FIFO が有効で、スマートカードモードが無効の場合）

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
				r	r	r	r	r	r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは､TXFIFO が USART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。USART_CR3 レジスタの TXFTIE ビット が 1（ビット 31）の場合、割込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、USART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したときに、ハードウェアによってセットされます。これは、受信 FIFO に（RXFTCFG - 1）個のデータがあり、USART_RDR レジスタに 1 個のデータがあることを意味します。USART_CR3 レジスタの RXFTIE ビット が 1（ビット 27）の場合、割込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

注： RXFTCFG 閾値が“101”に設定されているとき、16個のデータが利用可能になった場合（すなわち、RXFIFO に 15個のデータが入り、USART_RDR に 1個のデータが入った場合)、RXFT フラグがセットされます。したがって、17番目の受信データによってオーバーランエラーが起きることはありません。オーバーランエラーは 18 番目のデータを受信した後に発生します。

ビット 25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードが NACK を返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3 レジスタの TCBGTIE が 1 の場合、割込みが生成されます。

このビットは、USART_ICR レジスタの TCBGTCF に 1 を書き込むか、USART_TDR レジスタに書き込むことによってソフトウェアでクリアされます。

0 : 送信が完了していないか、または送信が正常に完了していません（つまり、カードから NACK を受信）。

1 : 送信は正常に完了しました（ガード時間完了前で、スマートカードから NACK なし）。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USART がスマートカードモードをサポートしている場合でこのモードが有効な場合、TCBGT のリセット値は“1”です。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 24 **RXFF** : RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、USART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

USART_CR1 レジスタの RXFFIE ビット が 1 の場合、割込みが生成されます。

0 : RXFIFO はフルではありません。

1 : RXFIFO はフルです。

ビット 23 **TXFE** : TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、USART_RQR レジスタのビット TXFRQ（ビット 4）に 1 を書き込むことによってセットすることもできます。

USART_CR1 レジスタの TXFEIE ビット が 1（ビット 30）の場合、割込みが生成されます。

0 : TXFIFO ノット･エンプティ。

1 : TXFIFO エンプティ。

ビット 22 **REACK** : 受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、USART が受信できる状態であることを確認できます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 21 **TEACK** : 送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

USART_CR1 レジスタで TE= 0 を書き込んだ後、TE = 1 を書き込むことによってアイドル･フレーム･リクエストを生成するときの TE = 0 の最小期間を確保する目的で使用できます。

ビット 20 **WUF** : 低電力モードからのウェイクアップ･フラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE が 1 の場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 19 **RWU**：レシーバのミュート･モードからのウェイクアップ･フラグ

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されます。

1：USART_RQR レジスタの SBKRQ ビットをセットすることでブレーク･キャラクタをリクエストします。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE が 1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジー･フラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボー･レート･フラグ

このビットは、自動ボー･レートが設定された場合（RXFNE もセットされ RXFNEIE = 1 の場合、割込みが生成されます）または自動ボー･レート動作が成功せずに完了したとき（ABRE = 1）（この場合、ABRE、RXFNE、FE もセットされます）にハードウェアによってセットされます。

新しい自動ボー･レート検出をリクエストするために、USART_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアでクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 14 **ABRE**：自動ボー･レート･エラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_RQR レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアでクリアされます。

注： USART が自動ボー･レート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 13 **UDR**：SPI スレーブ･アンダーラン･エラー･フラグ

スレーブ送信モードでは、ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICR レジスタの UDRCF ビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、ブロック全体が受信されたときに、ハードウェアによってセットされます（たとえば、T = 1 スマートカード･モード）。受信バイト数（ブロックの開始からプロローグも含む値）が BLEN + 4 以上である場合に、この状態が検出されます。

USART_CR1 レジスタの EOBIE が 1 の場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 11 **RTOF**：レシーバ･タイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの RTOIE が 1 の場合、割込みが生成されます。

スマートカード･モードでは、このタイムアウトは CWT または BWT タイミングに相当します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの CTSIE が 1 の場合、割込みが生成されます。

0：CTS ステータス･ラインでの変化はありません。

1：CTS ステータス･ラインに変化がありました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 **LBDF** : LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE が 1 である場合、割込みが生成されます。

0 : LIN ブレークは検出されませんでした。

1 : LIN ブレークが検出されました。

注 : USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 7 **TXFNF** : TXFIFO ノット･フル

TXFIFO がフルではない、つまり USART_TDR にデータを書き込めるとき、TXFNF はハードウェアによってセットされます。USART_TDR レジスタへの書込み動作ごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを USART_TDR に書き込むことができないことを示します。

USART_CR1 レジスタの TXFNFIE ビット =1 の場合、割込みが生成されます。

0 : 送信 FIFO はフルです。

1 : 送信 FIFO ノットフル

注 : 一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです（TXFNF および TXFE は同時にセットされます）。

このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC** : 送信完了

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから送信されたことを示します。

データを含むフレームの送信が完了し、TXFE がセットされたとき、ハードウェアによってセットされます。

USART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。

TC ビットは、USART_ICR レジスタの TCCF に 1 を書き込むか、USART_TDR レジスタに書き込むことによって、ソフトウェアでクリアされます。

0 : 送信は完了していません。

1 : 送信は完了しています。

注 : TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXFNE** : RXFIFO ノット･フル

RXFNE ビットは、RXFIFO が空でないとき、つまりデータが USART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。USART_RDR からの読出し動作のたびに、RXFIFO の 1 つの場所が解放されます。

RXFIFO が空になると、RXFNE がクリアされます。RXFNE フラグは、USART_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXFNEIE が 1 の場合、割込みが生成されます。

0 : データは受信されていません。

1 : 受信データを読み出すことができます。

ビット 4 **IDLE** : アイドル･ライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE が 1 の場合、割込みが生成されます。USART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0 : アイドルラインは検出されていません。

1 : アイドルラインが検出されました。

注 : RXFNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュート･モードが有効な場合（MME = 1）、USART がミュートでなければ（RWU = 0）、WAKE ビットで選択したミュート･モードに関係なく、IDLE がセットされます。RWU = 1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーラン･エラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、現在シフトレジスタに受信中のデータを USART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの RXFNEIE が 1 または EIE が 1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXFNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NE フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（1581 ページのセクション 37.5.8：クロック偏差に対する USART レシーバの許容誤差を参照）。

このエラーはUSART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミング･エラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはUSART_RDR 内のキャラクタに関連します。

ビット 0 **PE**：パリティ･エラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： このエラーはUSART_RDR 内のキャラクタに関連します。

37.8.10 USART 割込みおよびステータス･レジスタ [オルタネート]（USART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	RE ACK	TE ACK	WUF	RWU	SBKF	CMF	BUSY
						r			r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **TCBGT**：ガード時間前送信完了フラグ

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから正しく送信されたときにセットされます。

スマートカードモードで、データを含むフレームの送信が完了し、スマートカードが NACK を返信しなかった場合、このビットはハードウェアによってセットされます。USART_CR3 レジスタの TCBGTIE が 1 の場合、割込みが生成されます。

このビットは、USART_ICR レジスタの TCBGTCF に 1 を書き込むか、USART_TDR レジスタに書き込むことによってソフトウェアでクリアされます。

0：送信が完了していないか、または送信が正常に完了していません（つまり、カードから NACK を受信）。

1：送信は正常に完了しました（ガード時間完了前で、スマートカードから NACK なし）。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。USART がスマートカードモードをサポートしている場合でこのモードが有効な場合、TCBGT のリセット値は“1”です。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 24:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低電力モードに入る前に、USART が受信できる状態であることを確認できます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット／リセットされます。

USART_CR1 レジスタで TE= 0 を書き込んだ後、TE = 1 を書き込むことによってアイドル･フレーム･リクエストを生成するときの TE = 0 の最小期間を確保する目的で使用できます。

ビット 20 **WUF**：低電力モードからのウェイクアップ･フラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。USART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの WUFIE が 1 の場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 19 **RWU**：レシーバのミュート･モードからのウェイクアップ･フラグ

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、USART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4 : USART の実装を参照してください。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されます。

1：USART_RQR レジスタの SBKRQ ビットをセットすることでブレーク･キャラクタをリクエストします。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの CMIE が 1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジー･フラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：USART はアイドルです（受信なし）。

1：受信中です。

ビット 15 **ABRF**：自動ボー･レート･フラグ

このビットは、自動ボー･レートが設定された場合（RXNE もセットされ RXNEIE = 1 の場合、割込みが生成されます）または自動ボー･レート動作が成功せずに完了したとき（ABRE = 1）（この場合、ABRE、RXNE、FE もセットされます）にハードウェアによってセットされます。

新しい自動ボー･レート検出をリクエストするために、USART_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアでクリアされます。

注： USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 14 **ABRE**：自動ボー･レート･エラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USART_RQR レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアでクリアされます。

注： USART が自動ボー･レート機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 13 **UDR**：SPI スレーブ･アンダーラン･エラー･フラグ

スレーブ送信モードでは、ソフトウェアが USART_TDR にまだ値をロードしていない間に、データ送信用の最初のクロックパルスが現われると、このフラグがセットされます。このフラグは、USART_ICR レジスタの UDRCF ビットをセットすることによってリセットされます。

0：アンダーランエラーはありません。

1：アンダーランエラーが発生しました。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 12 **EOBF**：ブロック終了フラグ

このビットは、ブロック全体が受信されたときに、ハードウェアによってセットされます（たとえば、T = 1 スマートカード･モード）。受信バイト数（ブロックの開始からプロローグも含む値）が BLEN + 4 以上である場合に、この状態が検出されます。

USART_CR1 レジスタの EOBIE が 1 の場合、割込みが生成されます。

USART_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0：ブロック終了に達していません。

1：ブロック終了（文字数）に達しました。

注： スマートカードモードがサポートされない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 11 **RTOF**：レシーバ･タイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USART_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの RTOIE が 1 の場合、割込みが生成されます。

スマートカード･モードでは、このタイムアウトは CWT または BWT タイミングに相当します。

0：タイムアウト値に達していません。

1：データを受信せずにタイムアウト値に達しました。

注： 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間（オーバーサンプリング方式によって 2/16 または 2/8）を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。USART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR3 レジスタの CTSIE が 1 の場合、割込みが生成されます。

0：CTS ステータス･ラインでの変化はありません。

1：CTS ステータス･ラインに変化がありました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 **LBDF**：LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USART_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR2 レジスタの LBDIE が 1 である場合、割込みが生成されます。

0：LIN ブレークは検出されませんでした。

1：LIN ブレークが検出されました。

注： USART が LIN モードをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 7 **TXE**：送信データ･レジスタ･エンプティ割込み有効

このビットは、USART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。このビットは、USART_TDR レジスタへの書込みによってクリアされます。送信に失敗した場合は、スマートカード T = 0 モードでのみ、データを破棄するために USART_RQR レジスタの TXFRQ に 1 を書き込むことで TXE フラグがセットされます。

USART_CR1 レジスタの TXEIE ビット が 1 の場合、割込みが生成されます。

0：データレジスタはフルです。

1：データレジスタはフルではありません。

ビット 6 **TC**：送信完了

このビットは、USART_TDR に書き込まれた最後のデータがシフトレジスタから送信されたことを示します。

データを含むフレームの送信が完了し、TXE がセットされたとき、ハードウェアによってセットされます。

USART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。

TC ビットは、USART_ICR レジスタの TCCF に 1 を書き込むか、USART_TDR レジスタに書き込むことによって、ソフトウェアでクリアされます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読出しデータ･レジスタ･ノット･エンプティ

RXNE ビットは、USART_RDR シフトレジスタの内容が USART_RDR レジスタに転送されると、ハードウェアによってセットされます。これは、USART_RDR レジスタからの読出しによってクリアされます。RXNE フラグは、USART_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USART_CR1 レジスタの RXNEIE が 1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドル･ライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの IDLEIE が 1 の場合、割込みが生成されます。USART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュート･モードが有効な場合（MME = 1）、USART がミュートでなければ（RWU = 0）、WAKE ビットで選択したミュート･モードに関係なく、IDLE がセットされます。RWU = 1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーラン･エラー

このビットは、RXNE="1"（FIFO モードが有効な場合は RXFF="1"）のときに、

現在シフト･レジスタで受信中のデータを USART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアでクリアされます。

USART_CR1 レジスタの RXNEIE が 1 または EIE が 1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、USART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

USART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：ノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ラインがノイズフリーであるとき、NE フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます（1581 ページのセクション 37.5.8：クロック偏差に対する USART レシーバの許容誤差を参照）。

ビット 1 **FE**：フレーミング･エラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

USART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティ･エラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

37.8.11 USART 割込みフラグクリアレジスタ（USART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.
											w			w	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
		w	w	w		w	w	w	w	w	w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低電力モードからのウェイクアップ･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの WUF フラグがクリアされます。

注： USART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **UDRCF**：SPI スレーブアンダーランフラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの UDRF フラグがクリアされます。

注： USART が SPI スレーブモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 12 **EOBCF**：ブロック終了フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの EOBF フラグがクリアされます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 11 **RTOCF**：レシーバ･タイムアウト･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの RTOF フラグがクリアされます。

注： USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの CTSIF フラグがクリアされます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 8 **LBDCF**：LIN ブレーク検出フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの LBDF フラグがクリアされます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、リセット値に保持する必要があります。1564 ページのセクション 37.4：USART の実装を参照してください。

ビット 7 **TCBGTCF**：ガード時間前送信完了フラグクリア

このビットに 1 を書き込むと、USART_ISR レジスタの TCBGT フラグがクリアされます。

ビット 6 **TCCF**：送信完了フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの TC フラグがクリアされます。

ビット 5 **TXFECF**：TXFIFO エンプティ･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの TXFE フラグがクリアされます。

ビット 4 **IDLECF**：アイドル･ライン検出フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーラン･エラー･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミング･エラー･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティ･エラー･フラグのクリア

このビットに 1 を書き込むと、USART_ISR レジスタの PE フラグがクリアされます。

37.8.12 USART レシーバデータレジスタ（USART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図 531 を参照）。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

37.8.13 USART トランスミッタデータレジスタ（USART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

USART_TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図 531 を参照）。

パリティを有効にして（USART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF = 1 のときのみ書き込む必要があります。

37.8.14 USART プリスケーラ･レジスタ（USART_PRESC）

このレジスタは、USART が無効（UE = 0）のときのみ書き込むことができます。

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

USART 入力クロックは、以下のようにプリスケーラ分周比によって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済みです。

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は 1011 になります。すなわち、入力クロックは 256分周されます。

37.8.15 USART レジスタマップ

次の表に、USART のレジスタマップとリセット値を示します。

表 351. USART レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	USART_CR1 FIFO 有効	RXFFIE	TXFEIE	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	USART_CR1 FIFO 無効	Res.	Res.	FIFOEN	M1	EOBIE	RTOIE	DEAT[4:0]					DEDT[4:0]					OVER8	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	USART_CR2	ADD[7:0]								RTOEN	ABRMOD[1:0]		ABREN	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	Res.	LBDIE	LBDL	ADDM7	DIS_NSS	Res.	Res.	SLVEN
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0			0
0x08	USART_CR3	TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			TCBGTIE	TXFTIE	WUFIE	WUS [1:0]		SCAR CNT[2:0]			Res.	DEP	DEM	DDRE	OVRDIS	ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	USART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	USART_GTPR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	GT[7:0]								PSC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	USART_RTOR	BLEN[7:0]								RTO[23:0]																							
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	USART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	ABRRQ
	リセット値																												0	0	0	0	0
0x1C	USART_ISR FIFO モードが有効な場合	Res.	Res.	Res.	Res.	TXFT	RXFT	TCBGT	RXFF	TXFE	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値					X	X	X	X	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x1C	USART_ISR FIFO モードが無効の場合	Res.	Res.	Res.	Res.	Res.	Res.	TCBGT	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	ABRF	ABRE	UDR	EOBF	RTOF	CTS	CTSIF	LBDF	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値							0			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
0x20	USART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	UDRCF	EOBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGTCF	TCCF	TXFECF	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0				0	0	0		0	0	0	0	0	0	0	0	0	0
0x24	USART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	USART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表 351. USART レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	USART_PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、セクション 2.2：メモリ構成を参照してください。

38 低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）

このセクションでは、低電力ユニバーサル非同期レシーバトランスミッタ（LPUART）について説明します。

38.1 LPUART の概要

LPUART は、限られた消費電力で双方向 UART 通信が可能な UART です。32.768 kHz の LSE クロックだけで最大 9600 ボーの UART 通信が可能です。LSE クロックとは別のクロック･ソースによって LPUART にクロック供給すると、さらに高いボー･レートを達成できます。

デバイスが低電力モードのときでも、LPUART は UART フレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUART には、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェア･サポートが含まれています。

半二重単線通信とモデム操作（CTS/RTS）をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信に DMA（直接メモリアクセス）を使用できます。

38.2 LPUART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット（マーク／スペース）
- プログラム可能なボーレート
- 32.768 kHz クロック･ソースを使用して、300 baud から 9600 baud まで。
- より高い周波数のクロック･ソースを使用することにより、より高いボー･レートを達成可能
- データを送信および受信する 2 つの内部 FIFO

  各 FIFO はソフトウェアで有効／無効にすることができ、FIFO の状態用のステータスフラグを装備
- PCLK から独立したペリフェラル専用のカーネルクロックによるデュアルクロックドメイン
- プログラム可能なデータワード長（7 または 8 または 9 ビット）
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 設定可能なストップビット（1 または 2 個のストップビット）
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 転送検出フラグ：
  - 受信バッファフル
  - 送信バッファエンプティ
  - ビジーおよび送信終了フラグ
- パリティ制御：
  - パリティビットの送信
  - 受信したデータバイトのパリティ検査
- 4 つのエラー検出フラグ：
  - オーバーランエラー
  - ノイズ検出
  - フレームエラー
  - パリティエラー
- フラグ付き割込みソース
- マルチプロセッサ通信：アイドルライン検出またはアドレスマーク検出によるミュートモードからのウェイクアップ

38.3 LPUART の実装

USART と比較した LPUART の実装を次の一覧表に示します。

表 352. USART/LPUART の機能

USART/LPUART のモード／機能(1)	USART1/2/3	UART4/5	LPUART1
モデムのハードウェア·フロー制御	X	X	X
DMA を使用した連続通信	X	X	X
マルチプロセッサ通信	X	X	X
同期モード（マスタ／スレーブ）	X	-	-
スマートカードモード	X	-	-
単線半二重通信	X	X	X
Ir SIR ENDEC ブロック	X	X	-
LIN モード	X	X	-
デュアルクロックドメインと低電力モードからのウェイクアップ	X	X	X
レシーバタイムアウト割込み	X	X	-
Modbus 通信	X	X	-
自動ボーレート検出	X	X	-
ドライバイネーブル	X	X	X
USART データ長	7、8、および 9 ビット		
Tx/Rx FIFO	X	X	X
Tx/Rx FIFO サイズ	8		
低電力モードからのウェイクアップ	X(2)	X (2)	X(3)

1. X : サポートされています。
2. STOP 0 と STOP 1 モードからのウェイクアップをサポート
3. STOP 0、STOP 1、STOP 2 モードからのウェイクアップをサポート

38.4 LPUART の機能詳細

38.4.1 LPUART ブロック図

図 558. LPUART ブロック図

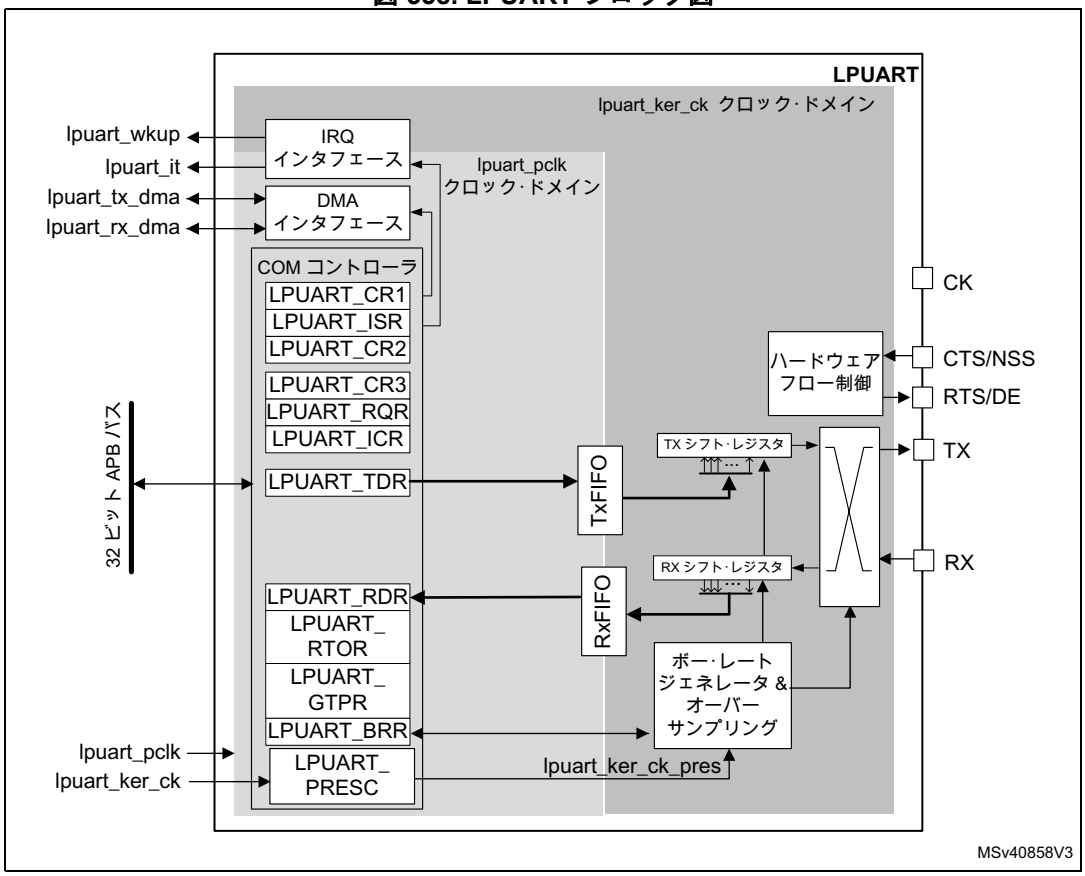


図 558 に示した簡易ブロック図には、以下の完全に独立した 2 つのクロック·ドメインが示されています。

- **lpuart_pclk** クロックドメイン

  lpuart_pclk クロック信号はペリフェラルバスインタフェースにクロック供給します。LPUART レジスタへアクセスが必要なとき、アクティブになっていなければなりません。

- **lpuart_ker_ck** カーネルクロックドメイン

  lpuart_ker_ck は LPUART クロックソースです。**lpuart_pclk** とは独立で、RCC によって供給されます。したがって、**lpuart_ker_ck** クロックが停止しているときでも、LPUART レジスタは読み書きできます。

  デュアルクロックドメイン機能が無効になっているときは、**lpuart_ker_ck** は lpuart_pclk クロックと同じになります。

lpuart_pclk と **lpuart_ker_ck** の間には何も制約はありません。**lpuart_ker_ck** は **lpuart_pclk** より速いことも遅いこともあり、十分に速く通信を管理できるソフトウェアの能力以外に制限はありません。

38.4.2 LPUART 信号

LPUART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力（RX）と送信データ出力（TX）です。

- **RX**（受信データ入力）
  RX はシリアルデータ入力です。
- **TX** （送信データ出力）
  トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線モードでは、この I/O はデータの送受信に使用されます。

RS232 ハードウェアフロー制御モード

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **CTS**（Clear To Send）
  ハイレベルのとき、この信号は現在の転送の終わりにデータ送信をブロックします。
- **RTS**（Request To Send）
  ローレベルのとき、この信号は USART がデータを受信する準備ができたことを示します。

RS485 ハードウェアフロー制御モード

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE**（Driver Enable）
  この信号は、外部トランシーバの送信モードを有効にします。

注： **DE と RTS は同じピンを共有します。**

38.4.3 LPUART キャラクタの説明

ワード長は、LPUART_CR1 レジスタの M ビット（M0：ビット 12 および M1：ビット 28）をプログラムすることによって、7、8、または 9 ビットに設定できます（図 532 を参照）。

- 7 ビットのキャラクタ長：M[1:0] = “10”
- 8 ビットのキャラクタ長：M[1:0] = “00”
- 9 ビットのキャラクタ長：M[1:0] = “01”

デフォルトでは、信号（TX または RX）はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

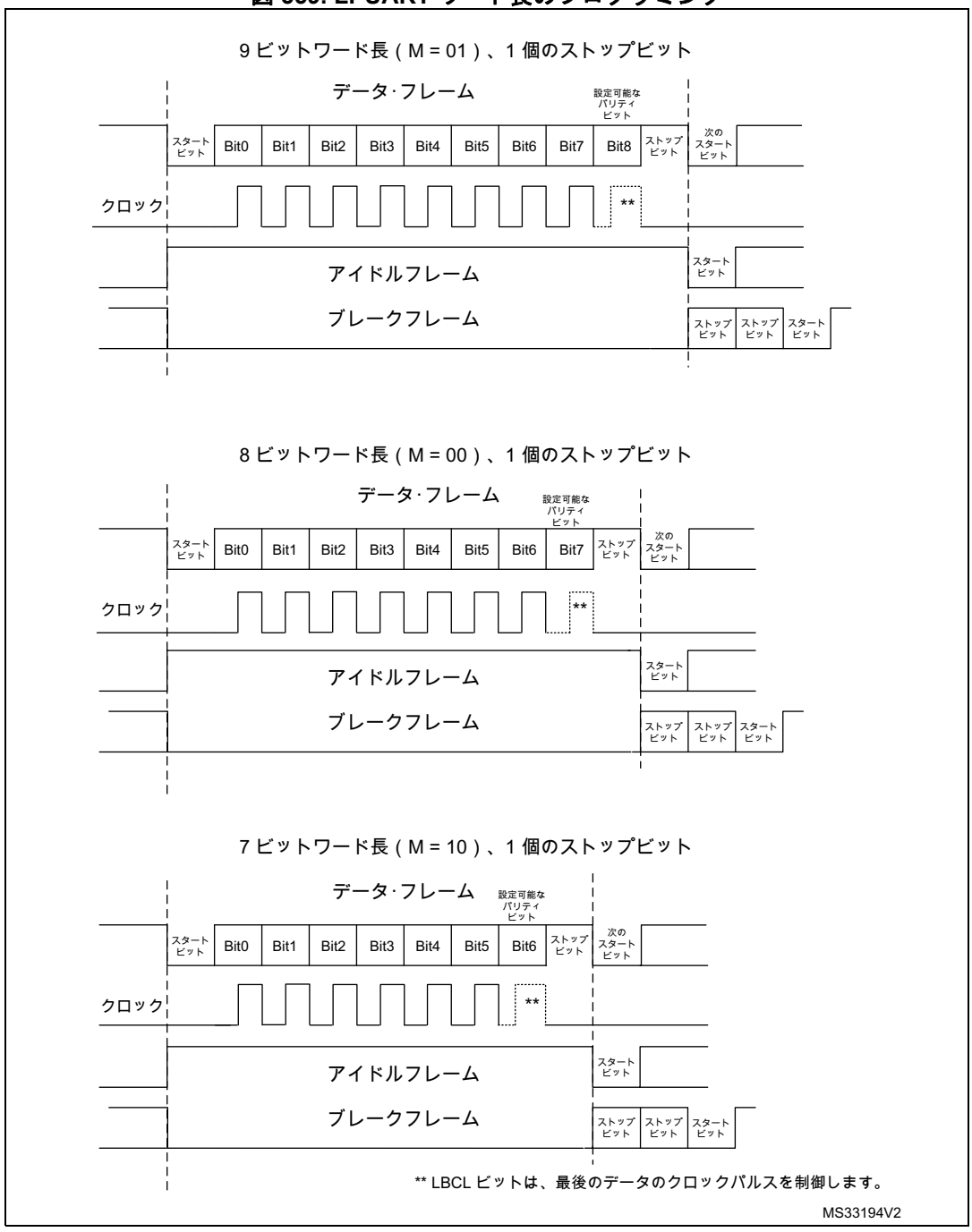
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます（「1」の数にはストップビットの数が含まれます）。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は共通ボーレートジェネレータによって駆動されます。送信および受信クロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 559. LPUART ワード長のプログラミング



38.4.4 LPUART の FIFO と閾値

LPUART は FIFO モードで動作できます。

LPUART は送信 FIFO（TXFIFO）と受信 FIFO（RXFIFO）を備えています。FIFO モードは、LPUART_CR1 レジスタの FIFOEN ビット（ビット 29）をセットすることによって有効になります。

最大のデータ･ワード長が 9 ビットなので、TXFIFO は 9 ビット幅です。しかし、RXFIFO のデフォルト幅は 12 ビットです。この理由は、レシーバは FIFO にデータを格納するだけでなく、各キャラクタに伴うエラーフラグ（パリティエラー、ノイズエラー、およびフレーミングエラーフラグ）も格納するためです。

注： **受信データは、対応するフラグとともに RXFIFO に格納されます。ただし、RDR を読み出すときは、データのみが読み出されます。**

ステータスフラグは、LPUART_ISR レジスタで入手可能です。

Tx および RX 割込みがトリガされる TXFIFO および RXFIFO のレベルを定義することができます。これらの閾値は、LPUART_CR3 制御レジスタの RXFTCFG および TXFTCFG ビットフィールドによってプログラムされます。

このとき、

- RXFIFO に受信したデータの数が RXFTCFG ビットフィールドでプログラムされた閾値に達すると、LPUART_ISR レジスタの RXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、RXFIFO 内のデータ数が、プログラムされた閾値と等しくなるまで RXFIFO が満たされることを意味します。

  RXFTCFG 分のデータが受信されると、1 データが LPUART_RDR にあり、(RXFTCFG - 1) データが RXFIFO に入っています。例として、RXFTCFG が“101”にプログラムされている場合、FIFO サイズに相当する量のデータが受信された時に、RXFT フラグがセットされます。つまり、FIFO サイズ - 1 のデータが RXFIFO に入り、1 データが LPUART_RDR に入っています。その結果、次に受信されるデータによってオーバーランフラグがセットされることはありません。
- TXFIFO の空き場所の数が TXFTCFG ビットフィールドでプログラムされた閾値に達すると、LPUART_ISR レジスタの TXFT フラグがセットされ、対応する割込みが（有効になっている場合）生成されます。

  これは、TXFIFO の空き場所の数が、プログラムされた閾値と等しくなるまで TXFIFO が空にされることを意味します。

38.4.5 LPUART トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット（TE）をセットする必要があります。送信シフトレジスタ内のデータは、TX ピンで出力されます。

キャラクタ送信

LPUART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、LPUART_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（図 558 を参照）。

FIFO モードが有効なとき、LPUART_TDR レジスタに書き込まれたデータは TXFIFO のキューに入ります。

各キャラクタの前には、スタートビット（1 ビット周期、ロー論理レベル）があります。キャラクタは、設定可能な数のストップビットで終端されます。

ストップビットの数は 1 または 2 にできます。

注： **送信データを LPUART_TDR に書き込む前に、TE ビットをセットする必要があります。**

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中の現在のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

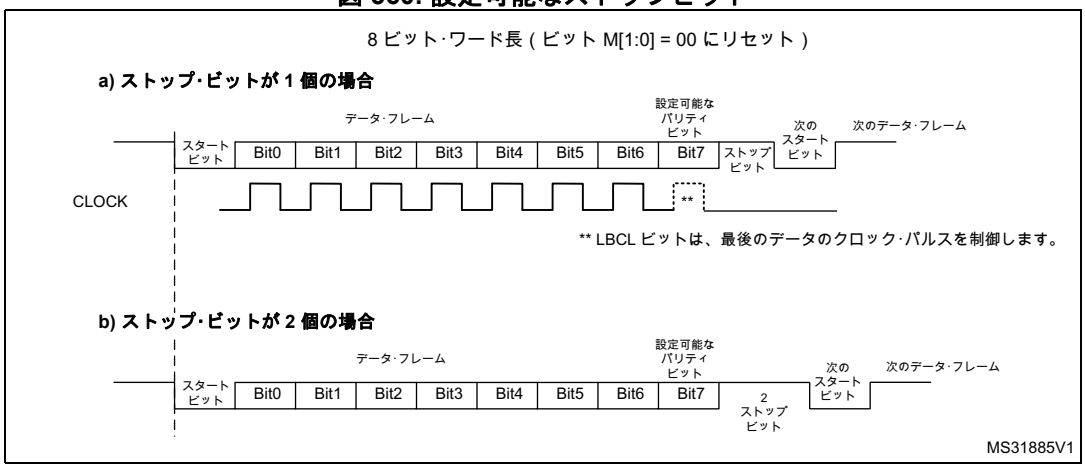
各キャラクタとともに送信されるストップビットの数は、LPUART_CR2（ビット 13、12）でプログラミングできます。

- **1 個のストップビット**：ストップビット数のデフォルト値です。
- **2 個のストップビット**：通常の LPUART モード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブレーク送信は、10個のロービット（M[1:0] = “00”のとき）または 11個のロービット（M[1:0] = “01”のとき）または 9個のロービット（M[1:0] = “10”のとき）の後に 2個のストップビットが続きます。長いブレーク（9/10/11 個のロービットを超える長さのブレーク）を送信することはできません。

図 560. 設定可能なストップビット



キャラクタ送信手順

キャラクタを送信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. LPUART_BRR レジスタを使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合は、LPUART_CR3 レジスタの DMA 有効（DMAT）を選択します。セクション 37.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを LPUART_TDR レジスタに書き込みますシングルバッファの場合、送信される各データにこの操作を繰り返します。
   – FIFO モードが無効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、TXE フラグがクリアされます。

– FIFO モードが有効になっている場合は、LPUART_TDR に 1 つのデータを書き込むと、1 つのデータが TXFIFO に追加されます。TXFNF フラグがセットされたとき、LPUART_TDR への書込み動作が行われます。このフラグは TXFIFO がフルになるまでセットされたままになります。

8. LPUART_TDR レジスタに最後のデータを書き込んだら、TC = 1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。
   – FIFO モードが無効になっている場合、これは、最後のフレームの送信が完了したことを示します。
   – FIFO モードが有効になっている場合、これは、TXFIFO とシフトレジスタの両方が空になっていることを示します。

   このチェックは、LPUART が無効になったり、停止モードに入ったりするときに、最後の送信が壊れないようにするために必要です。

1 バイト通信

- FIFO モードが無効の場合

  送信データレジスタに書き込むと、必ず TXE ビットがクリアされます。TXE フラグがハードウェアによってセットされ、以下のことを示します。
  – データは LPUART_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。

  TXEIE ビットがセットされている場合、TXE フラグは割込みを生成します。

  送信が行われているとき、LPUART_TDR レジスタへの書込み命令によってデータが TDR レジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

  送信が行われていないときには、LPUART_TDR レジスタへの書込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

- FIFO モードが有効になっている場合は、以下を示すためにハードウェアによって TXFNF（TXFIFO はフルではない）フラグがセットされます。
  – TXFIFO はフルではありません。
  – LPUART_TDR レジスタは空です。
  – 次のデータを、前のデータに上書きせずに、LPUART_TDR レジスタに書き込めます。送信が行われているとき、LPUART_TDR レジスタへの書込み命令によってデータが TXFIFO に格納されます。現在の送信の最後にデータが TXFIFO からシフトレジスタにコピーされます。

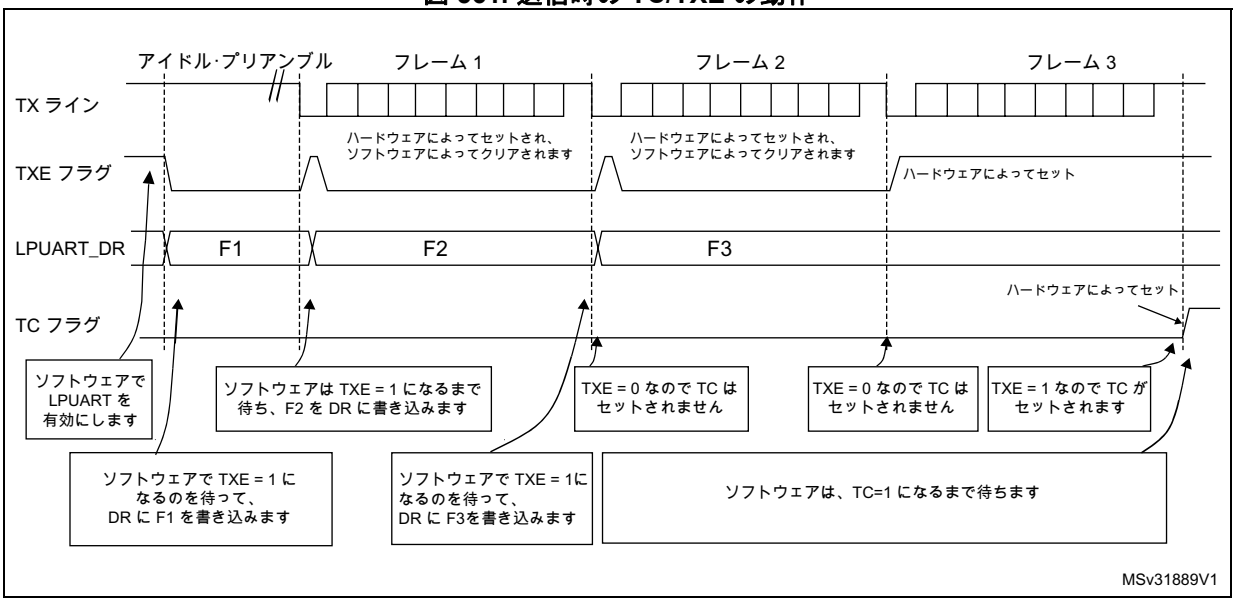
  TXFIFO がフルではない場合、LPUART_TDR レジスタへの書込み後も、TXFNF フラグは“1”のままに留まります。TXFIFO がフルになると、クリアされます。TXFNEIE ビットがセットされている場合、このフラグは割込みを生成します。

  あるいは、TXFIFO 閾値に達した時、割込みが生成され、データを TXFIFO に書き込むことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを書き込むことができます。

  フレームが送信され（ストップビットの後）、TXE フラグ（FIFO モードの場合は TXFE）がセットされると、TC ビットはハイレベルになります。LPUART_CR1 レジスタの TCIE ビットがセットされると、割込みが生成されます。

  LPUART_TDR レジスタに最後のデータを書き込んだ後は、LPUART の無効化やデバイスの低電力モード移行の前に TC = 1 になるまで待つ必要があります（図 561 : 送信時の TC/TXE の動作を参照）。

図 561. 送信時の TC/TXE の動作



注： **FIFO 管理が有効になっているときは、TXFNF フラグがデータ送信のために使用されます。**

ブレークキャラクタ

SBKRQ ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（図 559 を参照）。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブレークキャラクタが送信されます。SBKF ビットは書込み操作によってセットされ、ブレークキャラクタが完了すると（ブレークキャラクタの後のストップビット時に）ハードウェアによってリセットされます。LPUART は、次のフレームのスタートビットの認識を保証するために、ブレークフレームの終わりに 2 ビットの長さの間、ロジック 1 信号（STOP）を挿入します。

SBKRQ ビットがセットされると、現在のキャラクタ送信の最後に、ブレークキャラクタが送信されます。

FIFO モードが有効になっている場合、TXFIFO がフルになっていても、ブレークキャラクタの送信はデータ送信より優先されます。

アイドルキャラクタ

TE ビットをセットすると、LPUART は、最初のデータフレームの前にアイドルフレームを送信します。

38.4.6 LPUART レシーバ

LPUART は、LPUART_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

LPUART では、立ち下がりエッジが Rx ラインで発生したときスタートビットが検出され、その後、スタートビットの中央でサンプルが採取され、まだ“0”であることが確認されます。スタートサンプルが“1”であった場合、ノイズエラーフラグ（NE）がセットされ、スタートビットが破棄され、レシーバは新しいスタートビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUART の受信時には、データは RX ピンを通じて LSB ファースト（デフォルト設定）でシフトインされます。このモードでは、LPUART_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ（RDR）で構成されます。

キャラクタ受信手順

キャラクタを受信するには、次の手順に従います。

1. LPUART_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ LPUART_BRR を使用して、目的のボーレートを選択します。
3. LPUART_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUART_CR1 レジスタの UE ビットに“1”を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUART_CR3 レジスタの DMA 有効（DMAR）を選択します。セクション 37.5.10 : USART マルチプロセッサ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUART_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- FIFO モードが無効な場合は、RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読出し可能です（関連するエラーフラグも同様です）。
- FIFO モードが有効な場合、RXFIFO が空ではないことを示す RXFNE ビットがセットされます。LPUART_RDR を読み出すと、RXFIFO に入力された最も古いデータが返されます。データが受信されると、対応するエラービットとともに RXFIFO に格納されます。
- RXNEIE（FIFO モードの場合は RXFNEIE）ビットがセットされていた場合、割込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。
- マルチバッファ通信モードでは、
  - FIFO モードが無効のとき、RXNE フラグはバイト受信ごとにセットされ、受信データレジスタの DMA 読出しによってクリアされます。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。DMA リクエストのたびに、RXFIFO から 1 データが取り出されます。DMA リクエストは、RXFIFO が空ではないとき、すなわち、RXFIFO から読み出されるべきデータがあるときに、トリガされます。
- シングルバッファモードでは、
  - FIFO モードが無効の場合、RXNE フラグのクリアは、ソフトウェアによる LPUART_RDR レジスタからの読出しによって行われます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。
  - FIFO モードが有効な場合は、RXFIFO が空ではないときに RXFNE フラグがセットされます。LPUART_RDR レジスタからの読出し動作のたびに、RXFIFO から 1 つのデータが取り出されます。RXFIFO が空になると、RXFNE フラグがクリアされます。RXFNE フラグは、LPUART_RQR レジスタの RXFRQ ビットに 1 を書き込むことによってもクリアすることもできます。RXFIFO がフルのとき、オーバーランエラーを避けるため、次のキャラクタの受信が終了する前に、RXFIFO 内の最初のエントリを読み出す必要があります。RXFNEIE ビットがセットされている場合、RXFNE フラグは割込みを生成します。あるい

は、RXFIFO 閾値に達した時、割込みが生成され、データを RXFIFO から読み出すことができます。この場合、CPU は、プログラムされた閾値によって定義されたデータのブロックを読み出すことができます。

ブレークキャラクタ

ブレーク･キャラクタを受信すると、LPUART はブレーク･キャラクタをフレーミング･エラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出された場合、データキャラクタ受信と同じように処理されますが、違いは、IDLEIE ビットがセットされている場合に割込みが生成されることです。

オーバーランエラー

- FIFO モードが無効の場合

  RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。

  RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。RXNE フラグは、バイトを受信するたびにセットされます。

  次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RDR の内容は失われません。LPUART_RDR の読出しが行われると、前のデータが使用できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

- FIFO モードが有効な場合

  受信 FIFO がフルのときにシフトレジスタが転送される準備ができると、オーバーランエラーが発生します。

  RXFIFO に 1 つの空き場所ができるまで、データをシフトレジスタから LPUART_RDR レジスタに転送することはできません。RXFIFO が空でないとき、RXFNE フラグがセットされます。

  RXFIFO がフルで、シフトレジスタが転送される準備ができている場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

  – ORE ビットがセットされます。
  – RXFIFO の最初のエントリは失われません。LPUART_RDR の読出しを行うと、そのデータが入手できます。
  – シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
  – RXFNEIE ビットまたは EIE ビットがセットされている場合、割込みが生成されます。

ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： **ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。T**

FIFO モードが無効になっているとき、2 つの可能性があります。

- **RXNE = 1 の場合、有効な最後のデータは受信レジスタ（RDR）に格納されていて、読出しが可能です。**
- **RXNE= 0 の場合、最後の有効なデータは読出し済みなので、RDR レジスタには読み出すデータは残っていません。有効な最後のデータが RDR レジスタで読み出されると同時に新しいデータが受信されて失われると、この状況が発生します。**

クロック･ソースの選択

クロック･ソースの選択は、クロック制御システムを通じて行われます（リセットおよびクロック制御（RCC）のセクションを参照）。クロックソースは、UE ビットのセットによって LPUART を有効にする前に選ぶ必要があります。

クロックソースは、次の 2 つの基準に従って選択する必要があります。

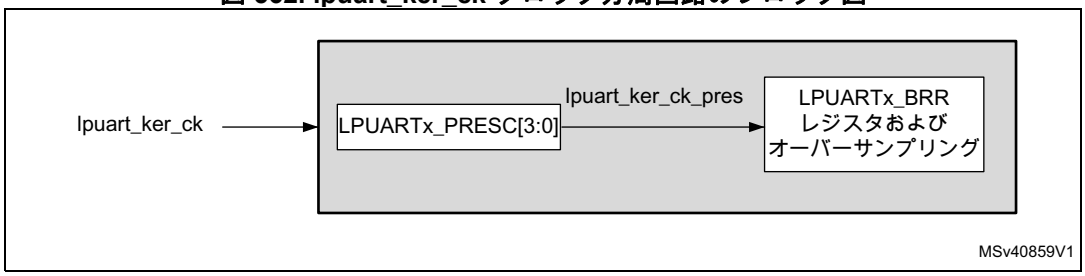
- LPUART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、lpuart_ker_ck です。

デュアルクロックドメインと低電力モードからのウェイクアップ機能がサポートされるときには、lpuart_ker_ck クロックソースは RCC で設定できます（*リセットおよびクロック制御（RCC）のセクション*を参照）。そうしない場合、lpuart_ker_ck クロックは lpuart_pclk と同じになります。

lpuart_ker_ck クロックは、LPUART_PRESC レジスタのプログラム可能な係数によって分周できます。

図 562. lpuart_ker_ck クロック分周回路のブロック図



lpuart_ker_ck ソースによっては、LPUART は MCU が低電力モードのときにデータを受信することができます。受信データとウェイクアップモードの選択に応じて、LPUART は必要なときに MCU をウェイクアップして、ソフトウェアが LPUART_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、LPUART 通信を可能にするためには、システムをアクティブにする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは各着信ビットをビット周期のできるだけ中央でサンプリングします。着信ビットごとに取得されるサンプルは 1 つのみです。

注： **データのノイズ検出は行われません。**

フレーミングエラー

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合、フレーミングエラーが検出されます。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから LPUART_RDR レジスタへ転送されます。
- 1 バイト通信の場合、割込みは生成されません。ただし、このビットは、割込みを生成する RXNE ビットと同時に立ち上がります。マルチバッファ通信の場合、LPUART_CR3 レジスタの EIE ビットがセットされている場合に割込みが発行されます。

FE ビットは、LPUART_ICR レジスタの FECF に 1 を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、LPUART_CR2 の制御ビットを通じて設定でき、通常モードでは 1 または 2 にできます。

- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- **2 個のストップビット**：2 個のストップビットのサンプリングは、2 番目のストップビットの中央で行われます。RXNE および FE フラグは、このサンプルの直後、すなわち 2 番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

38.4.7 LPUART ボーレート生成

レシーバとトランスミッタ（Rx と Tx）のボーレートは、LPUART_BRR レジスタでプログラムされた値に設定されます。

$$\text{Tx/Rx ボー} = \frac{256 \times \text{lpuartckpres}}{\text{LPUARTDIV}}$$

LPUARTDIV は、LPUART_BRR レジスタで定義されます。

注： **ボーカウンタは、LPUART_BRR への書込み操作後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。**

LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

f_{CK} は、3 x ボーレートから 4096 x ボーレートの範囲内でなければなりません。

LPUART クロックソースが LSE の場合に達成できる最大ボーレートは、9600 ボーです。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。たとえば、LPUART クロックソース周波数が 100 MHz の場合、達成できる最大ボーレートは約 33 M ボーです。

表 353. lpuart_ker_ck_pres = 32.768 kHz でプログラムされたボー・レートの誤差計算

ボー・レート		lpuart_ker_ck_pres = 32.768 kHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） B レート / 目標の B レート
1	300 ボー	300 ボー	0x6D3A	0
2	600 ボー	600 ボー	0x369D	0
3	1200 ボー	1200.087 ボー	0x1B4E	0.007
4	2400 ボー	2400.17 ボー	0xDA7	0.007
5	4800 ボー	4801.72 ボー	0x6D3	0.035
6	9600 ボー	9608.94 ボー	0x369	0.093

表 354. f_{CK} = 100 MHz でプログラムされたボー・レートの誤差計算

ボー・レート		f_{CK} = 100 MHz		
S.No	目標	実際	ボーレートレジスタに プログラミングされている値	誤差(%) =（計算値 - 目標値） B レート / 目標の B レート
1	38400 ボー	38400.04 ボー	A2C2A	0.0001
2	57600 ボー	57600.06 ボー	6C81C	0.0001
3	115200 ボー	115200.12 ボー	3640E	0.0001
4	230400 ボー	230400.23 ボー	1B207	0.0001
5	460800 ボー	460804.61 ボー	D903	0,001
6	921600 ボー	921625.81 ボー	6C81	0.0028
7	4000 K ボー	4000000.00 ボー	1900	0
8	10000 K ボー	10000000.00 ボー	A00	0
9	20000 K ボー	20000000.00 ボー	500	0
10	33000 K ボー	33032258.06 ボー	307	0.1

38.4.8 クロック偏差に対する LPUART レシーバの許容誤差

LPUART の非同期レシーバは、クロックシステムの合計偏差が LPUART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA：トランスミッタの誤差に起因する偏差（トランスミッタのローカルオシレータの偏差も含みます）
- DQUANT：レシーバのボーレート量子化に起因する誤差
- DREC：レシーバローカルオシレータの偏差
- DTCL：送信ラインに起因する偏差（一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因）

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{LPUART レシーバの許容誤差}$$

ここで、

DWU は、低電力モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

LPUART レシーバは、表 355 で指定された最大許容偏差まで、データを正しく受信できます。

- LPUART_CR2 レジスタの STOP[1:0] ビットによって定義されたストップビットの数
- LPUART_BRR レジスタの値

表 355. LPUART レシーバの許容誤差

M ビット	768 < BRR < 1024	1024 < BRR < 2048	2048 < BRR < 4096	4096 ≤ BRR
8 ビット（M = 00）、ストップ･ビット 1 個	1.82%	2.56%	3.90%	4.42%
9 ビット（M = 01）、ストップ･ビット 1 個	1.69%	2.33%	2.53%	4.14%
7 ビット（M = 10）、ストップ･ビット 1 個	2.08%	2.86%	4.35%	4.42%
8 ビット（M = 00）、ストップ･ビット 2 個	2.08%	2.86%	4.35%	4.42%
9 ビット（M = 01）、ストップ･ビット 2 個	1.82%	2.56%	3.90%	4.42%
7 ビット（M = 10）、ストップ･ビット 2 個	2.34%	3.23%	4.92%	4.42%

注： **表 355 で指定されたデータは、M ビット = 00 で受信フレームに正確に 10 ビット時間（M = 01 の場合は 11 ビット時間、M = 10 の場合は 9 ビット時間）のアイドル･フレームがいくつか含まれる特殊なケースでは若干異なることがあります。**

38.4.9 LPUART マルチプロセッサ通信

LPUART のマルチプロセッサ通信が可能です（ネットワーク内で複数の LPUART を接続して）。たとえば、1 つの LPUART をマスタとして、その TX 出力を別の LPUART の RX 入力に接続することができます。別の LPUART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する LPUART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUART_CR1 レジスタの MME ビットをセットする必要があります。

注： **FIFO マネージメントが有効になっていて MME がすでにセットされている場合は、MME ビットはクリアしてはなりません。クリアした場合はすぐに（2 lpuart_ker_ck サイクル以内に）再セットしてください。そうしないとミュートモードはアクティブのままになることがあります。**

ミュートモードが有効な場合、

- 受信ステータスビットはセットできません。
- 受信割込みはすべて禁止されます。
- LPUART_ISR レジスタの RWU ビットは“1”にセットされます。特定の条件下では、LPUART_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

LPUART は、LPUART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

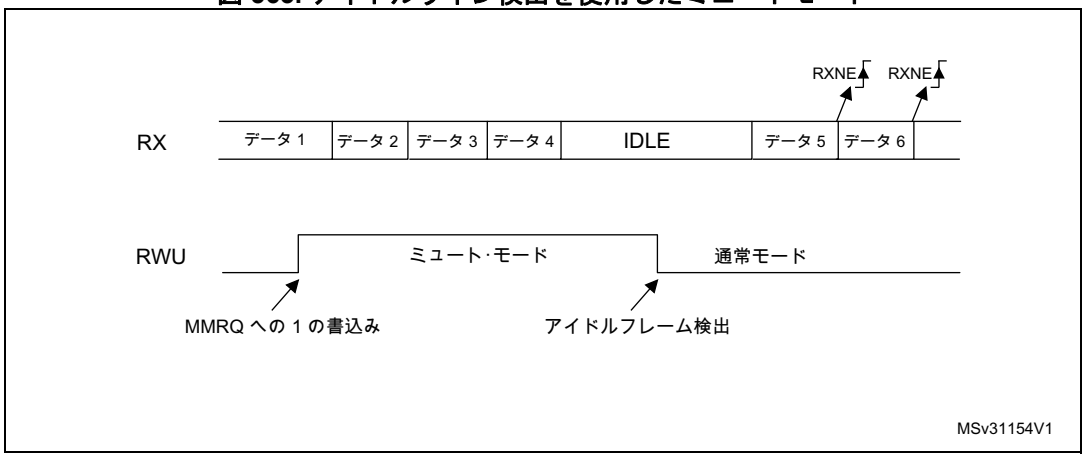
- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドル･ライン検出（WAKE = 0）

MMRQ ビットに 1 が書き込まれると LPUART はミュート･モードに移行し、RWU が自動的にセットされます。

LPUART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、LPUART_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を 図 563 に示します。

図 563. アイドルライン検出を使用したミュートモード



注： **IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません（RWU はセットされません）。**

ラインが IDLE のときに LPUART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます（1 キャラクタフレームの受信後だけでなく）。

4 ビット／7 ビット･アドレス･マーク検出（WAKE = 1）

このモードでは、バイトの MSB が 1 の場合、そのバイトはアドレスとして認識され、それ以外の場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDM7 ビットを使用して行われます。この 4 ビット／7 ビットワードは、レシーバによって、LPUART_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： **7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス（ADD[5:0] および ADD[7:0]）で行われます。**

プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。LPUART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割込みも DMA リクエストも発行されません。

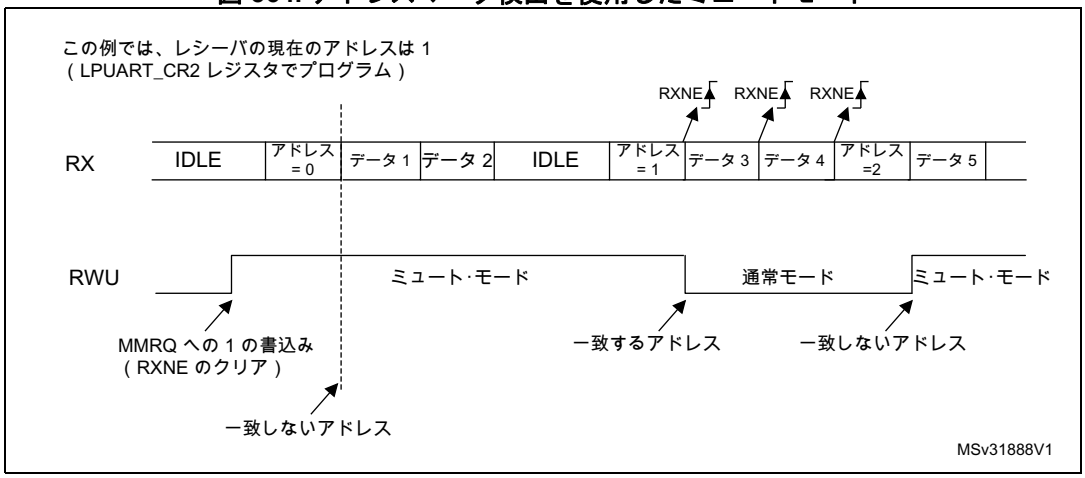
MMRQ ビットに“1”が書き込まれたときにも、LPUART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE/RXFNE ビットがセットされます。

注： **FIFO 管理が有効になっている場合、レシーバがデータの最後のビットをサンプリングしている間に MMRQ ビットがセットされると、ミュートモードに実際に移行する前にこのデータが受信されることがあります。**

アドレスマーク検出を使用したミュートモードの動作例を 図 564 に示します。

図 564. アドレスマーク検出を使用したミュートモード



38.4.10 LPUART パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティチェック）を有効にするには、LPUART_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な LPUART フレームフォーマットを表 356 に示します。

表 356. LPUART フレームのフォーマット

M ビット	PCE ビット	LPUART フレーム(1)
00	0	| SB | 8 ビットデータ | STB |
00	1	| SB | 7 ビットデータ | PB | STB |
01	0	| SB | 9 ビットデータ | STB |
01	1	| SB | 8 ビットデータ | PB | STB |
10	0	| SB | 7 ビットデータ | STB |
10	1	| SB | 6 ビットデータ | PB | STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。
2. データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、8 または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 のように 4 つのビットがセットされた場合、偶数パリティを選択していれば（LPUART_CR1 の PS ビット = 0）、パリティ･ビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で 「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 のように 4 つのビットがセットされた場合、奇数パリティを選択していれば（LPUART_CR1 の PS ビット = 1）、パリティ･ビットは 1 になります。

受信中のパリティ･チェック

パリティチェックに失敗した場合、LPUART_ISR レジスタの PE フラグがセットされ、LPUART_CR1 レジスタの PEIE ビットがセットされている場合は割込みが生成されます。PE フラグは、LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUART_CR1 の PCE ビットがセットされている場合、データ･レジスタに書き込まれたデータの MSB ビットは送信されますが、パリティ･ビットによって変更され、偶数パリティを選択している場合（PS = 0）は「1」が偶数個、奇数パリティを選択している場合（PS = 1）は「1」が奇数個になります。

38.4.11 LPUART 単線半二重通信

単線半二重モードを選択するには、LPUART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- LPUART_CR2 レジスタの LINEN および CLKEN ビット
- LPUART_CR3 レジスタの SCEN および IREN ビット

LPUART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに“1”が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の LPUART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： **LPUART 通信では、1 個のストップビット設定の場合、ストップビットの中央で RXNE フラグがセットされます。**

38.4.12 DMA および LPUART を使用した連続通信

LPUART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： **DMA モードがサポートされるかどうかについては、1564 ページのセクション 37.4 : USART の実装を参照してください。DMA がサポートされない場合は、セクション 37.5.6: の説明に従って LPUSRT を使用してください。連続通信を行うには。FIFO が無効のとき、LPUART_ISR レジスタの TXE/RXNE フラグをクリアします。**

DMA を使用した送信

DMA モードでの送信を有効にするには、LPUART_CR3 レジスタの DMAT ビットをセットします。データは、TXE フラグ（FIFO モードが有効な場合は TXFNF フラグ）がセットされると、DMA ペリフェラル（対応する「ダイレクトメモリアクセスコントローラ」セクションを参照）を使用して設定された SRAM 領域から LPUART_TDR レジスタにロードされます。DMA チャネルを LPUART 送信用に割り付けるには、次の手順を実行します（x はチャネル番号を示します）

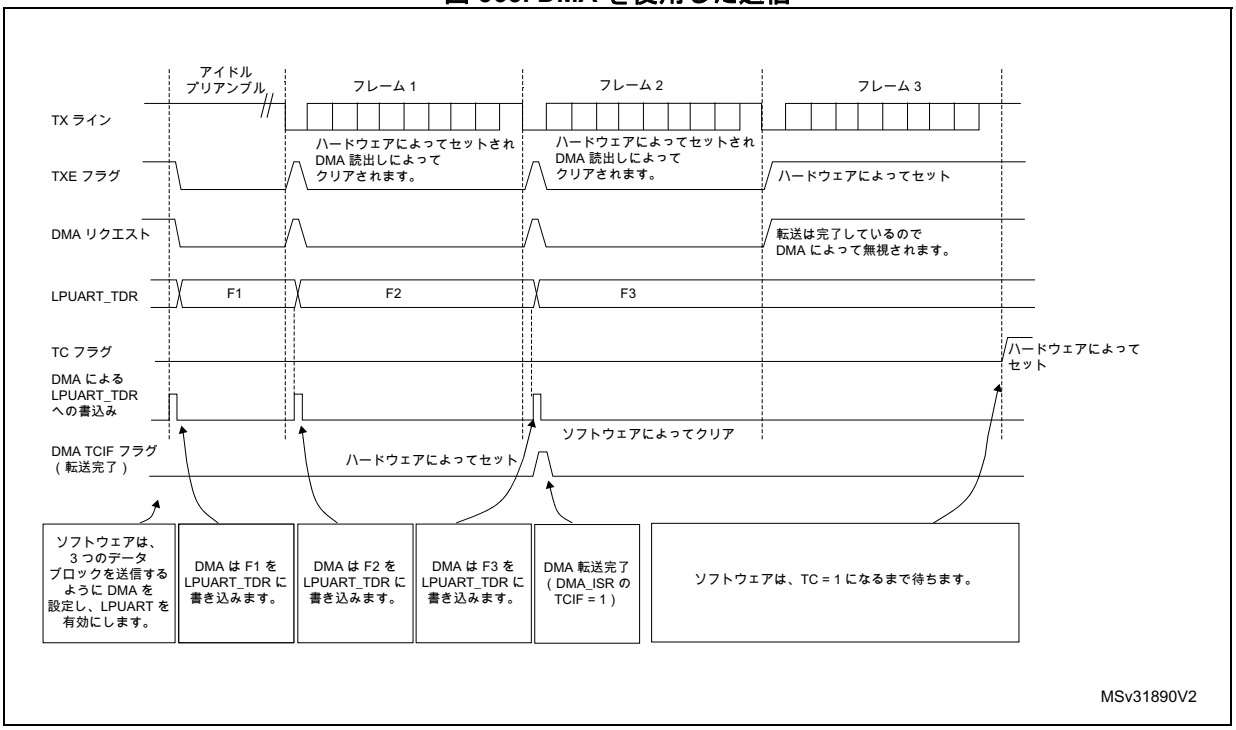
1. DMA 制御レジスタに LPUART_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、メモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE（または FIFO モードが有効な場合は TXFNF）イベント後に、このメモリ領域から LPUART_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割込み生成を設定します。

6. LPUART_ICR レジスタの TCCF ビットをセットすることによって、LPUART_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと（DMA_ISR レジスタの TCIF フラグがセットされます）、TC フラグを観察して LPUART 通信の完了を確認することができます。これは、LPUART を無効にしたり 低消費電力モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC = 1 になるまで待つ必要があります。データの転送中、TC フラグはクリアされた状態を維持し、最後のフレームの送信が終了するとハードウェアによってセットされます。

図 565. DMA を使用した送信



注： FIFO 管理が有効になっているときは、DMA リクエストは、送信 FIFO ノットフル（すなわち、TXFNF = 1）によってトリガされます。

DMA を使用した受信

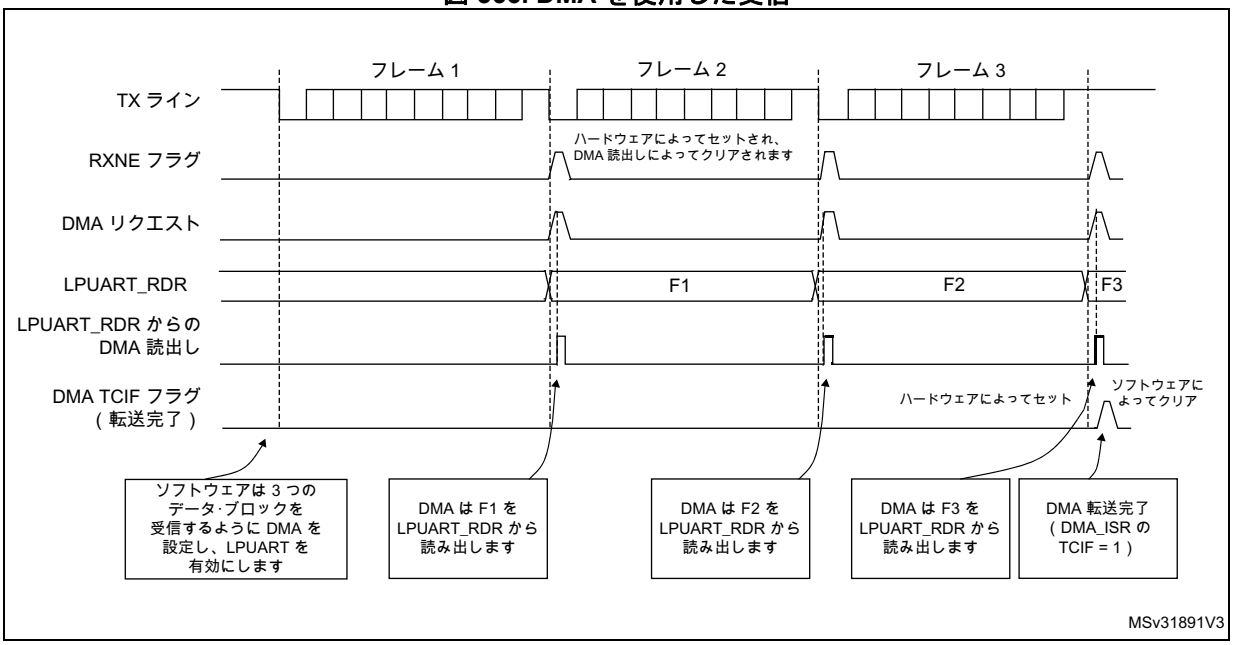
DMA モードでの受信を有効にするには、LPUART_CR3 レジスタの DMAR ビットをセットします。データバイトが受信されるたびに、データは、LPUART_RDR レジスタから DMA ペリフェラル（対応するダイレクトメモリアクセスコントローラ（DMA）のセクションを参照）を使用して設定された SRAM 領域にロードされます。DMA チャネルを LPUART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに LPUART_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、このアドレスからメモリに移動されます。

2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE（FIFO モードが有効な場合は RXFNE）イベント後に、LPUART_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割込み生成を設定します。
6. DMA 制御レジスタのチャネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャネルの割込みベクタに基づいて割込みを生成します。

図 566. DMA を使用した受信



注： **FIFO 管理が有効になっているときは、DMA リクエストは、受信 FIFO ノットエンプティ（すなわち、RXFNE = 1）によってトリガされます。**

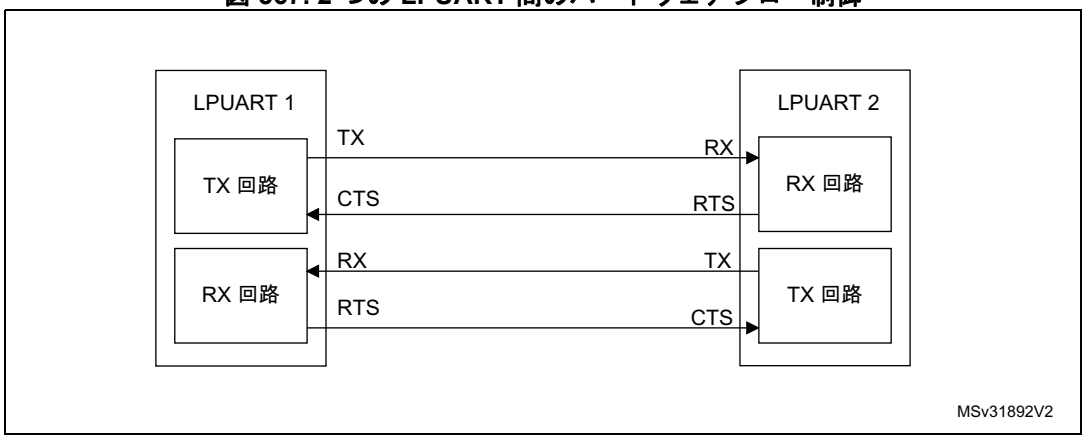
マルチバッファ通信における割込み生成とエラーフラグ

マルチバッファ通信モードでトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割込み有効フラグがセットされている場合、割込みが生成されます。1 バイト受信において RXNE（FIFO モードが有効な場合は RXFNE）とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割込み有効ビット（LPUART_CR3 レジスタの EIE ビット）があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割込みが有効になります。

38.4.13 RS232 ハードウェアフロー制御および RS485 ドライバ有効

CTS 入力と RTS 出力を使用すると、2 つのデバイス間のシリアル･データ･フローを制御できます。図 553 に、このモードで 2 つのデバイスを接続する方法を示します。

図 567. 2 つの LPUART 間のハードウェアフロー制御

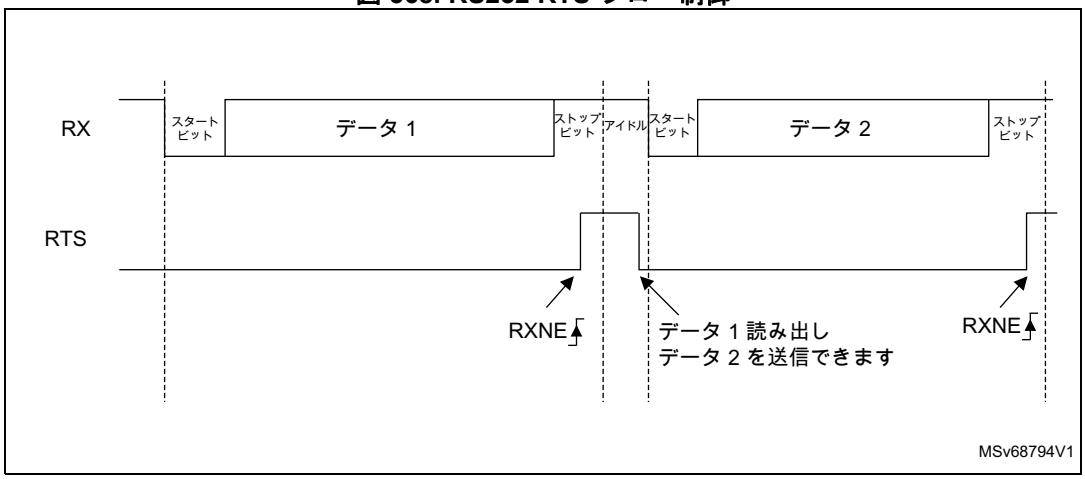


RS232 RTS と CTS のフロー制御は、LPUART_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合（RTSE = 1）、LPUART レシーバが新しいデータを受信可能である限り、RTS はデアサートされます（ロー固定）。受信レジスタがフルになると RTS がアサートされ、現在のフレームが終わると送信が停止する予定であることを示します。図 568 に、RTS フロー制御が有効な場合の通信例を示します。

図 568. RS232 RTS フロー制御



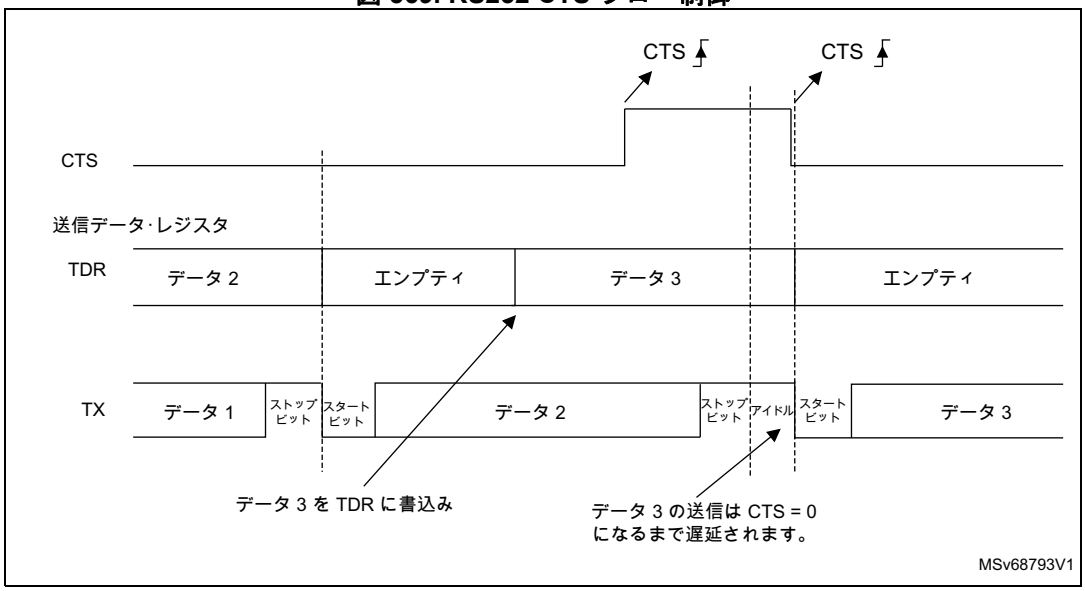
注： **FIFO モードが有効な場合は、RXFIFO がフルのときにのみ、RTS がアサートされます。**

RS232 CTS フロー制御

CTS フロー制御が有効な場合（CTSE = 1）、トランスミッタは、CTS 入力をチェックしてから次のフレームを送信します。CTS がデアサートされている場合（ローに固定）、次のデータが送信されます（データが送信される状態であること、つまり TXE/TXFE = 0 であるとします）。そうでない場合、データは送信されません。送信中に CTS がアサートされると、現在の送信が完了してからトランスミッタが停止します。

CTSE = 1 の場合、CTS 入力がトグルすると、CTSIF ステータス･ビットはハードウェアによってただちに自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUART_CR3 レジスタの CTSIE ビットがセットされている場合、割込みが生成されます。図 569 に、CTS フロー制御が有効な場合の通信例を示します。

図 569. RS232 CTS フロー制御



注： **正しい動作とするには、現在のキャラクタが終了する少なくとも 3 LPUART クロック･ソース周期前に CTS をデアサートする必要があります。また、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。**

RS485 ドライバ有効

ドライバ有効機能を有効にするには、LPUART_CR3 制御レジスタのビット DEM をセットします。これにより、DE（Driver Enable）信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から スタートビットの開始までの時間です。LPUART_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。LPUART_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、LPUART_CR3 制御レジスタの DEP ビットを使用して設定できます。

LPUART の DEAT および DEDT は LPUART クロックソース（f_{CK}）サイクルで表されます。

- ドライバ有効アサーション時間は次のようになります。
  - (1 + (DEAT x P)) x f_{CK} 、P ≠ 0 の場合
  - (1 + DEAT) x f_{CK}、P = 0 の場合
- ドライバ有効ネゲート時間は次のようになります。
  - (1 + (DEDT x P)) x f_{CK} 、P ≠ 0 の場合
  - (1 + DEDT) x f_{CK}、P = 0 の場合

ここで、P = BRR[20:11]

38.4.14 LPUART 低電力管理

LPUART には、高度な低消費電力モード機能があり、lpuart_pclk クロックが無効になっているときでもデータを適切に転送することができます。

LPUART は、UESM ビットがセットされているとき、MCU を低消費電力モードからウェイクアップできます。

lpuart_pclk がゲートされているとき、lpuart_pclk クロックの有効化を必要とする特定の動作が必要になった場合、LPUART はウェイクアップ割込み（lpuart_wkup）を生成します。

- FIFO モードが無効の場合

  LPUART データレジスタを空にするために luart_pclk クロックを有効にする必要があります。

  この場合、lpuart_wkup 割込みのソースは“1”にセットされた RXNE です。低消費電力モードに入る前に RXNEIE ビットをセットする必要があります。

- FIFO モードが有効な場合

  次のために luart_pclk クロックを有効にする必要があります。
  - TXFIFO を満たすため
  - または RXFIFO を空にするため

  この場合、lpuart_wkup 割込みのソースになる可能性のあるものは以下の通りです。
  - RXFIFO ノットエンプティ。この場合、低消費電力モードに入る前に RXFNEIE ビットをセットする必要があります。
  - RXFIFO フル。この場合、低消費電力モードに入る前に RXFFIE ビットをセットする必要があります。受信データの数は RXFIFO のサイズに一致し、RXFF フラグはセットされません。
  - TXFIFO エンプティ。この場合、低消費電力モードに入る前に TXFEIE ビットをセットする必要があります。

  これによって、低消費電力モード中にデータを TXFIFO/RXFIFO に送信／受信することができます。

  低消費電力モードで、オーバーラン／アンダーランエラーを避けてデータを送信／受信するために、lpuart_wkup 割込みソースになり得るのは次のイベントのうちの 1 つです。
  - TXFIFO 閾値に達した。この場合、低消費電力モードに入る前に TXFTIE ビットをセットする必要があります。
  - RXFIFO 閾値に達した。この場合、低消費電力モードに入る前に RXFTIE ビットをセットする必要があります。

  たとえば、ウェイクアップ時間が、ラインを経て 1 バイトを受信する時間より少ない場合は、アプリケーションは閾値を RXFIFO の最大サイズに設定できます。

  MCU を低消費電力モードからウェイクアップするための RXFIFO フル、TXFIFO エンプティ、RXFIFO ノットエンプティ、および RXFIFO/TXFIFO 閾値割込みを使用すれば、低消費電力モード中にできるだけ多くの LPUART 転送を行うことができ、電力消費を最適化できるメリットがあります。

あるいは、WUS ビットフィールドによって、特定の lpuart_wkup 割込みを選択することもできます。

ウェイクアップ·イベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合は lpuart_wkup 割込みが生成されます。

注： **低消費電力モードに移行する前に、LPUART 転送が進行中ではないことを確認してください。BUSY フラグをチェックすることでは、データ受信中に低消費電力モードに入らないことを保証できません。**

WUF フラグは、MCU が 低消費電力モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に低消費電力モードに入るときには、REACK ビットをチェックして、LPUART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、低消費電力モードに入る前に無効化し、低消費電力モードの終了時に再び有効にする必要があります。

FIFO が有効なときには、アドレス一致時の低消費電力モードからのウェイクアップはミュートモードが有効な場合のみ可能です。

低消費電力モードでのミュートモードの使用

低消費電力モードに入る前に LPUART がミュートモードになった場合は、

- アイドル検出は低消費電力モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、低消費電力モードからのウェイクアップのソースもアドレス一致でなければなりません。低消費電力モードに入るときに RXNE フラグがセットされた場合、アドレス一致によって低消費電力モードからウェイクアップしても、インタフェースはミュートモードのままです。

注： **FIFO 管理が有効なとき、ミュートモードは何の制約もなく低消費電力モードからのウェイクアップとともに使用されます（すなわち、ミュートおよび低消費電力モードについて上に述べた 2 点は、FIFO 管理が無効なときのみに有効です）。**

低消費電力モードで LPUART カーネルクロック（lpuart_ker_ck）がオフのときの低消費電力モードからのウェイクアップ

低消費電力モード中、lpuart_ker_ck クロックがオフになっている場合、LPUART 受信ラインの立ち下がりエッジが検出されると、lpuart_ker_ck_req 信号によって LPUART インタフェースが lpuart_ker_ck クロックをオンにするようリクエストします。その後、lpuart_ker_ck がフレーム受信に使用されます。

ウェイクアップイベントが確認された場合、MCU は低消費電力モードからウェイクアップし、データ受信が正常に続行します。

ウェイクアップ·イベントが確認されない場合、lpuart_ker_ck クロックが再度オフになり、マイクロコントローラはウェイクアップせずに低消費電力モードに留まり、カーネル·クロック·リクエストは解除されます。

以下の例は、ウェイクアップ·イベントが「アドレス一致検出」にプログラムされ、FIFO 管理が無効になっている場合を示しています。

図 570 に、ウェイクアップイベントが確認された時の 動作を示します。

図 570. 確認されたウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）

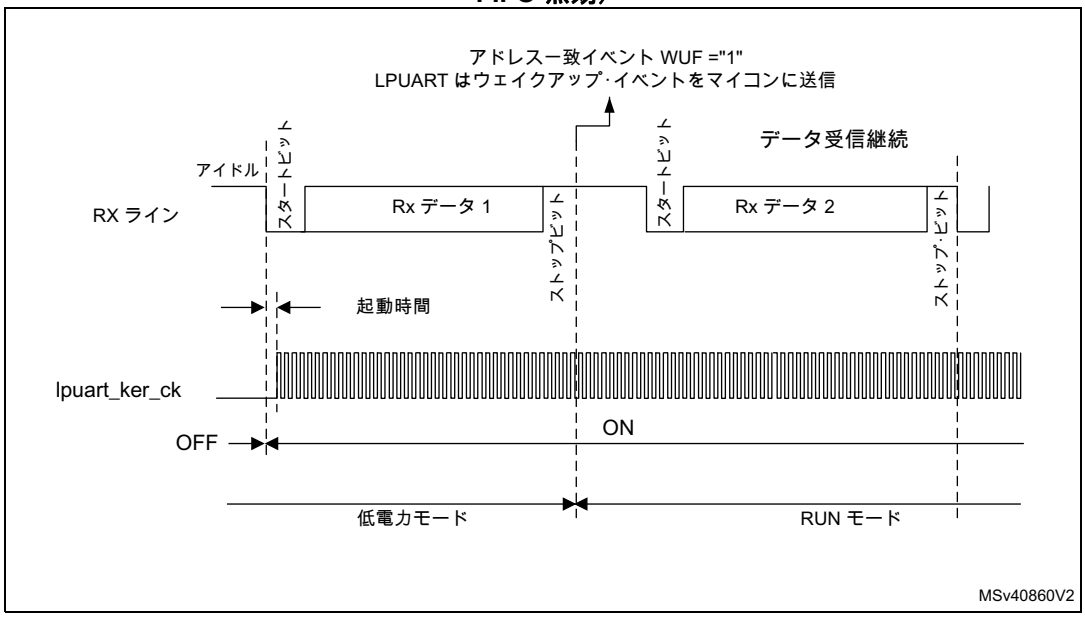
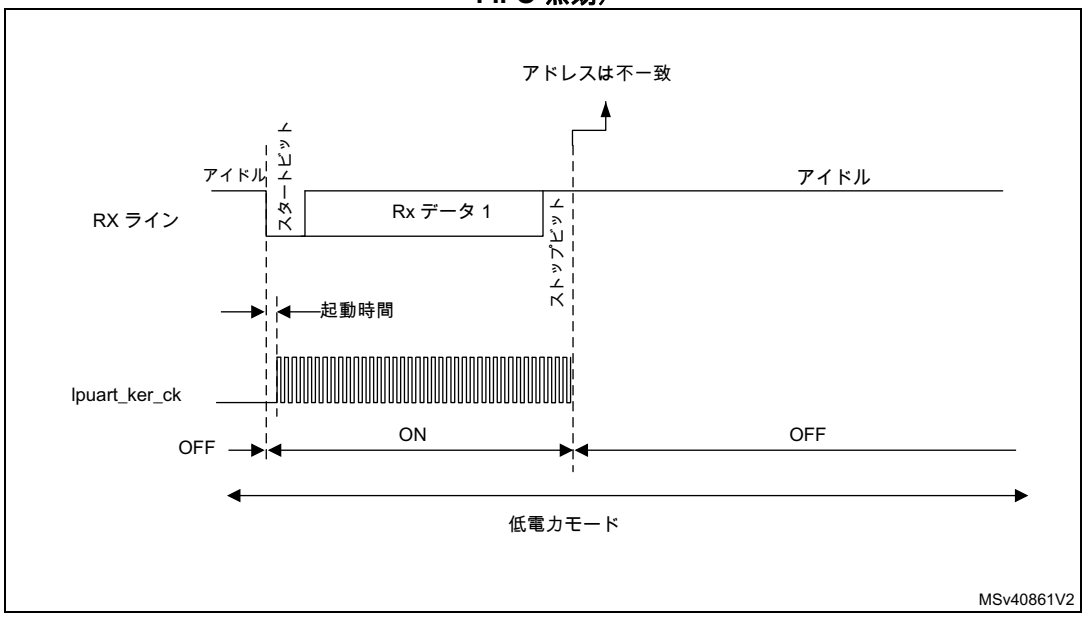


図 571 に、ウェイクアップイベントが確認されない時の動作を示します。

図 571. 確認されないウェイクアップイベント（ウェイクアップイベント = アドレス一致、FIFO 無効）



注： **上図は、アドレス一致または受信フレームがウェイクアップイベントとして使用されたとき、有効です。ウェイクアップイベントがスタートビット検出である場合、LPUART はスタートビットの終端にウェイクアップイベントを MCU に送ります。**

MCU の低消費電力モードからの正常なウェイクアップを可能にする最大 LPUART ボーレートの決定

MCU の低消費電力モードからの正常なウェイクアップを可能にする最大ボーレートは、ウェイクアップ時間パラメータ（デバイスのデータシートを参照）と LPUART レシーバの許容誤差（セクション 38.4.8：クロック偏差に対する LPUART レシーバの許容誤差を参照）に依存します。

例として、OVER8 = 0、M ビット = "01"、ONEBIT = 0、BRR [3:0] = 0000 とします。

この条件では、表 355：LPUART レシーバの許容誤差によると、LPUART レシーバの許容誤差は 3.41% です。

DTRA + DQUANT + DREC + DTCL + DWU < LPUART レシーバの許容誤差

$D_{WUmax} = t_{WULPUART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min} = t_{WULPUART} / (11 \times D_{WUmax})$

$t_{WULPUART}$ は、低消費電力モードからのウェイクアップ時間です。

パラメータ DTRA、DQUANT、DREC、および DTCL が 0% であるときに理想的なケースを考えた場合、DWU の最大値は 3.41% です。実際には、最低 lpuart_ker_ck 精度を考慮する必要があります。

たとえば HSI16 が puart_ker_ck として使用され、HSI16 の精度が 1% の場合、以下を得ることができます。

$t_{WULPUART}$ = 3 µs（値は例にすぎません。正確な値については、デバイスのデータシートを参照してください）。

D_{WUmax} = 3.41% - 1% = 2.41%

$T_{bit\ min}$ = 3 µs/ (11 x 2.41%) = 11.32 µs

その結果、低消費電力モードからの正しいウェイクアップを可能にする最大ボーレートは、次のようになります。1/11.32 µs = 88.36 K ボー

38.5 低消費電力モードでの LPUART

表 357. 低消費電力モードが LPUART に与える影響

モード	説明
SLEEP	影響はありません。LPUART 割込みによって、デバイスは SLEEP モードを終了します。
STOP(1)	LPUART レジスタの内容は保持されます。 STOP モードでも動作しているオシレータで LPUART のクロックが駆動されていれば、LPUART でマイクロコントローラを STOP モードからウェイクアップできます。
STANDBY	LPUART ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

1. 特定のペリフェラル·インスタンスで STOP モードからのウェイクアップがサポートされているかどうかを確認するには セクション 38.3：LPUART の実装を参照してください。特定の STOP モードで動作しないインスタンスは、その STOP モードに移行する前に無効化する必要があります。

38.6 LPUART 割込み

すべての LPUART 割込みリクエストの詳細な説明については、表 358 を参照してください。

表 358. LPUART 割込みリクエスト

割込みベクタ	割込みイベント	イベントフラグ	イネーブル制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP(1) モードの終了	STANDBY モードの終了
LPUART	送信データ·レジスタエンプティ	TXE	TXEIE	TDR に書き込む	可能	不可	不可
	送信 FIFO ノットフル	TXFNF	TXFNFIE	TXFIFO フル		不可	
	送信 FIFO エンプティ	TXFE	TXFEIE	TDR に書き込むか TXFRQ に 1 を書き込む		可能	
	送信 FIFO 閾値到達	TXFT	TXFTIE	TDR に書き込む		可能	
	CTS 割込み	CTSIF	CTSIE	CTSCF に 1 を書き込む		不可	
	送信完了	TC	TCIE	TDR に書き込むか TCCF に 1 を書き込む		不可	
	受信データ·レジスタノット·エンプティ（データの読出し可能）	RXNE	RXNEIE	RDR を読み出すか、RXFRQ に 1 を書き込む	可能	可能	
	受信 FIFO ノットエンプティ	RXFNE	RXFNEIE	RXFIFO がエンプティになるまで RDR を読み出すか、RXFRQ に 1 を書き込む		可能	
	受信 FIFO フル	RXFF(2)	RXFFIE	RDR を読み出す		可能	
	受信 FIFO 閾値到達	RXFT	RXFTIE	RDR を読み出す		可能	
	オーバーランエラー検出	ORE	RXNEIE/RXFNEIE	ORECF に 1 を書き込む		不可	
	アイドル·ライン検出	IDLE	IDLEIE	IDLECF に 1 を書き込む		不可	
	パリティ·エラー	PE	PEIE	PECF に 1 を書き込む		不可	
	マルチバッファ通信のノイズ·エラー	NE	EIE	NFCF に 1 を書き込む		不可	
	マルチバッファ通信のオーバーラン·エラー	ORE(3)		ORECF に 1 を書き込む		不可	
	マルチバッファ通信のフレーミング·エラー	FE		FECF に 1 を書き込む		不可	
	キャラクター致	CMF	CMIE	CMCF に 1 を書き込む		不可	
	低消費電力モードからのウェイクアップ	WUF	WUFIE	WUC に 1 を書き込む		可能	

1. ペリフェラル·インスタンスが STOP モードからのウェイクアップ機能をサポートしている場合にのみ、LPUART はデバイスを STOP モードからウェイクアップすることができます。サポート対象の STOP モードのリストについては、セクション 38.3 : LPUART の実装を参照してください。
2. RXFF フラグは、USART が RXFIFO に n 個のデータを置き、USART_RDR に 1 個のデータを置くように n+1 個のデータを受信した場合にアサートされます（n は RXFIFO のサイズ）。STOP モードでは、LPUART_RDR はクロック供給されません。その結果として、LPUART_RDR レジスタには書き込まれず、n 個のデータが受信されて RXFIFO に書き込まれると RXFF 割込みがアサートされます（RXFF フラグはセットされません）。
3. OVRDIS = 0 の場合

38.7 LPUART レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル・レジスタには、ワード（32 ビット）単位でアクセスする必要があります。

38.7.1 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）

アドレス・オフセット：0x00

リセット値：0x0000 0000

FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも、同じレジスタを使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **RXFFIE**：RXFIFO フル割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタの RXFF が 1 になると LPUART 割込みが生成されます。

ビット 30 **TXFEIE**：TXFIFO エンプティ割込み有効
このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタの TXFE が 1 になると LPUART 割込みが生成されます。

ビット 29 **FIFOEN**：FIFO モード有効
このビットは、ソフトウェアによってセット／クリアされます。
0：FIFO モードは無効です。
1：FIFO モードは有効です。

ビット 28 **M1**：ワード長
このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。
M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個
M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個
M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個
このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビット・データ長モードでは、スマートカード・モード、LIN マスタ・モード、および自動ボー・レート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 37.5.20 : RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 38.4.13 : RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割込みが生成されます。

ビット 13 **MME**：ミュート･モード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバ･ウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M = 1 の場合は 9 番目のビット、M = 0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの PE が 1 になると LPUART 割込みが生成されます。

ビット 7 **TXFNFIE**：TXFIFO ノットフル割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TXE/TXFNF=1 のときには LPUART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの TC が 1 になると LPUART 割込みが生成されます。

ビット 5 **RXFNEIE**：RXFIFO ノットエンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの ORE が 1 または RXNE/RXFNE が 1 になると LPUART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの IDLE が 1 になると LPUART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： 送信中に TE ビットにローパルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を低消費電力モードからウェイクアップできません。

このビットがセットされると、LPUART は マイクロコントローラを低消費電力モードからウェイクアップできますが、RCC で LPUART のクロックとして HSI16 または LSE が選択されている必要があります。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低消費電力モードから MCU をウェイクアップできません。

1：LPUART は低消費電力モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロック・ソースは HSI16 または LSE でなければなりません（RCC の章を参照）。

注： 低消費電力モードに入る直前に UESM ビットをセットし、低消費電力モードの終了時にクリアすることが推奨されます。

ビット 0 **UE**：LPUART 有効化

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低消費電力モードです。

1：LPUART は有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには､TE ビットをその前にリセットする必要があり、ソフトウェアは 、UE ビットをリセットする前にLPUART_ISR の TC ビットがセットされるのを待つ必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

38.7.2 LPUART 制御レジスタ 1[オルタネート]（LPUART_CR1）

アドレス･オフセット：0x00

リセット値：0x0000 0000

FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも、同じレジスタを使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	FIFO EN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]				
		rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29 **FIFOEN**：FIFO モード有効

このビットは、ソフトウェアによってセット／クリアされます。

0：FIFO モードは無効です。

1：FIFO モードは有効です。

ビット 28 **M1**：ワード長

このビットはビット 12（M0）と併せて使用して、ワード長を決定する必要があります。ソフトウェアによってセット／クリアされます。

M[1:0] = “00”：スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = “01”：スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = “10”：スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビット･データ長モードでは、スマートカード･モード、LIN マスタ･モード、および自動ボー･レート（0x7F および 0x55 フレーム検出）はサポートされません。

ビット 27:26 予約済みであり、リセット値に保持する必要があります。

ビット 25:21 **DEAT[4:0]**：ドライバ有効アサーション時間

この 5 ビット値は、DE（Driver Enable）信号の有効化からスタートビットの開始までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 37.5.20 : RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]**：ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE（Driver Enable）信号の無効化までの時間を定義します。lpuart_ker_ck クロックサイクルで表されます。詳細については、セクション 38.4.13：RS232 ハードウェアフロー制御および RS485 ドライバ有効を参照してください。

DEDT 時間中に LPUART_TDR レジスタに書込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **CMIE**：キャラクター一致割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CMF ビットがセットされると、LPUART 割込みが生成されます。

ビット 13 **MME**：ミュート･モード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0：レシーバは永続的にアクティブモードです。

1：レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0**：ワード長

このビットはビット 28（M1）と併せて使用して、ワード長を決定します。ソフトウェアによってセット／クリアされます（ビット 28（M1）の説明を参照）。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11 **WAKE**：レシーバ･ウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0：アイドルライン

1：アドレスマーク

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 10 **PCE**：パリティ制御有効

このビットは、ハードウェアのパリティ制御（生成と検出）を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置（M = 1 の場合は 9 番目のビット、M = 0 の場合は 8 番目のビット）に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0：パリティ制御は無効です。

1：パリティ制御は有効です。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 9 **PS**：パリティ選択

このビットは、パリティの生成／検出が有効である（PCE ビットがセットされている）とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0：偶数パリティ

1：奇数パリティ

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **PEIE**：PE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの PE が 1 になると LPUART 割込みが生成されます。

ビット 7 **TXEIE**：送信データ･レジスタ･エンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタで TXE/TXFNF=1 のときには LPUART 割込みが生成されます。

ビット 6 **TCIE**：転送完了割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの TC が 1 になると LPUART 割込みが生成されます。

ビット 5 **RXNEIE**：受信データ･レジスタ･ノット･エンプティ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの ORE が 1 または RXNE/RXFNE が 1 になると LPUART 割込みが生成されます。

ビット 4 **IDLEIE**：IDLE 割込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの IDLE が 1 になると LPUART 割込みが生成されます。

ビット 3 **TE**：トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0：トランスミッタは無効です。

1：トランスミッタは有効です。

注： 送信中に TE ビットにローパルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にプリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUART_ISR レジスタの TEACK ビットをポーリングできます。

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 **RE**：レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0：レシーバは無効です。

1：レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM**：STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を低消費電力モードからウェイクアップできません。

このビットがセットされると、LPUART は マイクロコントローラを低消費電力モードからウェイクアップできますが、RCC で LPUART のクロックとして HSI16 または LSE が選択されている必要があります。

このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART は低消費電力モードから MCU をウェイクアップできません。

1：LPUART は低消費電力モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロック･ソースは HSI16 または LSE でなければなりません（RCC の章を参照）。

注： 低消費電力モードに入る直前に UESM ビットをセットし、低消費電力モードの終了時にクリアすることが推奨されます。

ビット 0 **UE**：LPUART 有効化

このビットがクリアされると、LPUART プリスケーラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUART_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0：LPUART プリスケーラと出力は無効であり、低消費電力モードです。

1：LPUART は有効です。

注： ラインにエラーを生成せずに低消費電力モードに入るためには､TE ビットをその前にリセットする必要があり、ソフトウェアは 、UE ビットをリセットする前にLPUART_ISR の TC ビットがセットされるのを待つ必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャネルを無効にする必要があります。

38.7.3 LPUART 制御レジスタ 2（LPUART_CR2）

アドレス･オフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOP[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
rw		rw	rw								rw				

ビット 31:24 **ADD[7:0]**：LPUART ノードのアドレス

これらのビットは、ミュート･モードでは LPUART ノードのアドレス、低消費電力モードと RUN モードでは認識対象となるキャラクタ･コードを表します。

- ミュート･モード：マルチプロセッサ通信で、4 ビットまたは7 ビットのアドレス･マーク検出によってミュート･モードからウェイクアップするときに使用します。トランスミッタで送信するキャラクタの MSB は 1 であることが必要です。4 ビットのアドレス･マーク検出では、ADD[3:0] ビットのみを使用します。
- 低消費電力モード：キャラクター致によって低消費電力モードからウェイクアップするときに使用します。
  WUS[1:0] を 0b00 にプログラムしている場合（アドレス一致時に WUF をアクティブにする設定）、ADD[6:0] または ADD[3:0] ビットフィールド（どちらであるかは ADDM7 ビットで指定）にプログラムしたキャラクタと受信キャラクタが一致していて、WUFIE ビットのセットによって WUF 割込みが有効になっていれば、低消費電力モードからのウェイクアップが実行されます。トランスミッタで送信するキャラクタの MSB は 1 であることが必要です。
- ミュート･モードを無効にした RUN モード（ModBus プロトコルでのブロック終端検出など）：受信キャラクタ全体（8 ビット）を ADD[7:0] と比較し、一致していれば CMF フラグがセットされます。CMIE ビットがセットされている場合、割込みが生成されます。

これらのビットフィールドは、受信が無効のとき（RE = 0）または USART が無効のとき（UE = 0）にのみ書き込むことができます。

ビット 23:20 予約済みであり、リセット値に保持する必要があります。

ビット 19 **MSBFIRST**：MSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。

0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1：スタートビットに続いて、データは MSB（ビット 7/8）から順に送受信されます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 18 **DATAINV**：バイナリデータ反転

このビットは、ソフトウェアによってセット／クリアされます。

0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1 = H、0 = L）

1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1 = L、0 = H）パリティ・ビットも反転されます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 17 **TXINV**：TX ピン･アクティブ･レベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：TX ピン信号は標準ロジック･レベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：TX ピン信号の値を反転します（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 16 **RXINV**：RX ピン･アクティブ･レベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0：RX ピン信号は標準ロジック･レベルを使用して機能します（V_{DD} = 1/アイドル、Gnd = 0/マーク）。

1：RX ピン信号の値を反転します（V_{DD} = 0/マーク、Gnd = 1/アイドル）。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 15 **SWAP**：TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0：TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1：TX および RX ピンの機能はスワップされます。これにより、別の UART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13:12 **STOP[1:0]**：ストップビット

このビットは、ストップビットのプログラミングに使用します。

00：1 個のストップビット

01：予約済み。

10：2 個のストップビット

11：予約済みです。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 11:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **ADDM7**：7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0：4 ビットアドレス検出

1：7 ビットアドレス検出（8 ビットデータモード）

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： 7 ビットおよび 9 ビット･データ･モードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットのアドレス（ADD[5:0] および ADD[7:0]）に対して行われます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

38.7.4 LPUART 制御レジスタ 3（LPUART_CR3）

アドレス･オフセット：0x08

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	WUFIE	WUS[1:0]		Res.	Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

ビット 31:29 **TXFTCFG[2:0]**：TXFIFO 閾値設定

000：TXFIFO はその深さの 1/8 に達します。
001：TXFIFO はその深さの 1/4 に達します。
110：TXFIFO はその深さの 1/2 に達します。
011：TXFIFO はその深さの 3/4 に達します。
100：TXFIFO はその深さの 7/8 に達します。
101：TXFIFO は空になります。
残りの組み合わせ：予約済み。

ビット 28 **RXFTIE**：RXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：受信 FIFO が RXFTCFG でプログラムされた閾値に達すると、LPUART 割込みが生成されます。

ビット 27:25 **RXFTCFG[2:0]**：受信 FIFO 閾値設定

000：受信 FIFO はその深さの 1/8 に達します。
001：受信 FIFO はその深さの 1/4 に達します。
110：受信 FIFO はその深さの 1/2 に達します。
011：受信 FIFO はその深さの 3/4 に達します。
100：受信 FIFO はその深さの 7/8 に達します。
101：受信 FIFO はフルになります。
残りの組み合わせ：予約済み。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TXFTIE**：TXFIFO 閾値割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：TXFIFO が TXFTCFG でプログラムされた閾値に達すると、LPUART 割込みが生成されます。

ビット 22 **WUFIE**：低消費電力モードからのウェイクアップ割込み有効

このビットは、ソフトウェアによってセット／クリアされます。
0：割込みは禁止されています。
1：LPUART_ISR レジスタの WUF が 1 になると LPUART 割込みが生成されます。

注： **WUFIE は、低消費電力モードに入る前にセットする必要があります。**
LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 37.4 : USART の実装を参照してください。

ビット 21:20 **WUS[1:0]**：低消費電力モードからのウェイクアップ割込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します（低消費電力モードからのウェイクアップフラグ）。

00：WUF はアドレス一致時に有効になります（ADD[7:0] および ADDM7 による定義に従って）。

01：予約済み

10：WUF はスタートビット検出時に有効になります。

11：WUF は RXNE 時に有効になります。

このビットフィールドは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、リセット値に保持する必要があります。セクション 37.4 : USART の実装を参照してください。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **DEP**：ドライバ有効極性選択

0：DE 信号はアクティブハイです。

1：DE 信号はアクティブローです。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 14 **DEM**：ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0：DE 機能は無効です。

1：DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 13 **DDRE**：受信エラー時 DMA 無効

0：受信エラーの場合、DMA は無効になりません。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないので、エラーのあるデータは転送されず（DMA リクエストなし）、次の正しい受信データが転送されます。

1：受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか（DMAR=0）、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： 受信エラーは、パリティ･エラー、フレーミング･エラー、またはノイズ･エラーです。

ビット 12 **OVRDIS**：オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0：オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1：オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが LPUART_RDR レジスタの前の内容に上書きされます。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTSIE**：CTS 割込み有効

0：割込みは禁止されています。

1：LPUART_ISR レジスタの CTSIF が 1 になると割込みが生成されます。

ビット 9 **CTSE**：CTS 有効化

0：CTS ハードウェアフロー制御が無効です。

1：CTS モードが有効です。データは CTS 入力がデアサート（0 固定）されている場合にのみ転送されます。データの送信中に CTS 入力がアサートされた場合、送信が完了してから停止します。CTS がアサートされている間にデータがデータ･レジスタに書き込まれた場合、CTS がデアサートされるまで送信は延期されます。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 8 **RTSE**：RTS 有効化

0：RTS ハードウェアフロー制御が無効です。

1：RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、RTS 出力がデアサートされます（0 にプルされます）。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 7 **DMAT**：DMA 有効トランスミッタ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードは送信に有効です。

0：DMA モードは送信に無効です。

ビット 6 **DMAR**：DMA 有効レシーバ

このビットは、ソフトウェアでセット / クリアされます。

1：DMA モードが受信に有効です。

0：DMA モードが受信に無効です。

ビット 5:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **HDSEL**：半二重選択

単線半二重モードの選択です。

0：半二重モードは選択されません。

1：半二重モードが選択されます。

このビットは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

ビット 2:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EIE**：エラー割込み有効化

エラー割込み有効化ビットは、フレーミング･エラー、オーバーラン･エラー、またはノイズ･フラグ（LPUART_ISR レジスタで FE = 1、ORE = 1、または NE = 1）が発生したときの割込み生成を有効にするために必要です。

0：割込みは禁止されています。

1：LPUART_ISR レジスタの FE、ORE、または NE が 1 になると割込みが生成されます。

38.7.5 LPUART ボー･レート･レジスタ（LPUART_BRR）

このレジスタは、LPUART が無効（UE = 0）のときのみ書き込むことができます。自動ボー･レート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット：0x0C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **BRR[19:0]**：LPUART ボーレート

注： LPUART_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

ただし、LPUART_BRR は ≥ 0x300 でなければならず、LPUART_BRR は 20 ビットであり、高い fck 値を使用して高いボーレートを生成するときには、注意が必要です。fck は [3 x ボーレート～4096 x ボーレート] の範囲内でなければなりません。

38.7.6 LPUART リクエストレジスタ（LPUART_RQR）

アドレスオフセット：0x18

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
											w	w	w	w	

ビット 31:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TXFRQ**：送信データー掃リクエスト

このビットは FIFO モードが有効なときに使用されます。FIFO 全体を一掃するために TXFRQ ビットがセットされます。これによって TXFE フラグ（TXFIFO エンプティ、LPUART_ISR レジスタのビット 23）がセットされます。

注： FIFO モードでは、一掃リクエスト中にデータがデータレジスタに書き込まれないようにするために、TxFIFO が空になるまで TXFNF フラグはリセットされます。

ビット 3 **RXFRQ**：受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ**：ミュート･モード･リクエスト

このビットに 1 を書き込むと、LPUART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ**：ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注： アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 予約済みであり、リセット値に保持する必要があります。

38.7.7 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0080 00C0

同じレジスタが FIFO モード有効（このセクション）でも、FIFO モード無効（次のセクション）でも使用できます。

FIFO モードが有効な場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFE	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY
				r	r		r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **TXFT**：TXFIFO 閾値フラグ

このビットは、TXFIFO が LPUART_CR3 レジスタの TXFTCFG でプログラムされた閾値に達したとき、すなわち、TXFIFO に TXFTCFG の空き場所ができたときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの TXFTIE ビット が 1（ビット 31）の場合、割込みが生成されます。

0：TXFIFO はプログラムされた閾値に達していません。

1：TXFIFO はプログラムされた閾値に達しました。

ビット 26 **RXFT**：RXFIFO 閾値フラグ

このビットは、RXFIFO が LPUART_CR3 レジスタの RXFTCFG でプログラムされた閾値に達したとき、すなわち、受信 FIFO に RXFTCFG 分のデータが入ったときに、ハードウェアによってセットされます。LPUART_CR3 レジスタの RXFTIE ビット が 1（ビット 27）の場合、割込みが生成されます。

0：受信 FIFO はプログラムされた閾値に達していません。

1：受信 FIFO はプログラムされた閾値に達しました。

ビット 25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **RXFF**：RXFIFO フル

このビットは、受信したデータの数が RXFIFO サイズ + 1 と一致したときに（RXFIFO がフルで、LPUART_RDR レジスタに 1 個のデータ）、ハードウェアによってセットされます。

LPUART_CR1 レジスタの RXFFIE ビット が 1 の場合、割込みが生成されます。

0：RXFIFO はフルではありません。

1：RXFIFO はフルです。

ビット 23 **TXFE**：TXFIFO エンプティ

このビットは、TXFIFO が空のとき、ハードウェアによってセットされます。TXFIFO に少なくとも 1 データが入ったとき、このフラグはクリアされます。TXFE フラグは、LPUART_RQR レジスタのビット TXFRQ（ビット 4）に 1 を書き込むことによってセットすることもできます。

LPUART_CR1 レジスタの TXFEIE ビット が 1（ビット 30）の場合、割込みが生成されます。

0：TXFIFO は空ではありません。

1：TXFIFO は空です。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE= 0 を書き込んだ後、TE = 1 を書き込むことによってアイドル･フレーム･リクエストを生成するときの TE = 0 の最小期間を確保する目的で使用できます。

ビット 20 **WUF**：低消費電力モードからのウェイクアップ･フラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE が 1 の場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 19 **RWU**：レシーバのミュート･モードからのウェイクアップ･フラグ

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されます。

1：LPUART_RQR レジスタの SBKRQ ビットをセットすることでブレーク･キャラクタをリクエストします。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE が 1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジー･フラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの CTSIE が 1 の場合、割込みが生成されます。

0：CTS ステータス･ラインでの変化はありません。

1：CTS ステータス･ラインに変化がありました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXFNF**：TXFIFO ノット･フル

TXFIFO がフルではない、つまり LPUART_TDR にデータを書き込めるとき、TXFNF はハードウェアによってセットされます。LPUART_TDR への書込みごとにデータが TXFIFO に格納されます。このフラグは TXFIFO がフルになるまでセットされたままになります。TXFIFO がフルになると、このフラグはクリアされ、データを LPUART_TDR に書き込むことができないことを示します。

一掃リクエスト中、TXFIFO が空になるまで、TXFNF はリセットに維持されます。一掃リクエストを（TXFRQ ビットをセットすることによって）送信した後、TXFIFO に書き込む前に TXFNF フラグをチェックするべきです（TXFNF および TXFE は同時にセットされます）。

LPUART_CR1 レジスタの TXFNFIE ビット が 1 の場合、割込みが生成されます。

0：データレジスタはフル／送信 FIFO はフルです。

1：データレジスタ／送信 FIFO はフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXFF がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。LPUART_ICR レジスタの TCCF に 1 を書き込むか、LPUART_TDR レジスタに書き込むことによってソフトウェアでクリアされます。

LPUART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXFNE**：RXFIFO ノット･フル

RXFNE ビットは、RXFIFO が空でなく、したがってデータが LPUART_RDR レジスタから読み出せるときに、ハードウェアによってセットされます。LPUART_RDR からの読出しのたびに、RXFIFO の 1 つの場所が解放されます。RXFIFO が空になると、クリアされます。

RXFNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUART_CR1 レジスタの RXFNEIE が 1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドル･ライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE が 1 の場合、割込みが生成されます。LPUART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXFNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュート･モードが有効な場合（MME = 1）、LPUART がミュートでなければ（RWU = 0）、WAKE ビットで選択したミュート･モードに関係なく、IDLE がセットされます。RWU = 1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーラン･エラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、

現在シフトレジスタに受信中のデータを LPUART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの RXFNEIE が 1 または EIE が 1 の場合、割込みが生成されます。

0：オーバーランエラーはありません。

1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：スタート･ビット･ノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0：ノイズは検出されていません。

1：ノイズが検出されました。

注： このビットは、割込みを生成する RXFNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

このエラーはLPUART_RDR 内のキャラクタに関連します。

ビット 1 **FE**：フレーミング･エラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。

LPUART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。

0：フレーミングエラーは検出されていません。

1：フレーミングエラーまたはブレークキャラクタが検出されました。

注： このエラーはLPUART_RDR 内のキャラクタに関連します。

ビット 0 **PE**：パリティ･エラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。

0：パリティエラーはありません。

1：パリティエラー

注： このエラーはLPUART_RDR 内のキャラクタに関連します。

38.7.8 LPUART 割込みとステータスレジスタ [オルタネート]（LPUART_ISR）

アドレスオフセット：0x1C

リセット値：0x0000 00C0

同じレジスタが FIFO モード有効（前のセクション）でも、FIFO モード無効（このセクション）でも使用できます。

FIFO モードが無効の場合

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
					r	r		r	r	r	r	r	r	r	r

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **REACK**：受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、低消費電力モードに入る前に、LPUART が受信できる状態であることを確認できます。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 21 **TEACK**：送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

LPUART_CR1 レジスタで TE= 0 を書き込んだ後、TE = 1 を書き込むことによってアイドル･フレーム･リクエストを生成するときの TE = 0 の最小期間を確保する目的で使用できます。

ビット 20 **WUF**：低消費電力モードからのウェイクアップ･フラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUART_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの WUFIE が 1 の場合、割込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 19 **RWU**：レシーバのミュート･モードからのウェイクアップ･フラグ

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア／セットされます。ミュートモード制御シーケンス（アドレスまたは IDLE）は、LPUART_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUART_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0：レシーバはアクティブモードです。

1：レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 18 **SBKF**：ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUART_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0：ブレークキャラクタが送信されます。

1：LPUART_RQR レジスタの SBKRQ ビットをセットすることでブレーク･キャラクタをリクエストします。

ビット 17 **CMF**：キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR1 レジスタの CMIE が 1 の場合、割込みが生成されます。

0：キャラクター致は検出されていません。

1：キャラクター致が検出されました。

ビット 16 **BUSY**：ビジー･フラグ

このビットは、ハードウェアによってセット／リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0：LPUART はアイドルです（受信なし）。

1：受信中です。

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CTS**：CTS フラグ

このビットは、ハードウェアによってセット／リセットされます。CTS 入力ピンのステータスの反転コピーです。

0：CTS ラインはセットされました。

1：CTS ラインはリセットされました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 9 **CTSIF**：CTS 割込みフラグ

このビットは、CTSE ビットがセットされていた場合、CTS 入力がトグルしたときにハードウェアによってセットされます。LPUART_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUART_CR3 レジスタの CTSIE が 1 の場合、割込みが生成されます。

0：CTS ステータス･ラインでの変化はありません。

1：CTS ステータス･ラインに変化がありました。

注： ハードウェア･フロー制御機能がサポートされない場合、このビットは予約済みであり、リセット値に保持されます。

ビット 8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **TXE**：送信データ･レジスタ･エンプティ／TXFIFO ノット･フル

TXE は、LPUART_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUART_TDR レジスタへの書込みによってクリアされます。

LPUART_CR1 レジスタの TXEIE ビット =1 の場合、割込みが生成されます。

0：データレジスタはフルです。

1：データレジスタはフルではありません。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 **TC**：送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。LPUART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。LPUART_ICR レジスタの TCCF に 1 を書き込むか、LPUART_TDR レジスタに書き込むことによってソフトウェアでクリアされます。

LPUART_CR1 レジスタの TCIE が 1 の場合、割込みが生成されます。

0：送信は完了していません。

1：送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 **RXNE**：読出しデータ･レジスタ･ノット･エンプティ

RXNE ビットは、LPUART_RDR シフトレジスタの内容が LPUART_RDR レジスタに転送されると、ハードウェアによってセットされます。このフラグは、LPUART_RDR レジスタからの読出しによってクリアされます。RXNE フラグは、LPUART_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUART_CR1 レジスタの RXNEIE が 1 の場合、割込みが生成されます。

0：データは受信されていません。

1：受信データを読み出すことができます。

ビット 4 **IDLE**：アイドル･ライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUART_CR1 レジスタの IDLEIE が 1 の場合、割込みが生成されます。LPUART_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアでクリアされます。

0：アイドルラインは検出されていません。

1：アイドルラインが検出されました。

注： RXNE ビットがセットされるまで（新しいアイドルラインが発生するまで）、IDLE ビットは再びセットされません。

ミュート･モードが有効な場合（MME = 1）、LPUART がミュートでなければ（RWU = 0）、WAKE ビットで選択したミュート･モードに関係なく、IDLE がセットされます。RWU = 1 の場合、IDLE はセットされません。

ビット 3 **ORE**：オーバーラン･エラー

このビットは、RXNE=“1”（FIFO モードが有効な場合は RXFF=“1”）のときに、
現在シフト･レジスタで受信中のデータを LPUART_RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアでクリアされます。
LPUART_CR1 レジスタの RXNEIE が 1 または EIE が 1 の場合、割込みが生成されます。
0：オーバーランエラーはありません。
1：オーバーランエラーが検出されました。

注： このビットがセットされると、LPUART_RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割込みが生成されます。

LPUART_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます（オーバーラン検出なし）。

ビット 2 **NE**：スタート･ビット･ノイズ検出フラグ

このビットは、受信フレームのスタートビットでノイズが検出されるとハードウェアによってセットされます。LPUART_ICR レジスタの NECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。
0：ノイズは検出されていません。
1：ノイズが検出されました。

注： このビットは、割込みを生成する RXNE ビットと同時に出現するため、割込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NE フラグがセットされると、割込みが生成されます。

ビット 1 **FE**：フレーミング･エラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUART_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。
スマートカードモードでデータを送信しているとき、送信時、成功せずに（カードがデータフレームを NACK）最大送信試行回数に達すると、このビットがセットされます。
LPUART_CR1 レジスタの EIE=1 の場合、割込みが生成されます。
0：フレーミングエラーは検出されていません。
1：フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 **PE**：パリティ･エラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUART_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。
LPUART_CR1 レジスタの PEIE=1 である場合、割込みが生成されます。
0：パリティエラーはありません。
1：パリティエラー

38.7.9 LPUART 割込みフラグクリアレジスタ（LPUART_ICR）

アドレスオフセット：0x20

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.
											w			w	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
						w			w		w	w	w	w	w

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **WUCF**：低消費電力モードからのウェイクアップ･フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの WUF フラグがクリアされます。
注： LPUART が STOP モードからのウェイクアップをサポートしない場合、このビットは予約済みであり、リセット値に保持されます。セクション 37.4：USART の実装を参照してください。

ビット 19:18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **CMCF**：キャラクター致フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの CMF フラグがクリアされます。

ビット 16:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **CTSCF**：CTS フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの CTSIF フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TCCF**：送信完了フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **IDLECF**：アイドル･ライン検出フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF**：オーバーラン･エラー･フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NECF**：ノイズ検出フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの NE フラグがクリアされます。

ビット 1 **FECF**：フレーミング･エラー･フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF**：パリティ･エラー･フラグのクリア
このビットに 1 を書き込むと、LPUART_ISR レジスタの PE フラグがクリアされます。

38.7.10 LPUART 受信データレジスタ（LPUART_RDR）

アドレスオフセット：0x24

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
							r	r	r	r	r	r	r	r	r

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **RDR[8:0]**：受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入力シフトレジスタと内部バスとの間にパラレルインタフェースを提供します（図 558 を参照）。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

38.7.11 LPUART 送信データレジスタ（LPUART_TDR）

アドレスオフセット：0x28

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
							rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **TDR[8:0]**：送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します（図 558 を参照）。

パリティを有効にして（LPUART_CR1 レジスタの PCE ビットに 1 をセット）送信しているとき、MSB（データ長に応じてビット 7 または 8）に書き込まれた値は、パリティによって置き換えられるため、無効です。

注： このレジスタは、TXE/TXFNF = 1 のときのみ書き込む必要があります。

38.7.12 LPUART プリスケーラ･レジスタ（LPUART_PRESC）

このレジスタは、LPUART が無効（UE = 0）のときのみ書き込むことができます。

アドレス･オフセット：0x2C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PRESCALER[3:0]**：クロックプリスケーラ

LPUART 入力クロックは、以下のようにプリスケーラによって分周できます。

0000：入力クロックは分周されません。
0001：入力クロックが 2 分周されます。
0010：入力クロックが 4 分周されます。
0011：入力クロックが 6 分周されます。
0100：入力クロックが 8 分周されます。
0101：入力クロックが 10 分周されます。
0110：入力クロックが 12 分周されます。
0111：入力クロックが 16 分周されます。
1000：入力クロックが 32 分周されます。
1001：入力クロックが 64 分周されます。
1010：入力クロックが 128 分周されます。
1011：入力クロックが 256 分周されます。
残りの組み合わせ：予約済み。

注： PRESCALER が許容される値と異なる値でプログラムされたとき、プログラムされるプリスケーラ値は 1011 になります。すなわち、入力クロックは 256分周されます。

38.7.13 LPUART レジスタマップ

次の表に、LPUART のレジスタマップとリセット値を示します。

表 359. LPUART レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	LPUART_CR1 FIFO モードが有効な場合	RXFFIE	TXFEIE	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]					Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値	0	0	0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x00	LPUART_CR1 FIFO モードが無効の場合	Res.	Res.	FIFOEN	M1	Res.	Res.	DEAT[4:0]					DEDT[4:0]					Res.	CMIE	MME	M0	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	UESM	UE
	リセット値			0	0			0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	LPUART_CR2	ADD[7:0]								Res.	Res.	Res.	Res.	MSBFIRST	DATAINV	TXINV	RXINV	SWAP	Res.	STOP [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDM7	Res.	Res.	Res.	Res.
	リセット値	0	0	0	0	0	0	0	0					0	0	0	0	0		0	0								0				
0x08	LPUART_CR3	TXFTCFG[2:0]			RXFTIE	RXFTCFG[2:0]			Res.	TXFTIE	WUFIE	WUS [1:0]		Res.	Res.	Res.	Res.	DEP	DEM	DDRE	OVRDIS	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res.	Res.	HDSEL	Res.	Res.	EIE
	リセット値	0	0	0	0	0	0	0		0	0	0	0					0	0	0	0		0	0	0	0	0			0			0
0x0C	LPUART_BRR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BRR[19:0]																			
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10 ~0x14	予約済みです。																																
0x18	LPUART_RQR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXFRQ	RXFRQ	MMRQ	SBKRQ	Res.
	リセット値																												0	0	0	0	
0x1C	LPUART_ISR FIFO モードが有効な場合	Res.	Res.	Res.	Res.	TXFT	RXFT	Res.	RXFF	TXFF	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NE	FE	PE
	リセット値					0	0		0	1	0	0	0	0	0	0	0						0	0		1	1	0	0	0	0	0	0
0x1C	LPUART_ISR FIFO モードが無効の場合	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	REACK	TEACK	WUF	RWU	SBKF	CMF	BUSY	Res.	Res.	Res.	Res.	Res.	CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NE	FE	PE
	リセット値										0	0	0	0	0	0	0						0	0		1	1	0	0	0	0	0	0
0x20	LPUART_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUCF	Res.	Res.	CMCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTSCF	Res.	Res.	TCCF	Res.	IDLECF	ORECF	NECF	FECF	PECF
	リセット値												0			0								0			0		0	0	0	0	0
0x24	LPUART_RDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0
0x28	LPUART_TDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDR[8:0]								
	リセット値																								0	0	0	0	0	0	0	0	0

表 359. LPUART レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x2C	LPUART_ PRESC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PRESCALER[3:0]			
	リセット値																													0	0	0	0

レジスタ境界アドレスについては、セクション 2.2 : メモリ構成を参照してください。

39 シリアル･ペリフェラル･インタフェース／インター IC サウンド（SPI/I2S）

39.1 概要

SPI/I²S インタフェースを使用して、SPI プロトコルまたは I^2S オーディオプロトコルに基づき外部デバイスと通信することができます。SPI または I^2S モードはソフトウェアによって選択可能です。デバイスのリセット後は、デフォルトで SPI モトローラモードが選択されます。

SPI（シリアルペリフェラルインタフェース）プロトコルは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタとして設定することも可能で、その場合、外部スレーブデバイスに通信クロック（SCK）を供給します。このインタフェースは、マルチマスタ設定で動作することもできます。

インター IC サウンド(I^2S) プロトコルも、同期シリアル通信インタフェースです。スレーブまたはマスタ･モードによる、半二重通信が可能です。フィリップス I^2S 規格、MSB/LSB 詰め規格、PCM 規格など、4 種類のオーディオ規格に対応できます。

39.2 SPI の主な機能

- マスタまたはスレーブ動作
- 3 本のラインでの全二重同期転送
- 2 本のラインでの半二重同期転送（双方向データライン有り）
- 2 本のラインでの単方向同期転送（単方向データライン有り）
- 4 ~ 16 ビットデータサイズ選択
- マルチマスタモード機能
- 8 個のマスタモードボーレートプリスケーラ（最大周波数 $f_{PCLK}/2$）
- スレーブモード周波数（最大周波数 $f_{PCLK}/2$）
- マスタとスレーブの両方に対するハードウェア／ソフトウェアによる NSS 管理 : マスタ／スレーブ動作の動的切り替え
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能（MSB ファースト／LSB ファーストのシフト）
- 専用の送受信フラグ（割込み機能付き）
- SPI バスビジーステータスフラグ
- SPI モトローラモードをサポート
- ハードウェア CRC 機能による信頼性の高い通信 :
  - Tx モードでは CRC 値を最終バイトとして送信可能
  - 最終受信バイトに対する CRC エラーの自動チェック
- マスタモードの障害、オーバーランの各フラグ（割込み機能付き）
- CRC エラーフラグ
- DMA 機能付きの 2 つの 32 ビット内蔵Rx および Tx FIFO
- 拡張 TI および NSS パルスモードのサポート

39.3 I²S の主な機能

- 半二重通信（トランスミッタまたはレシーバのみ）
- マスタまたはスレーブ動作
- 正確なオーディオサンプリング周波数（8～192 kHz）を実現するプログラム可能な 8 ビットのリニアプリスケーラ
- 16、24、または 32 ビットのデータフォーマット
- パケットフレームはオーディオチャネルによって 16 ビット（16 ビットデータフレーム）または 32 ビット（16、24、32 ビットデータフレーム）に固定。
- プログラム可能なクロック極性（定常状態）
- スレーブ送信モードのアンダーランフラグ、受信モード（マスタおよびスレーブ）のオーバーランフラグ、受信モードと送信モード（スレーブの場合のみ）のフレームエラーフラグ
- 送受信用の 16 ビットレジスタ（両チャネルサイドに対して 1 個のデータレジスタ）
- 以下の I²S プロトコルをサポート。
  - フィリップス I²S 規格
  - MSB 詰め規格（左詰め）
  - LSB 詰め規格（右詰め）
  - PCM 規格（16 ビットチャネルフレーム、または 32 ビットチャネルフレームに拡張された 16 ビットデータフレームでの、ショートおよびロングフレーム同期付き）
- データ方向は常に MSB ファースト。
- 送受信（16 ビット幅）用の DMA 機能
- 外部オーディオコンポーネントを駆動するためのマスタクロックを出力可能。周波数比は、256 × F_S（F_S はオーディオサンプリング周波数）に固定。

39.4 SPI/I2S の実装

次の表に、デバイスに組み込まれているすべての SPI インスタンスとその機能を示します。

表 360. STM32G4 シリーズ SPI および SPI/I2S の実装

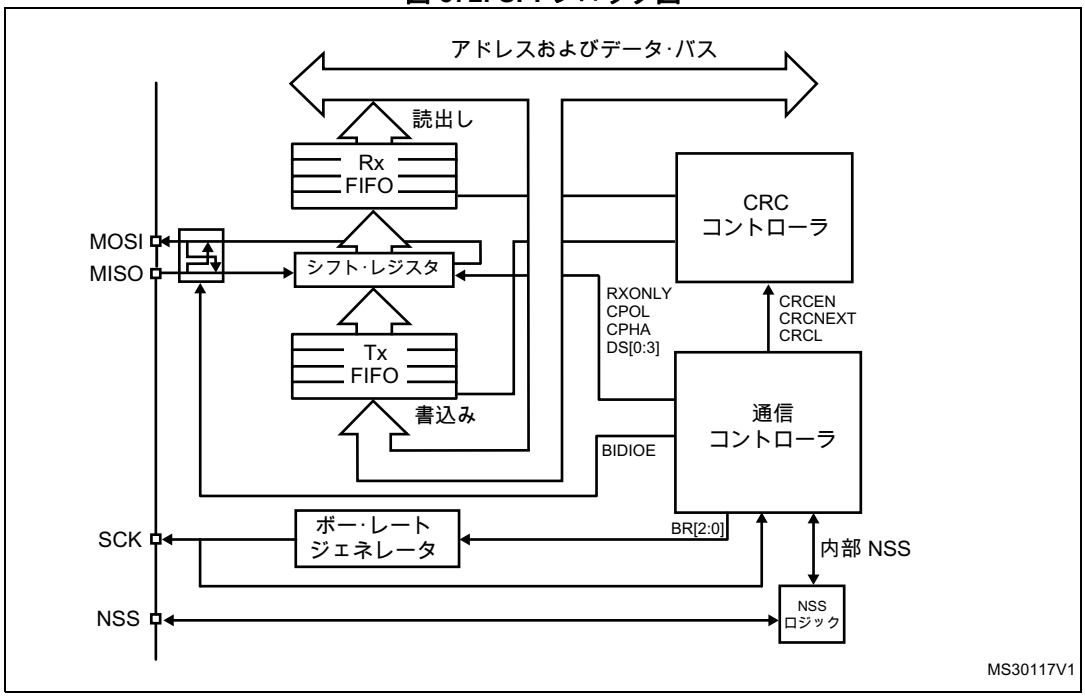
SPI の機能	SPI1	SPI2S2	SPI2S3	SPI4
拡張 NSSP & TI モード	可能	はい	はい	はい
ハードウェア CRC 計算	可能	はい	はい	はい
I²S サポート	不可	はい	はい	不可
設定可能なデータ·サイズ	4～16 ビット	4～16 ビット	4～16 ビット	4～16 ビット
Rx/Tx FIFO サイズ	32 ビット	32 ビット	32 ビット	32 ビット
低電力スリープからのウェイクアップ機能	可能	はい	はい	はい

39.5 SPI の機能詳細

39.5.1 概要

SPI では、MCU と外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用の SPI 割込みを使用することで、通信を管理することができます。SPI の主要要素およびそれらの相互作用を以下のブロック図（図 572 に示します。

図 572. SPI ブロック図



4 本の I/O ピンが外部デバイスとの SPI 通信専用に使用されます。

- **MISO**：マスタ･イン／スレーブ･アウト･データ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使用されます。
- **MOSI**：マスタ･アウト／スレーブ･イン･データ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使用されます。
- **SCK**：SPI マスタではシリアル･クロックの出力ピンに、SPI スレーブでは入力ピンに使用されます。
- **NSS**：スレーブ選択用のピンです。このピンは、SPI および NSS の設定に応じて、以下のいずれかに使用できます。
  - 個々の通信用スレーブデバイスを選択する
  - データフレームを同期させる
  - 複数のマスタ間での競合を検出する

  詳細については、セクション 39.5.5：スレーブ選択（NSS）ピンの管理を参照してください。

SPI バスを使用することで、1 つのマスタデバイスと 1 つ以上のスレーブデバイスとの間で通信することができます。バスは 2 本以上の線から成り、1 本はクロック信号用、その他はデータの同期転送用です。SPI ノード間でのデータ交換とそれらのスレーブ選択信号管理に応じて、その他の信号を追加することができます。

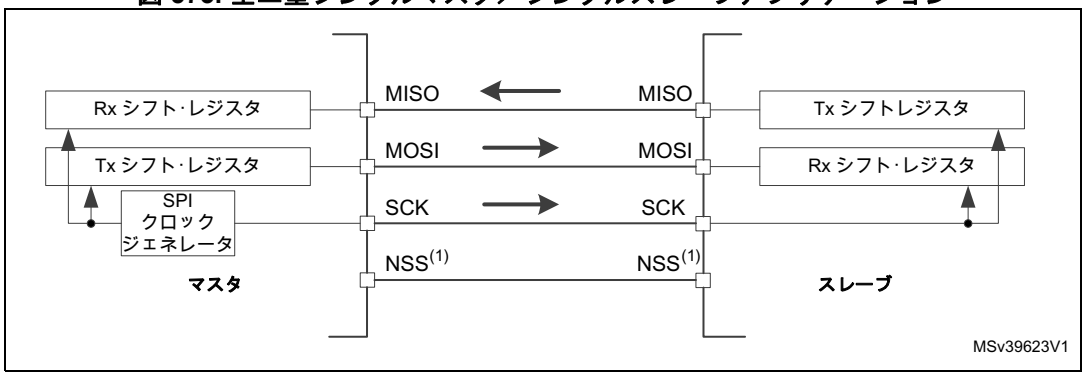
39.5.2 マスタとスレーブの 1 対 1 の通信

SPI を使用することで、MCU は対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2 または 3 本の線（ソフトウェア NSS 管理あり）、あるいは 3 または 4 本の線（ハードウェア NSS 管理あり）が使われます。通信は常にマスタによって開始されます。

全二重通信

SPI は、デフォルトで全二重通信に設定されます。この設定では、マスタおよびスレーブのシフトレジスタは、MOSI ピンと MISO ピンの間に 2 本の単方向ラインを介してリンクされます。SPI 通信の間、データはマスタから供給される SCK クロックのエッジに同期してシフトされます。マスタは、送信すべきデータを MOSI ライン経由でスレーブに送信し、MISO ライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図 573. 全二重シングルマスタ／シングルスレーブアプリケーション

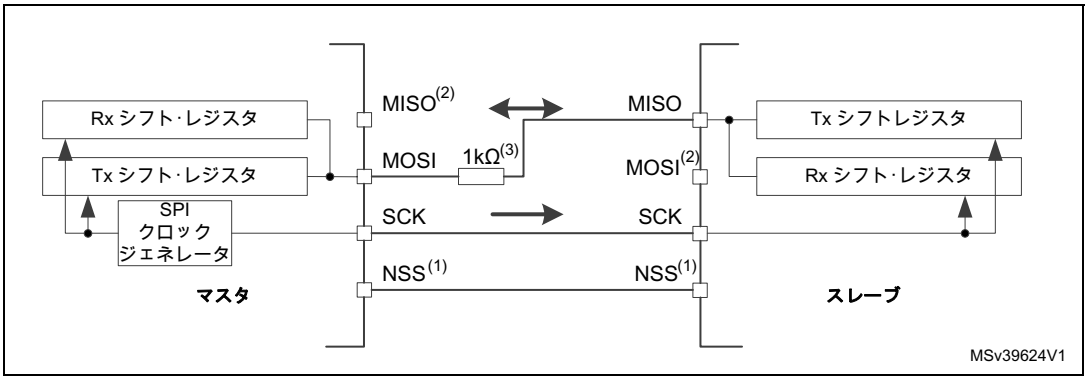


1. NSS ピンを使用して、マスタとスレーブ間のハードウェア制御フローを実現できます。オプションで、ペリフェラルによってピンを未使用状態のままにできます。そのとき、マスタとスレーブ両方に対して内部操作を行う必要があります。詳細については、セクション 39.5.5：スレーブ選択（NSS）ピンの管理を参照してください。

半二重通信

SPIx_CR1 レジスタの BIDIMODE ビットをセットすることで、SPI は半二重モードで通信できます。この設定では、1 本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データは SCK クロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方が SPIx_CR1 レジスタの BDIOE ビットを使用して相互に選択された転送方向となります。この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、他のアプリケーションで使用でき、GPIO として機能します。

図 574. 半二重シングルマスタ／シングルスレーブアプリケーション



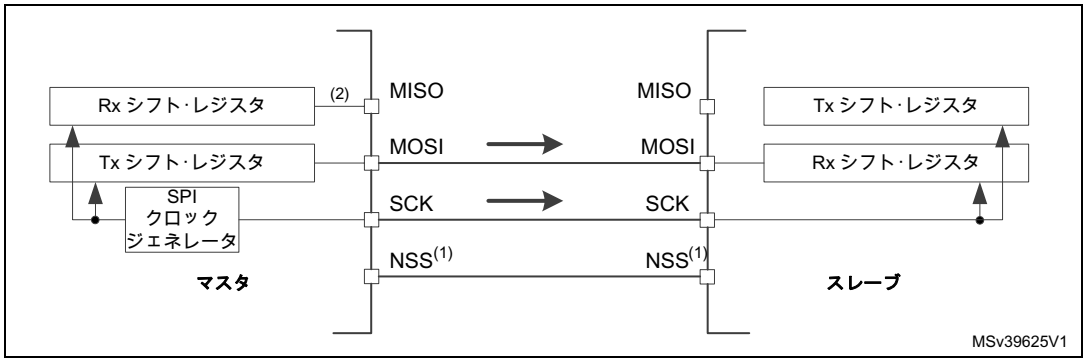
1. NSS ピンを使用して、マスタとスレーブ間のハードウェア制御フローを実現できます。オプションで、ペリフェラルによってピンを未使用状態のままにできます。そのとき、マスタとスレーブ両方に対して内部操作を行う必要があります。詳細については、セクション 39.5.5：スレーブ選択（NSS）ピンの管理を参照してください。
2. この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、GPIO として使用できます。
3. 双方向モードで動作している 2 つのノード間で同期されず、通信方向が変更されて、新しいトランスミッタが共通のデータラインにアクセスし、前のトランスミッタが逆の値をライン上に保持している場合（値は SPI 設定および通信データによります）、重大な事態が発生する可能性があります。そのとき、両方のノードが競合し、次のノードが対応する方向設定に変更するまで、共有データラインで一時的に逆の出力レベルも供給されます。このモードでは MISO ピンと MOSI ピンの間に直列抵抗を挿入して、出力を保護し、この状況で流れる電流を制限することをお奨めします。

単方向通信

SPI は、SPIx_CR1 レジスタの RXONLY ビットを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは 1 ラインのみです。残りの MISO ピンと MOSI ピンのペアは通信には使用されず、標準の GPIO として使用できます。

- **送信専用モード（RXONLY = 0）の場合**：設定は全二重の場合と同じです。アプリケーションは、未使用の入力ピンでキャプチャされた情報を無視する必要があります。このピンは標準の GPIO として使用できます。
- **受信専用モード（RXONLY = 1）の場合**：アプリケーションにて、RXONLY ビットをセットすることによって、SPI 出力機能を無効にできます。スレーブ設定では、MISO 出力が無効化され、ピンを GPIO として使用することができます。スレーブ選択信号がアクティブな間は、スレーブは MOSI ピンからデータを受信し続けます（39.5.5：スレーブ選択（NSS）ピンの管理を参照）。データバッファの設定に応じて、受信データイベントが出現します。マスタ設定では、MOSI 出力が無効化され、ピンを GPIO として使用することができます。SPI が有効である間はクロック信号が生成され続けます。クロックを停止させる唯一の方法は、クロックの設定に応じて、RXONLY ビットまたは SPE ビットをクリアし、MISO ピンからの受信パターンが終了し、データバッファ構造への書込みが行われるまで待つことです。

図 575. 単方向シングルマスタ／シングルスレーブアプリケーション（送信専用モードのマスタ／受信専用モードのスレーブ）



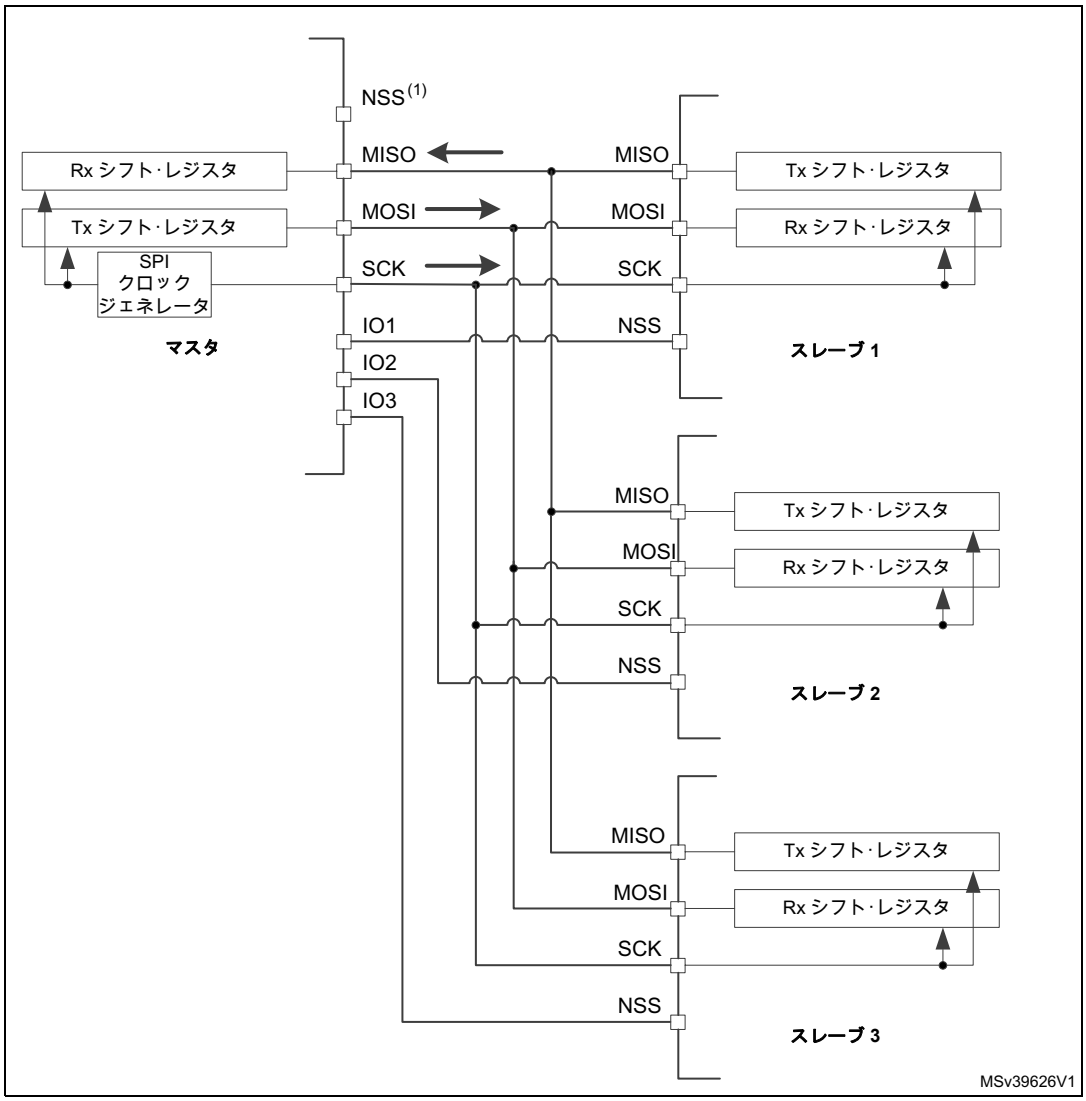
1. NSS ピンを使用して、マスタとスレーブ間のハードウェア制御フローを実現できます。オプションで、ペリフェラルによってピンを未使用状態のままにできます。そのとき、マスタとスレーブ両方に対して内部操作を行う必要があります。詳細については、セクション 39.5.5：スレーブ選択（NSS）ピンの管理を参照してください。
2. 予期しない入力情報はトランスミッタ Rx シフト･レジスタの入力でキャプチャされます。トランスミッタ受信フローに関連するすべてのイベントは、標準の送信専用モード（たとえば、OVR フラグ）では無視する必要があります。
3. この設定では、両方のMISO ピンを GPIO として使用できます。

注： **すべての単方向通信は、トランザクション方向の設定を固定して（双方向モードは BDIO ビットが変化しない限り有効）、別の半二重通信に置き換えることができます。**

39.5.3 標準マルチスレーブ通信

2 つ以上の独立したスレーブがある設定の場合、マスタは GPIO ピンを使用して、各スレーブのチップセレクトラインを管理します（図 576 を参照）。マスタは、スレーブの NSS 入力に接続されている GPIO をローレベルにプルダウンすることによって、スレーブの 1 つを選択する必要があります。これを行うことにより、標準マスタと専用スレーブの通信が確立します。

図 576. マスタと 3 つの独立したスレーブ



1. この設定では、NSS ピンはマスタ側では使用しません。内部的に管理（SSM=1、SSI=1）し、あらゆる MODF エラーを防ぐ必要があります。
2. スレーブの MISO ピンは相互接続されているので、すべてのスレーブにおいて、その MISO ピンの GPIO 設定をオルタネート機能のオープンドレインに設定する必要があります（I/O オルタネート機能の入力／出力セクション（GPIO）を参照）。

39.5.4 マルチマスタ通信

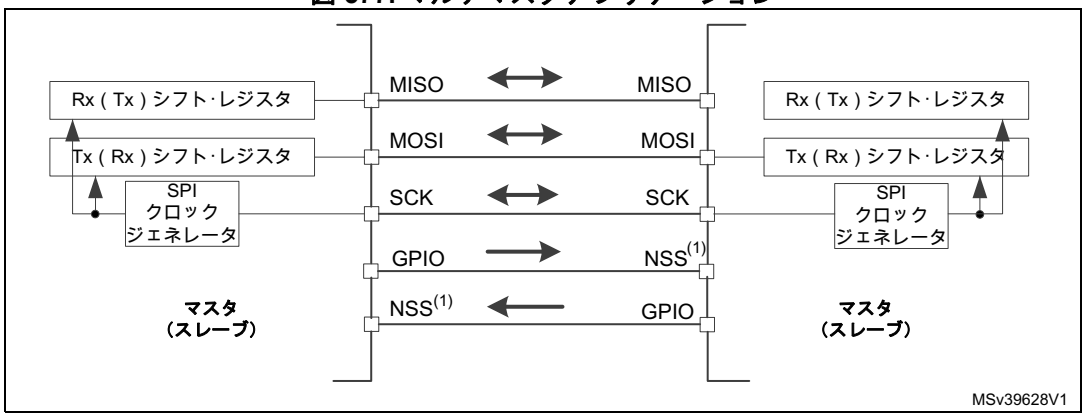
SPI バスが本来マルチマスタ機能向けに設計されていない限り、同時にバスを支配しようとする 2 つのノード間での競合の可能性を検出する内蔵機能を使用できます。この検出では、NSS ピンをハードウェア入力モードで設定して使用します。

共通のデータラインで一度に出力を適用できるノードは 1 つだけであるため、このモードで動作している 2 つ以上の SPI ノードの接続は、不可能です。

ノードが非アクティブであるとき、両ノードはデフォルトでスレーブモードのままとなります。1 つのノードがバスでの制御を得る場合、自身をマスタモードに切り替えて、専用の GPIO ピンを介して他のノードのスレーブ選択入力にアクティブレベルを適用します。セッションが完了すると、アクティブスレーブ選択信号が解放され、バスを一時的に支配するノードがパッシブスレーブモードに戻り、次のセッション開始を待機します。

両ノードのマスタリングリクエストが同時に上がった場合、バス競合イベントが発生します（モードフォールト MODF イベントを参照）。その場合、いくつかの簡単なアービトレーションプロセスを適用できます（たとえば、両方のノードに適用する事前定義された異なるタイムアウトによって次の試行を延期します）。

図 577. マルチマスタアプリケーション



1. NSS ピンは両ノードでハードウェア入力モードで設定されます。アクティブ･レベルにすることで、パッシブ･ノードがスレーブとして設定されるため、MISO ライン出力の制御が可能になります。

39.5.5 スレーブ選択（NSS）ピンの管理

スレーブモードでは、NSS は標準の「チップセレクト」入力として機能し、スレーブをマスタと通信させます。マスタモードでは、NSS は出力としても入力としても使用できます。入力として使用する場合、NSS はマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は 1 つのスレーブのスレーブ選択信号を駆動させることができます。

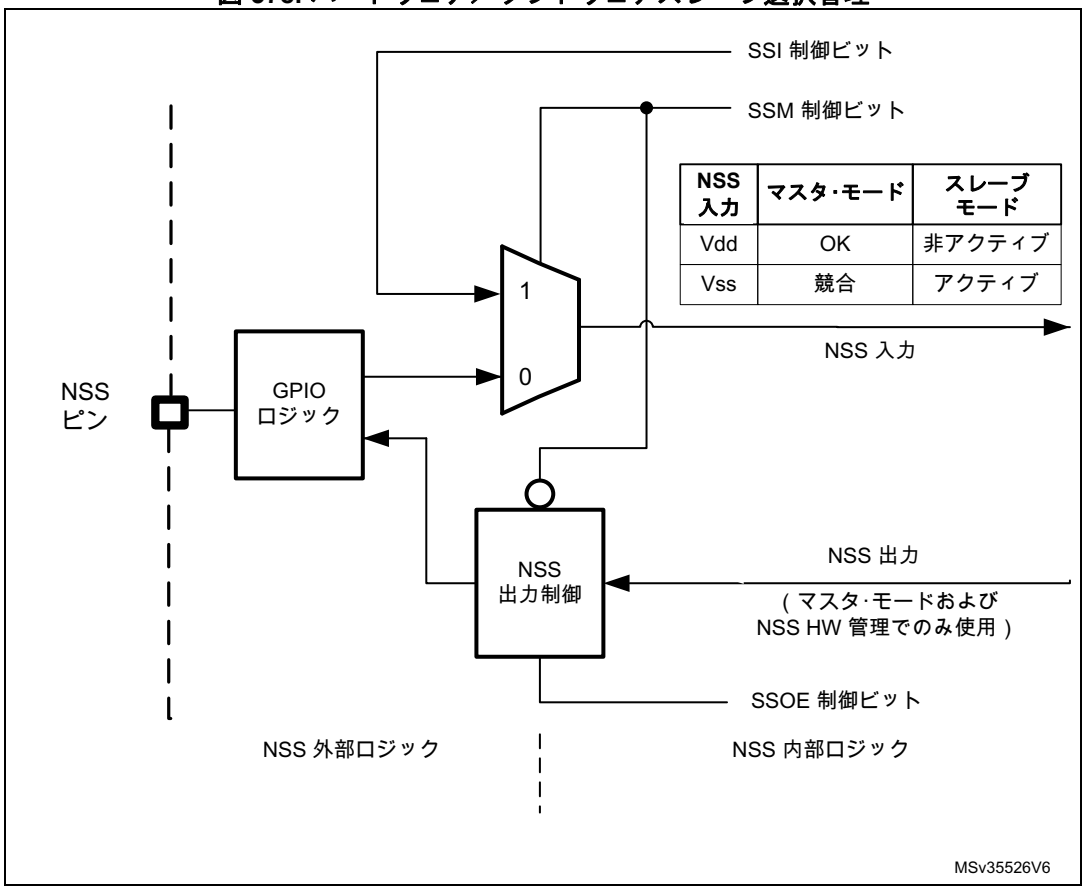
ハードウェアまたはソフトウェアのスレーブ選択管理は、SPIx_CR1 レジスタの SSM ビットを使用して、以下のようにセットすることができます。

- **ソフトウェア NSS 管理（SSM = 1）**：この設定では、スレーブ選択情報は SPIx_CR1 レジスタの SSI ビットの値によって内部で駆動されます。外部 NSS ピンは他のアプリケーションで使用できます。
- **ハードウェア NSS 管理（SSM = 0）**：この場合、2 通りの設定が可能です。次のどちらの設定を使用するかは、NSS 出力設定（SPIx_CR1 レジスタの SSOE ビット）によって決まります。
  - **NSS 出力が有効な場合（SSM = 0、SSOE = 1）**：この設定は、MCU がマスタとしてセットされている場合にのみ使用します。NSS ピンはハードウェアによって管理されます。NSS 信号は、SPI がマスタモードで有効になる（SPE = 1）とすぐにローレベルに駆動さ

れ、SPI が無効化される（SPE = 0）までローレベルに保たれます。パルスは、NSS パルスモードが有効になると連続通信間で生成できます（NSSP = 1）。この NSS 設定では、SPI はマルチマスタ設定で機能させることはできません。

– **NSS 出力が無効な場合 (SSM = 0, SSOE = 0)**：マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、NSS ピンがローレベルにプルダウンされた場合、SPI はマスタモードのフォールト状態に入り、デバイスは自動的にスレーブモードに再設定されます。スレーブモードでは、NSS ピンは標準の「チップセレクト」入力として機能し、NSS ラインがローレベルの間はスレーブが選択されます。

図 578. ハードウェア／ソフトウェアスレーブ選択管理



NSS 入力	マスタ･モード	スレーブ モード
Vdd	OK	非アクティブ
Vss	競合	アクティブ

39.5.6 通信フォーマット

SPI 通信中は受信と送信の操作が同時に行われます。シリアルクロック（SCK）は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従う必要があります。

クロックの位相および極性の制御

SPIx_CR1 レジスタの CPOL ビットと CPHA ビットを使用することによって、考えられる 4 つのタイミングの関係をソフトウェアで選択できます。CPOL（クロック極性）ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOL がリセットされると、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされると、SCK ピンはハイレベルのアイドル状態になります。

CPHA ビットがセットされると､SCK ピンの 2 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがリセットされていれば立ち下がりエッジ、CPOL ビットがセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。CPHA ビットがリセットされている場合、SCK ピンの 1 番目のエッジがトランザクションの最初のデータビットをキャプチャします（CPOL ビットがセットされていれば立ち下がりエッジ、CPOL ビットがリセットされていれば立ち上がりエッジ）。データは、この種のクロック遷移が発生するたびにラッチされます。

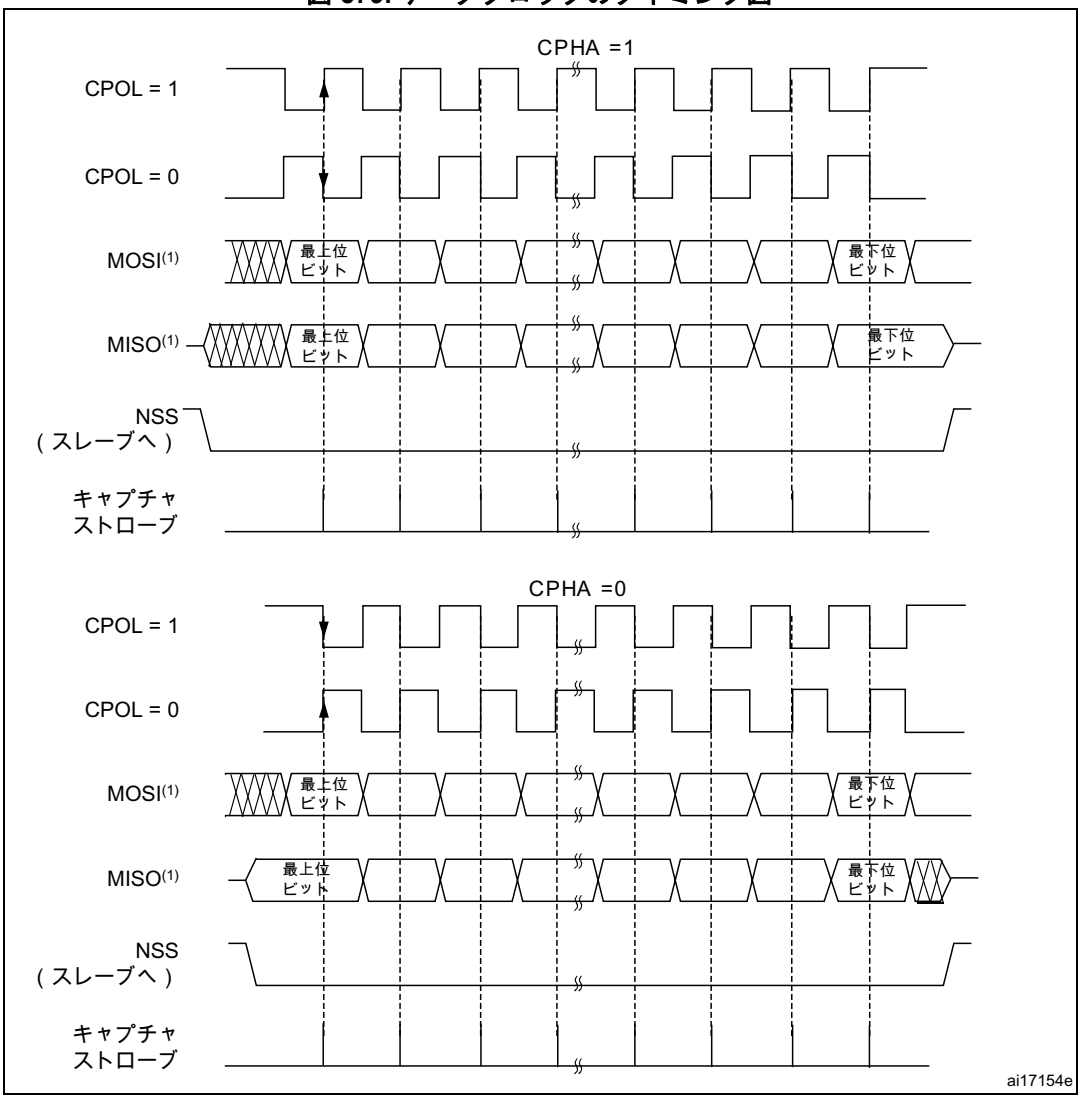
CPOL（クロック極性）ビットと CPHA（クロック位相）ビットの組み合わせによって、データキャプチャのクロックエッジを選択できます。

図 579 は､CPHA ビットと CPOL ビットの 4 つの組み合わせによる SPI 全二重転送を示しています。

注： **CPOL または CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。**

SCK のアイドル状態は、SPIx_CR1 レジスタで（CPOL = 1 なら SCK のプルアップ、CPOL = 0 なら SCK のプルダウンによって）選択された極性に一致する必要があります。

図 579. データクロックのタイミング図

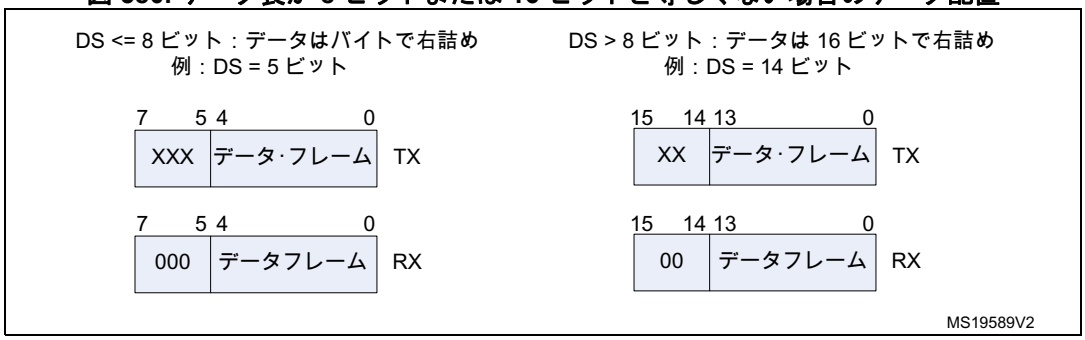


1. データビットの順序は LSBFIRST ビットの設定値に依存します。

データフレームフォーマット

LSBFIRST ビットの値に応じて、SPI シフトレジスタを設定することで、シフトをMSB ファーストまたは LSB ファーストに設定することができます。データフレームサイズは、DS ビットを使用して選択できます。4 ビットから 16 ビット長までの間にセットでき、この設定は送信と受信の両方に適用できます。選択されたデータフレームサイズにかかわらず、FIFO への読出しアクセスは FRXTH レベルに整列されている必要があります。SPIx_DR レジスタにアクセスすると、データフレームはバイト（データがバイトに適合する場合）またはハーフワード（図 580 を参照）のいずれかに常に右詰めされます。通信中、データフレーム内のビットのみがクロック供給され、送信されます。

図 580. データ長が 8 ビットまたは 16 ビットと等しくない場合のデータ配置



注： **最小データ長は 4 ビットです。4 ビット未満のデータ長が選択された場合は、8 ビットのデータフレームサイズに固定されます。**

39.5.7 SPI の設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関するセクションを参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

1. 適切な GPIO レジスタに書込みを行います。MOSIピン、MISOピン、SCKピンの GPIO 設定を行います。
2. SPI_CR1 レジスタに書込みを行います。
   a) BR[2:0] ビットを使用して、シリアルクロックボーレートを設定します（注：4）。
   b) CPOL ビットと CPHA ビットの組み合わせを設定して、データ転送とシリアルクロックの 4 つの関係のうちの 1 つを定義します（CPHA は NSSP モードでクリアする必要があります）。（注：2 - CRC が TI モードで有効になっている場合を除きます）。
   c) RXONLY または BIDIMODE、および BIDIOE を設定することによって（RXONLY と BIDIMODE は同時にセットできません）、単方向または半二重モードを選択します。
   d) LSBFIRST ビットを設定して、フレーム･フォーマットを定義します（注：2）。
   e) CRC が必要な場合は（SCK クロック信号がアイドル状態のとき）、CRCL と CRCEN ビットを設定します。
   f) SSM と SSI を設定します（注：2 および 3）。
   g) MSTR ビットを設定します（マルチマスタ NSS 設定では、MODF エラーを防ぐためにマスタが設定されている場合、NSS での競合を避けること）。
3. 以下のように、SPI_CR2 レジスタに書込みを行います。
   a) 転送のデータ長を選択するために DS[3:0] ビットを選択します。
   b) SSOE を設定します（注：1、2、3）。
   c) TI プロトコルが必要な場合は FRF ビットをセットします（TI モードで NSSP ビットをクリアされたままにする）。
   d) 2 つのデータユニット間で NSS パルスモードが必要な場合は NSSP ビットをセットします（NSSP モードで CHPA と TI ビットをクリアされたままにする）。
   e) FRXTH ビットを設定します。RXFIFO 閾値は、SPIx_DR レジスタの読出しアクセスサイズに揃える必要があります。
   f) DMA がパックモードで使用される場合は、LDMA_TX および LDMA_RX ビットを初期化します。
4. SPI_CRCPR レジスタに書込みを行います。必要に応じて CRC 多項式を設定します。

5. 適切な DMA レジスタに書込みを行います。DMA ストリームが使用されている場合は、DMA レジスタに SPI Tx および Rx 専用の DMA ストリームを設定します。

注： **（1）このステップはスレーブモードでは必要ありません。**

（2）このステップは TI モードでは必要ありません。

（3）このステップは NSSP モードでは必要ありません。

（4）このステップは、スレーブモードにて TI モードで動作している場合を除き、スレーブモードでは必要ありません。

39.5.8 SPI を有効にする手順

マスタがクロックを送信する前に、SPI スレーブを有効にすることを推奨します。さもなければ、望ましくないデータ送信が発生することがあります。スレーブのデータレジスタは、マスタとの通信を開始する前に、送信データをすでに格納していなければなりません（通信クロックの 1 番目のエッジに、またはクロック信号が連続的なときは現在の通信の最後の前に）。SPI スレーブが有効になる前に、SCK 信号を選択された極性に対応するアイドル状態のレベルに安定させる必要があります。

全二重（または送信専用モード）のマスタは、SPI が有効で TXFIFO がエンプティでない場合、または次の TXFIFO への書込み時に通信を開始します。

あらゆるマスタ受信専用モードにおいて（RXONLY = 1 または BIDIMODE = 1、および BIDIOE = 0）、SPI が有効になるとすぐに、マスタは通信を開始し、クロックは動作を開始します。

DMA を処理するには、該当するセクションを参照してください。

39.5.9 データの送受信手順

RXFIFO および TXFIFO

すべての SPI データトランザクションは 32 ビット埋め込み FIFO を通過します。これにより、SPI は連続フローで動作できるようになり、またデータフレームサイズが短い場合にオーバーランを防ぐことができます。各方向にはそれぞれ TXFIFO と RXFIFO と呼ばれる固有の FIFO があります。これらの FIFO は、CRC 計算を有効にした状態で、受信専用モード（スレーブまたはマスタ）を除くすべての SPI モードで使用されます（セクション 39.5.14：CRC 計算を参照）。

FIFO の扱いはデータ変換モード（二重、単方向）、データフレームフォーマット（フレーム内のビット数）、FIFO データレジスタで実行されるアクセスサイズ（8 ビットまたは 16 ビット）、および FIFO アクセス時にデータパッキングが使用されるかどうかに従います（セクション 39.5.13：TI モードを参照）。

SPIx_DR レジスタへの読出しアクセスからは、RXFIFO に保管された、まだ読み出されていない一番古い値が返されます。SPIx_DR への書込みアクセスでは、送信キューの最後に TXFIFO に書き込まれるデータを保管します。読出しアクセスは、必ず SPIx_CR2 レジスタの FRXTH ビットによって設定された RXFIFO 閾値に揃える必要があります。FTLVL[1:0] および FRLVL[1:0] ビットは、両方の FIFO について現在の占有レベルを示します。

SPIx_DR レジスタへの読出しアクセスは RXNE イベントで管理する必要があります。このイベントは、データが RXFIFO に保管され、閾値（FRXTH ビットで定義される）に達した場合にトリガされます。RXNE がクリアされると、RXFIFO はエンプティであるとみなされます。同じように、送信するデータフレームの書込みアクセスは TXE イベントで管理されます。このイベントは、TXFIFO レベルが容量の半分以下である場合にトリガされます。そうでない場合、TXE はクリアされ、TXFIFO がフルであるとみなされます。このように、データフレームフォーマットが 8 ビットを超えない場合、TXFIFO では最大 3 つのデータフレームのみを保管できるのに対し、RXFIFO では最大 4 つを保管することができます。この差異は、ソフトウェアが 16 ビットモードで TXFIFO により多くのデータの書込みを試みた場合に、すでに TXFIFO に保管されている 3x 8 ビットデータフレームが破損する可

能性を防ぎます。TXE イベントと RXNE イベントの両方を割込みによってポーリングまたは処理できます。図 582 から 図 585 までを参照してください。

データ交換を管理するもう一つの方法は、DMA を使用することです（DMA（ダイレクトメモリアクセス）を使用する通信を参照）。

RXFIFO がフルのときに次のデータを受信した場合、オーバーランイベントが発生します（セクション 39.5.10：SPI ステータスフラグの OVR フラグの説明を参照）。オーバーランイベントは割込みによってポーリングまたは処理できます。

セットされる BSY ビットは、現在のデータフレームの進行中のトランザクションを示します。クロック信号が流れ続けているときは、BSY フラグはマスタ側のデータフレーム間でセットされたままになりますが、スレーブの各データフレーム転送間の 1 つの SPI クロックの最小時間においてローレベルになります。

シーケンス処理

いくつかのデータフレームを単一シーケンスに渡してメッセージを完成させることができます。送信が有効な場合、シーケンスはマスタ側の TXFIFO に何らかのデータが存在する場合に開始し、その間続行します。TXFIFO がエンプティになるまで、マスタによってクロック信号が供給され続け、その後追加のデータを待つことを停止します。

半二重（BIDIMODE=1、BIDIOE=0）または単方向（BIDIMODE=0、RXONLY=1）の受信専用モードでは、SPI が有効化され、受信専用モードがアクティブ化されると、直ちにマスタによってシーケンスが開始されます。マスタによってクロック信号が供給されますが、この信号はマスタが SPI または受信専用モードを無効にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。

マスタはあらゆるトランザクションを連続モードで供給できる（SCK 信号は連続的）一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。必要に応じて、マスタは通信速度を下げ、より低速のクロックか、または十分な遅延を含む個別のフレーム／データセッションを供給する必要があります。次の 2 点に注意してください。SPI モードのマスタまたはスレーブに対するアンダーフローエラー信号はありません。また、スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されます。スレーブが DMA を使用することが特に望ましいのは、データフレームが短く、バスが高速の場合です。

マルチスレーブシステムにおいて通信用のスレーブを 1 つだけ選択するには、各シーケンスをNSS パルス内に収める必要があります。単一のスレーブシステムでは NSS によってスレーブを制御する必要はありませんが、スレーブを各データ転送シーケンスの開始と同期させるために、ここにもパルスを供給することをお勧めします。NSS は、ソフトウェアとハードウェアの両方で管理できます（セクション 39.5.5：スレーブ選択（NSS）ピンの管理を参照）。

セットされた BSY ビットは、進行中のデータフレームトランザクションを示します。専用のフレームトランザクションが終了すると、RXNE フラグが立てられます。最後のビットは単にサンプリングされ、すべてのデータフレームが RXFIFO に保管されます。

SPI を無効にする手順

SPI を無効にする場合は、本項に記載されている無効化手順に従ってください。この手順は、ペリフェラルクロックが停止し、システムが低電力モードに入る前に行うことが重要です。この場合、進行中のトランザクションが破壊されることがあります。モードによっては、この無効化手順が連続通信を停止させる唯一の方法です。

全二重または送信専用モードでは、マスタは、転送するデータの供給を停止した時点でいかなるトランザクションも終了することができます。この場合、クロックは最後のデータトランザクション後に停止します。奇数のデータフレームがダミーバイトの変換を防ぐためにトランザクション処理された

場合、パッキングモードでは特に注意する必要があります（データパッキングセクションを参照）。これらのモードで SPI を無効化する前に、ユーザは標準的な無効化手順に従う必要があります。フレームトランザクションの進行中または次のデータフレームを TXFIFO に保管しているときにマスタトランスミッタで SPI を無効化した場合、SPI の動作は保証されません。

マスタがいずれかの受信専用モードに設定されている場合、連続クロックを停止する唯一の方法は SPE=0 にしてペリフェラルを無効にすることです。これは、最初のビットのサンプリング時間の間、および最後のビット転送が開始される前における最後のデータフレームトランザクション内の特定の時間枠内で発生する必要があります（予測されるデータフレームの総数を受信し、最後の有効データフレーム後に追加の「ダミー」データの読出しを防ぐため）。このモードで SPI を無効にするには、特定の手順に従う必要があります。

SPI が無効の場合、受信済みで読み出されていないデータは RXFIFO に保管されたままになり、次に SPI を有効にする際に、新しいシーケンスを開始する前に処理する必要があります。未読のデータを防ぐには、SPI を無効にする際に RXFIFO がエンプティであることを確認してください。これは、正しい無効化手順を使用して、またはペリフェラルリセット専用の固有レジスタを制御してソフトウェアリセットを行い、すべての SPI レジスタを初期化することで確認できます(RCC_APBiRSTR レジスタの SPIiRST ビットを参照)。

標準的な無効化手順は､送信セッションが完全に終わったかどうかをチェックするために､FTLVL[1:0] および BSY フラグの状態をプルして行われます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- NSS 信号がソフトウェアによって管理されており､マスタはスレーブに適切な NSS パルスの終わりを提供する必要がある場合
- 最後のデータフレームまたは CRC フレームのトランザクションがまだペリフェラルバスで進行している間に、DMA または FIFO からのトランザクションのストリームが完了した場合

正しい無効化手順を以下に示します（受信専用モードが使用されている場合を除く）。

1. FTLVL[1:0] = 00（送信するデータがなくなる）まで待ちます。
2. BSY = 0 （最後のデータフレームが処理される）まで待ちます。
3. SPI を無効にします（SPE = 0）。
4. FRLVL[1:0] = 00（受信したデータをすべて読み出す）までデータを読み出します。

受信専用モードの場合の正しい無効化手順を以下に示します。

1. 最後のデータフレームの進行中に特定の時間枠内で SPI を無効（SPE = 0）にすることにより、受信フローへの割込みを行います。
2. BSY = 0 （最後のデータフレームが処理される）まで待ちます。
3. FRLVL[1:0] = 00（受信したデータをすべて読み出す）までデータを読み出します。

注： **パッキングモードが使用され､8 ビット以下（1 バイトに適合）のフォーマットの奇数のデータフレームを受信する必要がある場合は､FRLVL[1:0] = 01 のときに FRXTH をセットしなければなりません。これにより、最後の奇数データフレームを読み出し、良好な FIFO ポインタの整列を維持するための RXNE イベントが生成されます。**

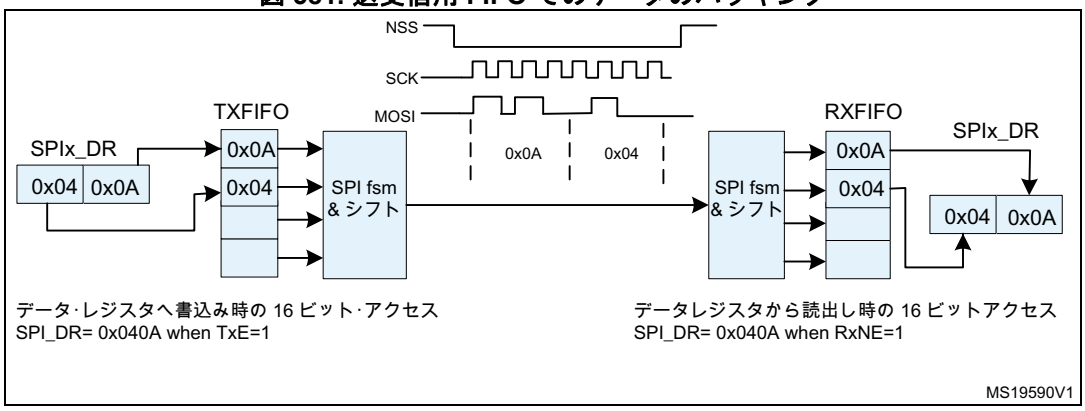
データパッキング

データフレームサイズが 1 バイト（8 ビット以下）に適合する場合、SPIx_DR レジスタで 16 ビットの読出しまたは書込みアクセスが実行されたときに、自動的にデータパッキングが使用されます。この場合、二重データフレームパターンが並行して処理されます。最初に、SPI はアクセスしたワードの LSB に保管されたパターンを使用して動作します。次に、MSB に保管された残り半分を使用して動作します。図 581 に、データパッキングモードのシーケンス処理の例を示します。2 つのデータフレームは、トランスミッタの SPIx_DR レジスタに単一の 16 ビットがアクセスした後で送信されます。このシーケンスでは、RXFIFO 閾値が 16 ビット（FRXTH=0）にセットされている場合に、レシーバで 1 つの RXNE イベントのみを生成できます。この場合、この単一の RXNE イベントへのレ

スポンスとして、レシーバは SPIx_DR の単一の 16 ビット読出しにより、両方のデータフレームにアクセスする必要があります。RxFIFO 閾値設定および以下の読出しアクセスは、常にレシーバ側に整列させる必要があります。整列されていない場合は、データを失う可能性があります。

これらの奇数の「1 バイトに適合」するデータフレームを処理する必要がある場合、特定の問題が発生します。トランスミッタ側では、SPIx_DR への 8 ビットアクセスを持つ任意の奇数シーケンスの最後のデータフレームを書き込むだけで十分です。レシーバでは、RXNE イベントを生成するために、フレームの奇数シーケンスで受信した最後のデータフレームの Rx_FIFO 閾値レベルを変更する必要があります。

図 581. 送受信用 FIFO でのデータのパッキング



1. 例：データサイズDS[3:0]は4bitで設定され、CPOL=0、CPHA=1および LSBFIRST =0。有効ビットがバスでのみ実行される場合、LSB バイトの内容が最初に処理される場合、トランスミッタ側で未使用のビットが考慮される場合、およびレシーバ側でゼロパディングされる場合、データ保存は常に右詰めです。

DMA（ダイレクトメモリアクセス）を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読出し／書込み処理を容易にするために、SPI は簡単なリクエスト／確認応答プロトコルを実現する DMA 機能を備えています。

SPIx_CR2 レジスタの TXDMAEN または RXDMAEN 有効ビットをセットすると、DMA アクセスがリクエストされます。Tx バッファと Rx バッファには、別々のリクエストを発行する必要があります。

- 送信では、TXE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタに書き込みます。
- 受信では、RXNE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタを読み出します。

図 582 から 図 585 までを参照してください。

SPI がデータの送信にのみ使用される場合、SPI Tx DMA チャネルのみを有効にすることができます。この場合、受信したデータは読み出されないため、OVR フラグがセットされます。SPI がデータの受信にのみ使用される場合、SPI Rx DMA チャネルのみを有効にすることができます。

送信モードで、DMA がすべての送信データを書き込んだとき（DMA_ISR レジスタの TCIF フラグがセットされます）、BSY フラグを監視することで SPI 通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPI を無効にする前、または STOP モードに入る前にこの操作を行う必要があります。ソフトウェアは、まず FTLVL[1:0] = 00 になるまで待ってから、BSY=0 になるまで待つ必要があります。

DMA を使用して通信を開始する場合、DMA チャネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rx が使用される場合、SPI_CR2 レジスタの RXDMAEN ビットの DMA Rx バッファを有効にします。
2. ストリームが使用される場合、DMA レジスタにて TxおよびRxの DMA ストリームを有効にします。
3. DMA Tx が使用される場合、SPI_CR2 レジスタの TXDMAEN ビットにてDMA Tx バッファを有効にします。
4. SPE ビットをセットして、SPI を有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. ストリームが使用される場合、DMA レジスタにて TxおよびRxの DMA ストリームを無効にします。
2. SPI 無効化手順に従って SPI を無効にします。
3. DMA Tx および（または）DMA Rx が使用されている場合、SPI_CR2 レジスタの TXDMAEN ビットおよび RXDMAEN ビットをクリアすることにより、DMA Tx バッファおよび Rx バッファを無効にします。

DMA によるパッキング

DMA（SPIx_CR2 レジスタでセットされた TXDMAEN および RXDMAEN）で転送が管理される場合、パッキングモードは SPI TX および SPI RX DMA チャネルで設定される PSIZE 値によって、自動的に有効／無効になります。DMA チャネルの PSIZE 値が 16 ビットと等しい場合、SPI データサイズは 8 ビット以下で、パッキングモードは有効になります。その後、DMA は SPIx_DR レジスタへの書込み動作を管理します。

データパッキングモードを使用する場合で転送データ数が 2 の倍数でない場合は、LDMA_TX/LDMA_RX ビットをセットする必要があります。SPI では、最後の DMA 転送の送受信に 1 つのデータのみを考慮します（詳細については、1715 ページのデータパッキングを参照）。

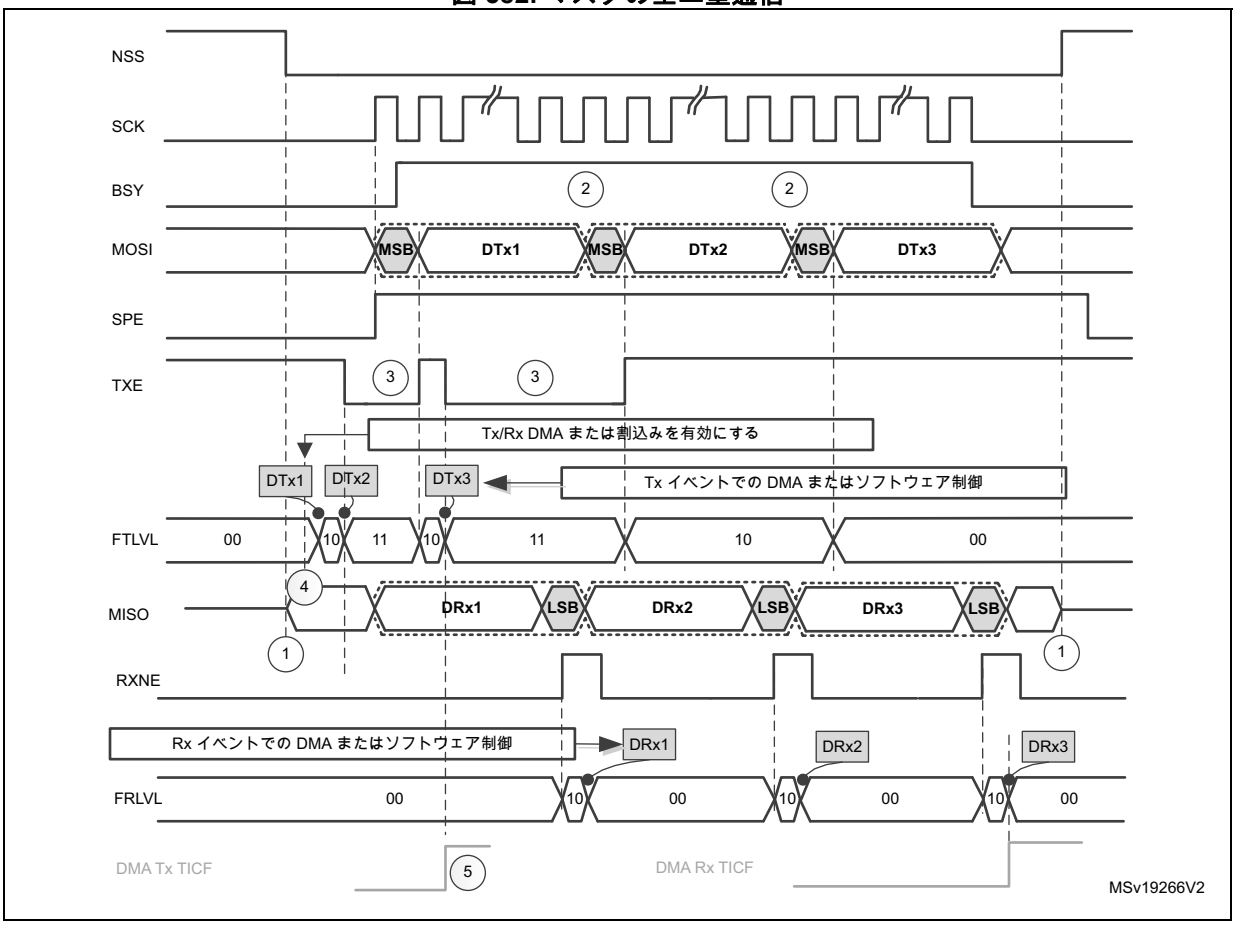
通信図

このセクションでは、一部の標準的なタイミング構成について説明します。これらの構成は、SPI イベントの処理方法（ポーリング、割込み、または DMA）にかかわらず有効です。単純にするには、ここでは LSBFIRST=0、CPOL=0、および CPHA=1 設定を共通の前提として使用します。DMA ストリームの完全な設定は提供されません。

以下の番号付された注は、1719 ページの図 582 から 1722 ページの図 585 に共通です。

1. スレーブは、NSS がアクティブで SPI が有効になると MISO ラインを制御し始め、そのいずれかがリリースされるとラインから切断されます。事前にマスタ専用のデータを準備するには、トランザクションを開始する前にスレーブに十分な時間を与える必要があります。
   マスタでは、SPI が有効な場合のみ、SPI ペリフェラルが MOSI および SCK 信号で（また、ときに NSS 信号でも）制御できるようになります。SPI が無効の場合、SPI ペリフェラルは GPIO ロジックから切断され、これによりこれらのラインのレベルは GPIO 設定にのみ依存します。
2. マスタでは、通信（クロック信号）が連続的な場合に、BSY はフレーム間でアクティブなままとなります。スレーブでは、BSY 信号は必ずフレーム間で最低でも 1 クロックサイクルは無効になります。
3. TXE 信号は TXFIFO がフルの場合にのみクリアされます。
4. DMA アービトレーションプロセスは TXDMAEN ビットのセット後すぐに開始します。TXE 割込みは TXEIE のセット後すぐに生成されます。TXE 信号がアクティブレベルにあると、TxFIFO へのデータ転送が開始され、TxFIFO がフルまたは DMA 転送が完了するまで続きます。
5. 送信するデータをすべて TxFIFO に適合することができる場合、SPI バスでの通信が開始される前に DMA Tx TCIF フラグを立てることができます。このフラグは、常に SPI トランザクションが完了する前に立てられます。
6. パッケージの CRC 値は、SPIx_TXCRCR および SPIx_RXCRCR レジスタで、フレームごとに連続的に計算されます。CRC 情報はすべてのデータパッケージが完了した後で処理されます。これは、自動的に DMA で（必ず Tx チャネルを処理するフレーム数にセットすること）、またはソフトウェアで（必ず最後のデータフレーム処理中に CRCNEXT を処理すること）行われます。SPIx_TXCRCR で計算された CRC 値がトランスミッタによって簡単に送信されるのに対し、受信した CRC 情報は RxFIFO に読み込まれ、SPIx_RXCRCR レジスタの内容と比較されます（差異がある場合は、ここで CRC エラーフラグが立てられます）。このため、ユーザは注意してこの情報を FIFO から一掃する必要があります。これには、RxFIFO のすべての保管された内容を読み出すソフトウェアか、Rx チャネルの適切なデータフレーム数がプリセットされている場合は DMA を使用します（データフレーム数 + CRC フレーム数）（前提条件の例の設定を参照）。
7. データがパックされたモードで、TxE および RxNE イベントはペアにされ、データフレーム数が偶数になるまで FIFO への各読出し／書込みアクセスは 16 ビット幅となります。TxFIFO が ¾ フルの場合、FTLVL ステータスは FIFO フルレベルのままです。このため、最後の奇数のデータフレームを TxFIFO が ½ フルになる前に格納することはできません。このフレームは 8 ビットアクセスを持つ TxFIFO に格納されます。これは、ソフトウェアで、または LDMA_TX 制御がセットされている場合は自動的に DMA で行われます。
8. パックされたモードで最後の奇数のデータフレームを受信するには、最後のデータフレームが処理された時に Rx 閾値を 8 ビットに変更する必要があります。これは、ソフトウェアで FRXTH=1 に設定するか、LDMA_RX がセットされている場合は自動的に DMA 内部信号で行われます。

図 582. マスタの全二重通信



マスタの全二重通信の前提条件の例：

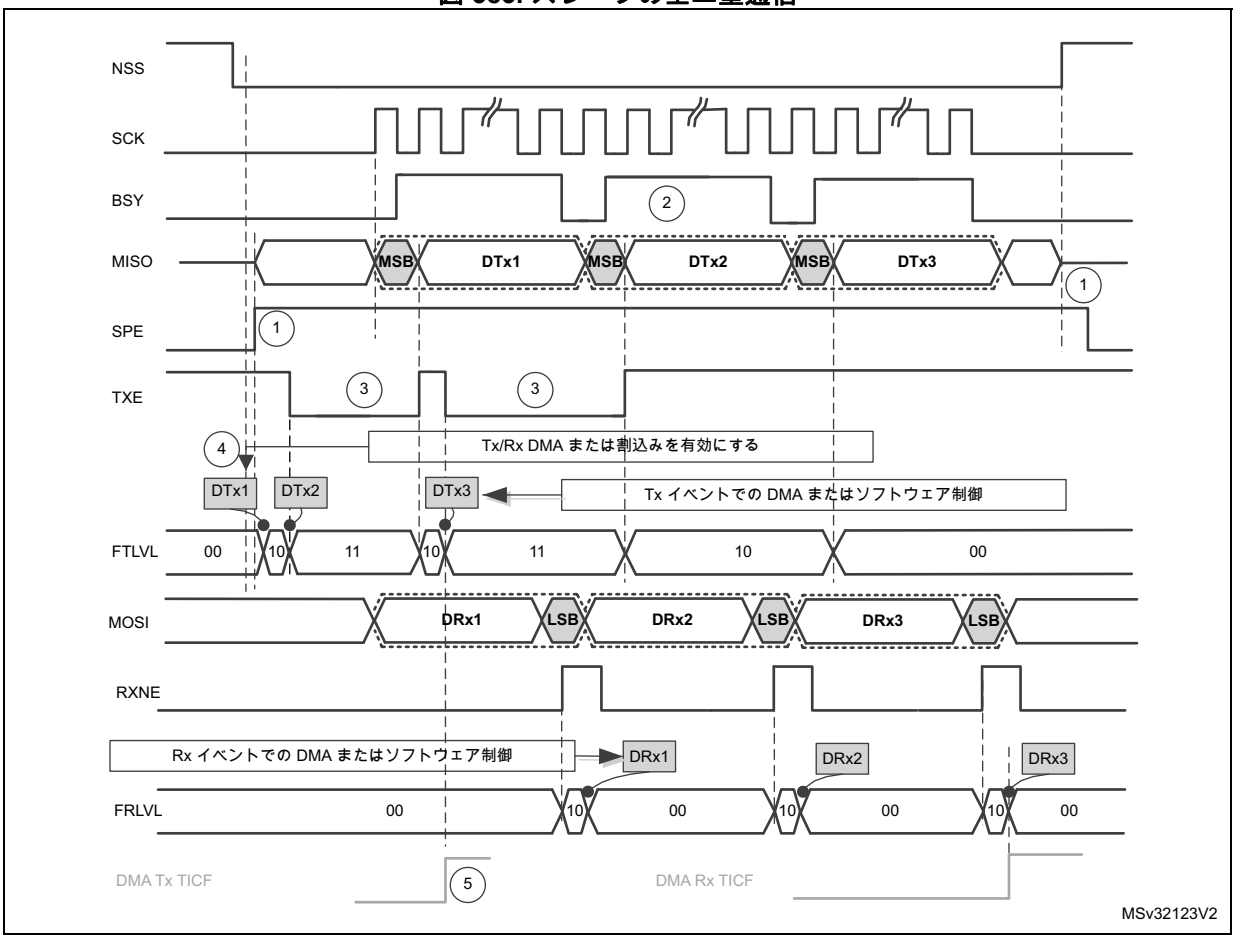
- データサイズ > 8 ビット

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、1718 ページの通信図を参照してください。

図 583. スレーブの全二重通信



スレーブの全二重通信の前提条件の例：

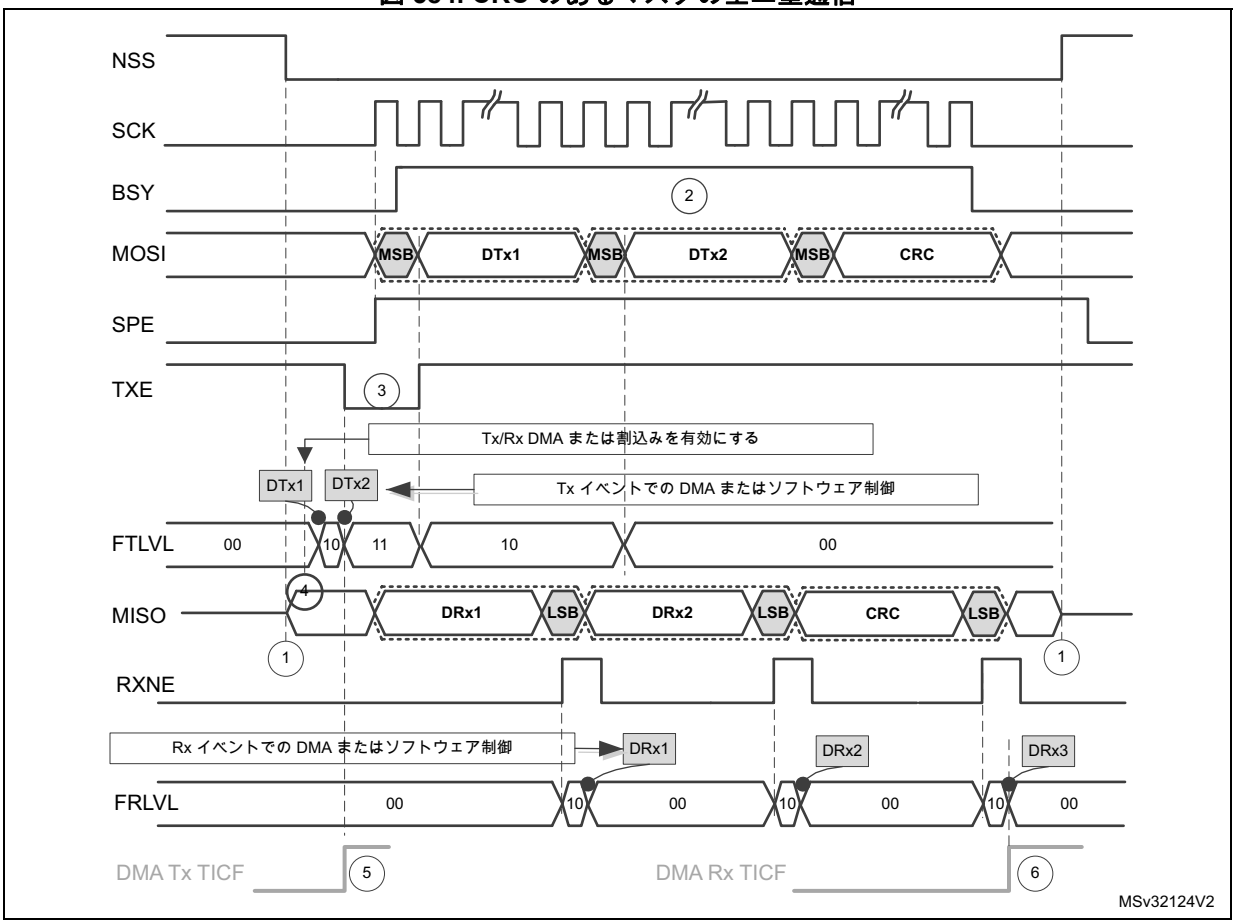
- データサイズ > 8 ビット

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、1718 ページの通信図を参照してください。

図 584. CRC のあるマスタの全二重通信



CRC のあるマスタの全二重通信の前提条件の例：

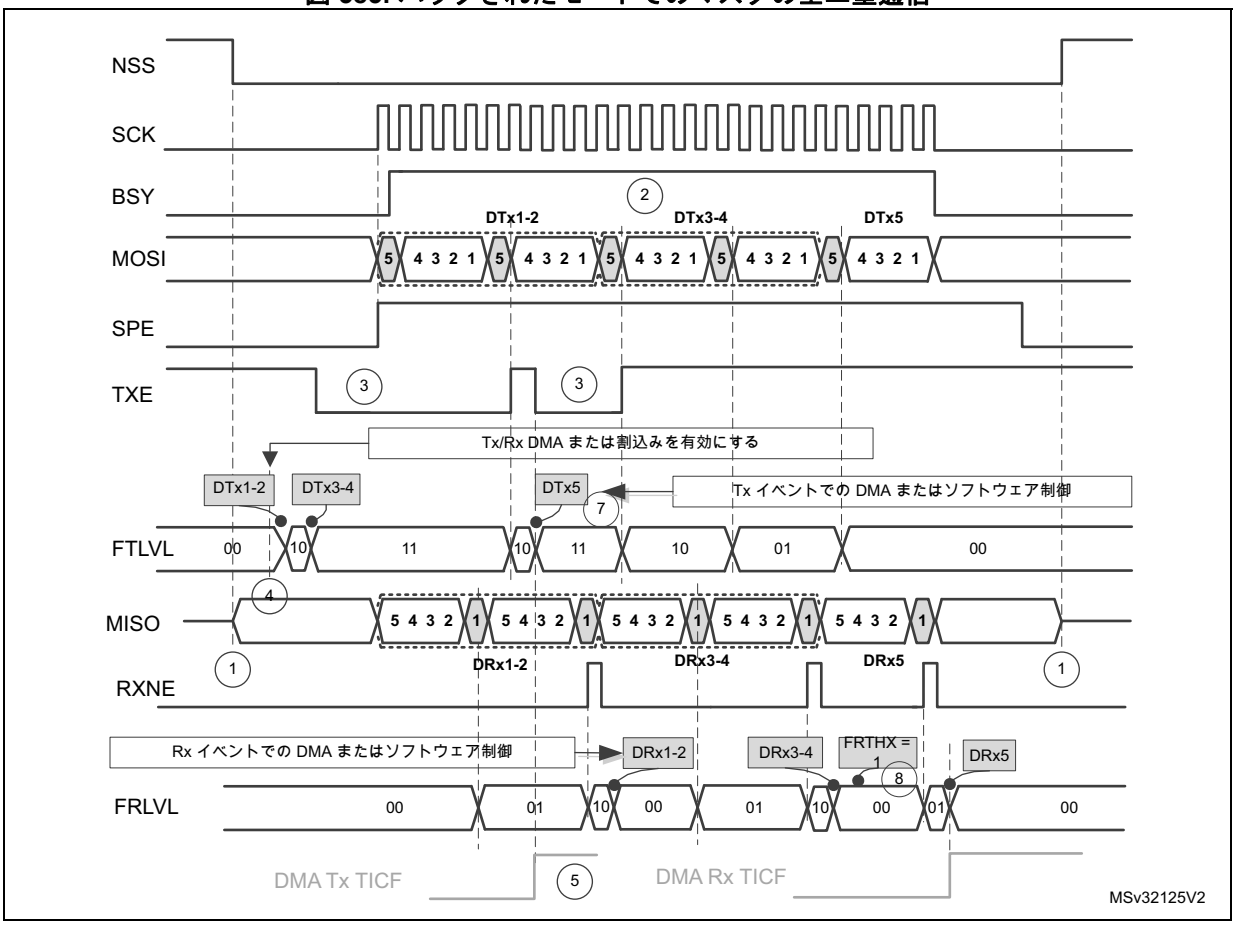
- データサイズ = 16 ビット
- CRC は有効です。

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 2 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット

共通の前提条件および注の詳細については、1718 ページの通信図を参照してください。

図 585. パックされたモードでのマスタの全二重通信



パックされたモードでのマスタの全二重通信の前提条件の例：

- データサイズ = 5 ビット
- 読出し／書込み FIFO は、ほとんど 16 ビットアクセスで実行されます。
- FRXTH = 0

DMA を使用する場合：

- DMA でトランザクションされる Tx フレーム数を 3 にセット
- DMA でトランザクションされる Rx フレーム数を 3 にセット
- Tx と Rx DMA チャネルの両方の PSIZE を 16 ビットにセット
- LDMA_TX=1 および LDMA_RX=1

共通の前提条件および注の詳細については、1718 ページの通信図を参照してください。

39.5.10 SPI ステータスフラグ

アプリケーションが SPI バスの状態を完全に監視できるように、3 つのステータスフラグが用意されています。

Tx バッファエンプティフラグ（TXE）

TXE フラグは､送信 TXFIFO に送信データを格納するための十分なスペースがある場合にセットされます｡TXE フラグは TXFIFO レベルにリンクされます｡フラグはハイになり､TXFIFO レベルが FIFO の深さの 1/2 以下になってもハイを維持します｡SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は、割込みを生成できます。このビットは、TXFIFO レベルが 1/2 を超えると自動的にクリアされます。

Rx バッファノットエンプティ（RXNE）

SPIx_CR2 レジスタの FRXTH ビット値によって、RXNE フラグがセットされます。

- FRXTH をセットすると RXNE はハイになり、RXFIFO レベルが 1/4（8 ビット）以上になってもハイを維持します。
- FRXTH をクリアすると RXNE はハイになり、RXFIFO レベルが 1/2（16 ビット）以上になってもハイを維持します。

SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は、割込みを生成できます。

RXNE は、上記の条件が真ではなくなった場合にハードウェアによって自動的にクリアされます。

ビジーフラグ（BSY）

BSY フラグは、ハードウェアによってセット／クリアされます（このフラグへの書込みは無効）。

BSY フラグがセットされると､SPI上 でデータ転送が進行中であることを示します（SPI バスはビジー）。

BSY は、転送終了を検出するために特定のモードで使用できます。これにより、低電力モード（ペリフェラルにクロックを供給しない）に入る前に、ソフトウェアで SPI やそのペリフェラルクロックを無効にすることができます。これによって、最後の転送データの破壊を回避します。

BSY フラグは、マルチマスタシステムでの書込み衝突の回避にも役立ちます。

BSY フラグは次のいずれかの条件下でクリアされます。

- SPI が正常に無効にされたとき
- マスタモードで、障害が検出（MODF ビットが 1 にセットされます）されたとき
- マスタモードで、データ送信が終了し、送信準備ができている新しいデータがないとき
- スレーブモードで、各データ転送間で少なくとも SPI の 1 クロックサイクルの間、BSY フラグが 0 にセットされているとき

注： **次の送信がすぐにマスタで処理される場合（マスタが受信専用モードに設定されているか、その送信 FIFO がエンプティでない場合）、通信は連続的で、BSY フラグはマスタ側での送信の間、“1”にセットされたままになります。スレーブとは異なりますが、必ず（BSY フラグを使用する代わりに）TXE フラグと RXNE フラグを使用して、データの送受信の処理を行うことを推奨します。**

39.5.11 SPI エラーフラグ

次のいずれかのエラーフラグがセットされていて、ERRIE ビットをセットすることにより割込みが有効になっている場合、SPI 割込みが生成されます。

オーバーランフラグ（OVR）

マスタまたはスレーブでデータを受信し、RXFIFO に受信データを格納するための十分なスペースがない場合に、オーバーラン状態が発生します。これは、ソフトウェアまたは DMA が RXFIFO に格納された前の受信データを読み出すために必要な時間が十分になかったか、データストレージのための空間が制限されている場合に発生する可能性があります。たとえば、RXFIFO は CRC が受信専用モードでのみ有効な場合は使用できないため、この場合、受信バッファは単一データフレームバッファに制限されます（セクション 39.5.14：CRC 計算を参照）。

オーバーラン条件が発生すると、RXFIFO にある前の値を新しく受信した値で上書きすることはできません。新しく受信した値は破棄され、それ以降に送信されたすべてのデータは失われます。OVR ビットをクリアするには、SPI_DR レジスタを読み出し、続けて SPI_SR レジスタに読出しアクセスを行います。

モードフォールト（MODF）

モードフォールトは、マスタデバイスが内部 NSS 信号（NSS ハードウェアモードでは NSS ピン、NSS ソフトウェアモードでは SSI ビット）をローレベルにプルダウンしたときに発生します。これにより、MODF ビットが自動的にセットされます。マスタモードフォールトは、SPI インタフェースに次のような影響を与えます。

- MODF ビットがセットされ、ERRIE ビットがセットされている場合は SPI 割込みが生成されます。
- SPE ビットがクリアされます。これによって、デバイスからのすべての出力がブロックされ、SPI インタフェースが無効になります。
- MSTR ビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODF ビットをクリアするには、次のソフトウェアシーケンスを実行します。

1. MODF ビットがセットされている間、SPIx_SR レジスタに読出し／書込みアクセスを行います。
2. 次に、SPIx_CR1 レジスタに書込みを行います。

複数の MCU で構成されるシステムでスレーブ間の競合を避けるには、MODF ビットをクリアするシーケンス中、NSS ピンをハイレベルにプルアップする必要があります。このクリアシーケンスの後、SPE ビットと MSTR ビットは、元の状態に戻すことができます。安全のため、MODF ビットがセットされている間、ハードウェアは SPE ビットと MSTR ビットのセットを許可しません。スレーブデバイスでは、MODF ビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

CRC エラー（CRCERR）

このフラグを使用して、SPIx_CR1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。シフトレジスタに受信された値が、レシーバである SPIx_RXCRCR の値と一致しなかった場合、SPIx_SR レジスタの CRCERR フラグがセットされます。フラグはソフトウェアによってクリアされます。

TI モードフレームフォーマットエラー（FRE）

SPI がスレーブモードで動作し、かつ TI モードプロトコルに準拠した設定となっている場合、通信の実行中に NSS パルスが発生すると、TI モードフレームフォーマットエラーが検出されます。このエラーが発生すると、SPIx_SR レジスタの FRE フラグがセットされます。エラー発生時には SPI は無効にされず、この NSS パルスは無視されます。SPI は次の NSS パルスを待ってから新規の転送を開

始します。このエラーの検出により 2 バイトのデータが失われるため、データは破壊される可能性があります。

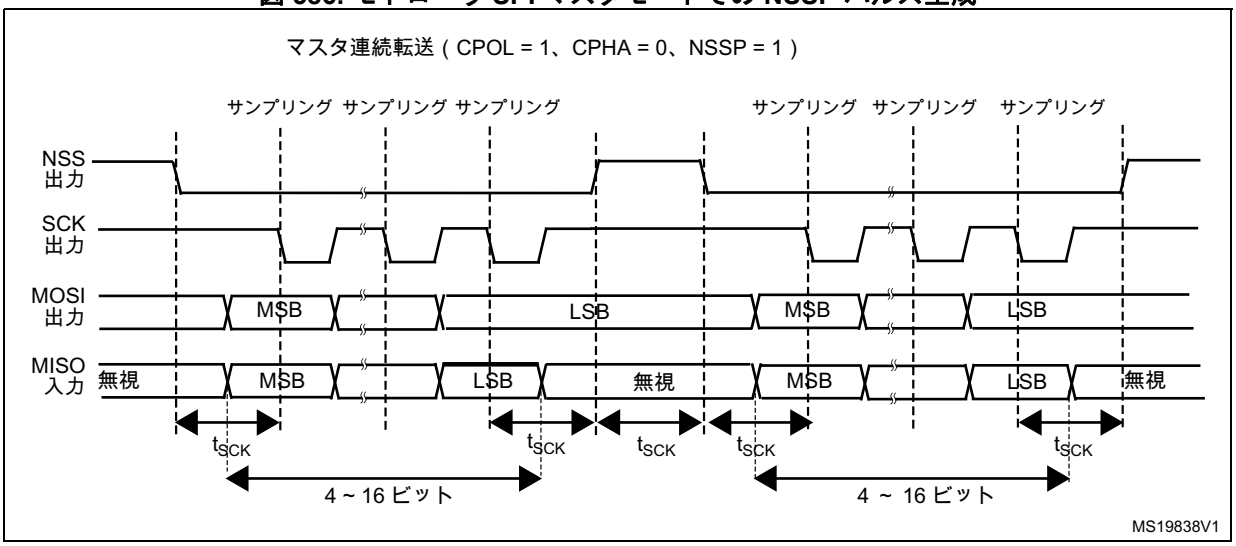
SPIx_SR レジスタを読み出すと、FRE フラグがクリアされます。ERRIE ビットがセットされている場合、NSS エラー検出時に割込みが生成されます。この場合、データの一貫性が保証されなくなるため、SPI を無効にする必要があり、またスレーブ SPI が再び有効化された場合は、マスタによって通信を再起動する必要があります。

39.5.12 NSS パルスモード

このモードは SPIx_CR2 レジスタの NSSP ビットで有効化され､SPI インタフェースが 1 番目のエッジにキャプチャのある モトローラ SPI マスタ（FRF=0）として設定されている場合にのみ有効になります（SPIx_CR1 CPHA = 0、CPOL 設定は無視されます）。有効にすると、最低 1 つのクロック周期の間、NSS がハイレベルを維持する場合に、2 つの連続したデータフレーム転送の間で NSS パルスが生成されます。このモードでは、スレーブでデータをラッチできます。NSSP パルスモードは、シングルマスタ - スレーブペアのアプリケーション用に設計されています。

図 586 に、NSSP パルスモードが有効な場合の NSS ピンの管理の図を示します。

図 586. モトローラ SPI マスタモードでの NSSP パルス生成



注： **CPOL = 0 の場合に同様の動作が発生します。この場合、サンプリングエッジは SCK の立ち上がりエッジで、NSS のアサートおよびネゲートはこのサンプリングエッジを参照します。**

39.5.13 TI モード

マスタモードでの TI プロトコル

SPI インタフェースは TI プロトコルと互換性があります。SPIx_CR2 レジスタの FRF ビットを使って、SPI をこのプロトコルに準拠させるように設定することができます。

SPIx_CR1 レジスタにセットされる値によらず、クロックの極性と位相は TI プロトコル要件に必ず適合します。NSS 管理も TI プロトコルに固有なものになります。これにより、この場合の SPIx_CR1 レジスタと SPIx_CR2 レジスタによる NSS 管理の設定（SSM、SSI、SSOE）ができなくなります。

スレーブモードでは、SPI ボーレートプリスケーラを使用して、現在のトランザクションが終了した時点で、MISO ピンの状態がハイインピーダンスに変化するタイミングを制御します（図 587 を参照）。任意のボーレートが使用できるため、このタイミングを非常に柔軟に決定することができます。

ただし、ボーレートは外部マスタクロックボーレートに設定されるのが一般的です。MISO 信号がハイインピーダンス（$t_{release}$）になるまでの遅延は、内部再同期と SPIx_CR1 レジスタの BR[2:0] ビットで設定されたボーレート値によって変わります。この値は次式で求められます：

$$\frac{t_{baud_rate}}{2} + 4 \times t_{pclk} < t_{release} < \frac{t_{baud_rate}}{2} + 6 \times t_{pclk}$$

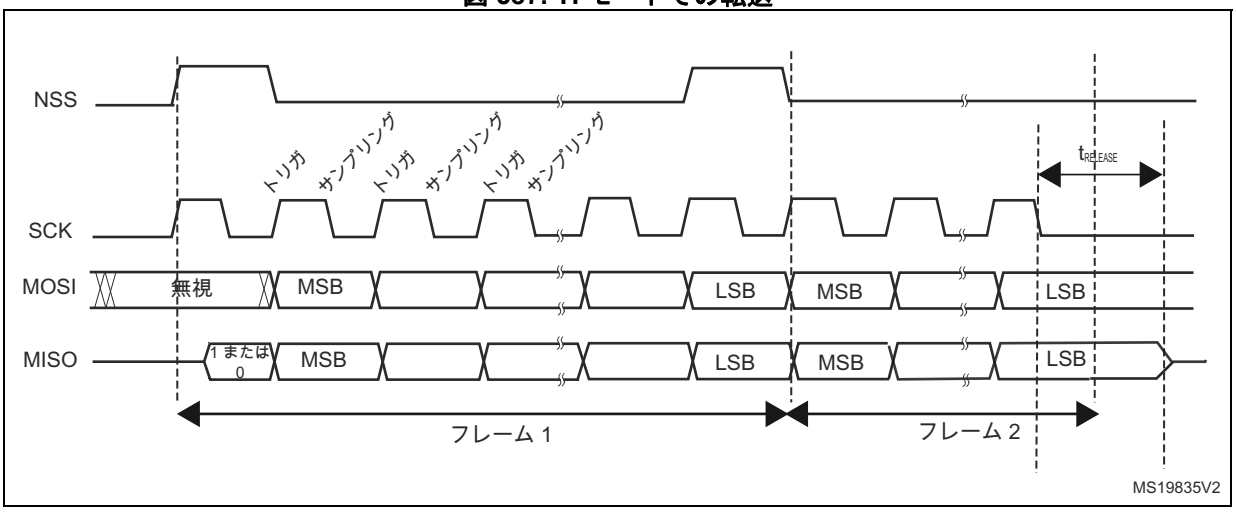
スレーブがデータフレームトランザクション中に NSS パルスの誤配置を検出すると、TIFRE フラグがセットされます。

データサイズが 4 ビットまたは 5 ビットである場合、全二重モードまたは送信専用モードのマスタは、最下位ビット後に追加された 1 つまたは複数のダミーデータビットを持つプロトコルを使用します。TI NSS パルスは、各周期の最下位ビットではなく、このダミービットクロックサイクル上に生成されます。

この機能はモトローラの SPI 通信には使用できません（FRF ビットを 0 に設定）。

図 587：TI モードでの転送 TI モードが選択されているときの SPI の通信波形を示します。

図 587. TI モードでの転送



39.5.14 CRC 計算

2 つの CRC 計算機が、送信データおよび受信データの信頼性をチェックするために実装されています。SPI は、フレームデータ長とは別に、8 ビットまたは 16 ビットに固定される CRC8 または CRC16 を計算します。その他のデータフレーム長では、CRC は使用できません。

CRC の原理

CRC 計算は、SPI が有効（SPE = 1）になる前に、SPIx_CR1 レジスタの CRCEN ビットをセットすることによって有効になります。CRC 値は、各ビットに対して奇数のプログラム可能な多項式の値を使用して計算されます。計算は、SPIx_CR1 レジスタの CPHA ビットと CPOL ビットによって定義されるサンプリングクロックエッジで行われます。計算された CRC 値は、データブロックの最後のみならず、CPU またはDMA によって管理される転送に関しても、自動的にチェックされます。受信データをもとに内部で計算された CRC 値とトランスミッタが送信した CRC 値の間に不一致が検出された場合、データ破壊エラーを示すためにCRCERR フラグがセットされます。CRC 計算を処理する正しい手順は、SPI の設定および選択された転送管理によって変わります。

注： **多項式の値は必ず奇数でなければなりません。偶数の値はサポートされていません。**

CPU によって管理される CRC 転送

通信が開始され、最後のデータフレームが SPIx_DR レジスタで送信または受信されるまで正常に続きます。次に、CRC フレーム･トランザクションが現在処理中のデータ･フレーム･トランザクションの後に行われることを示すために、SPIx_CR1 レジスタの CRCNEXT ビットをセットする必要があります。CRCNEXT ビットは、最後のデータ･フレーム･トランザクションの終了前にセットしてください。CRC のトランザクション中は CRC 計算は停止されます。

受信した CRC は、データバイトまたはワードのように RXFIFO に格納されます。このため、CRC モードでは、受信バッファを一度に 1 データフレームずつ受信するために使用する単一の 16 ビットバッファとみなす必要があります。

CRC フォーマットトランザクションは、通常データシーケンスの最後に通信を行うためにデータフレームをもう 1 つ受け取ります。ただし、16 ビット CRC によってチェックされた 8 ビットデータフレームを設定する場合、CRC 全体を送信するにはフレームがあと 2 つ必要です。

最後の CRC データを受信すると、受信した値と SPIx_RXCRC レジスタの値を比較する自動チェックが行われます。ソフトウェアは、SPIx_SR レジスタの CRCERR フラグをチェックして、データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCERR フラグに”0”を書き込んでクリアします。

CRC 受信後、CRC 値は RXFIFO に格納され、RXNE フラグをクリアするために SPIx_DR レジスタを読み出す必要があります。

DMA によって管理される CRC 転送

SPI 通信が CRC 通信と DMA モードで有効化される場合、CRC の送受信は通信の最後に自動で行われます（ただし、受信専用モードで CRC データを読みだす場合を除く）。CRCNEXT ビットはソフトウェアで処理する必要はありません。SPI 送信用DMA チャネルのカウンタは、転送するデータフレーム数からCRC フレームを除いてセットする必要があります。レシーバ側では、受信した CRC 値はトランザクションの終了時に DMA によって自動的に処理されますが、RXFIFO から受信した CRC 情報は常にレシーバにロードされますので、ユーザは注意してそれを一掃してください。全二重モードでは、受信 DMA チャネルのカウンタを、受信するデータフレームの数（CRC を含む）にセットすることができます。つまり、たとえば 16 ビット CRC でチェックされた 8 ビットデータフレームの特定の例を、次の式で表すことができます。

DMA_RX = Numb_of_data + 2

受信専用モードでは、DMA 受信チャネルカウンタには転送されたデータ量のみが含まれます（CRC 計算を除く）。次に、DMA からの全体の転送に基づき、すべての CRC 値はこのモードでは単一のバッファとして動作するため、ソフトウェアによって FIFO から読み戻す必要があります。

転送中に内容の破壊が生じた場合、データと CRC の転送の最後に SPIx_SR レジスタの CRCERR フラグがセットされます。

パッキングモードが使用され、データ数が奇数の場合は、LDMA_RX ビットを管理する必要があります。

SPIx_TXCRC および SPIx_RXCRC の値のリセット

CRC フェーズのあとで新しいデータがサンプリングされた場合、SPIx_TXCRC 値および SPIx_RXCRC 値は自動的にクリアされます。これにより、データを中断することなく転送するために、DMA サーキュラモードが使用可能になります（受信専用モードでは使用不可）。いくつかのデータブロックは中間の CRC チェックフェーズによってカバーされます。

通信中に SPI が無効化された場合は、次のシーケンスに従う必要があります。

1. SPI を無効にします。
2. CRCEN ビットをクリアします。
3. CRCEN ビットを有効にします。

4. SPI を有効にします。

注： **SPI インタフェースが スレーブとして設定されている場合、CRCNEXT 信号がリリースされたら CRC フェーズのトランザクション中は NSS 内部信号をローレベルに維持する必要があります。そのため、NSS ハードウェアモードをスレーブで正常に適用する必要がある場合､CRC 計算を NSS パルスモードで使用できません。**

TI モードでは、クロックフェーズおよびクロック極性の設定が固定で、SPIx_CR1 レジスタに依存していなくても、CRC が適用される場合、対応する設定 CPOL=0 CPHA=1 は SPIx_CR1 レジスタで保持される必要があります。さらに、マスタおよびスレーブの両方で上記の CRCEN ビットを再有効化しつつ、CRC 計算は SPI 無効化シーケンスでセッション間でリセットされる必要があります。そうしない場合、CRC 計算がこの特定のモードで破損する可能性があります。

39.6 SPI 割込み

SPI 通信中、割込みは次のイベントによって生成できます。

- ロードする準備ができている TXFIFO の送信
- 受信 RXFIFO で受信したデータ
- マスタモードフォールト
- オーバーランエラー
- TI フレームフォーマットエラー
- CRC プロトコルエラー

割込みは個別に有効または無効にできます。

表 361. SPI 割込みリクエスト

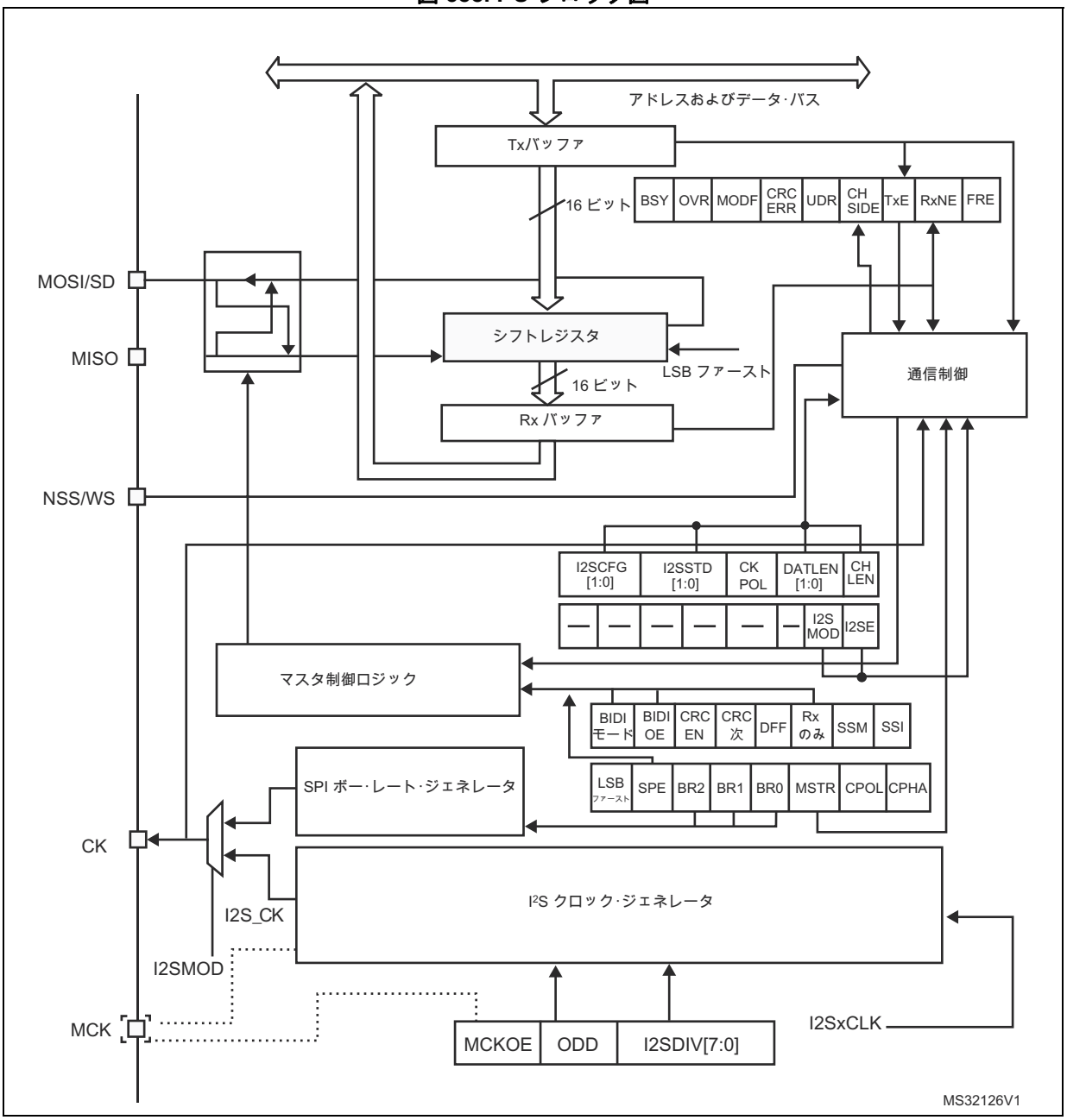
割込みイベント	イベントフラグ	イネーブル制御ビット
ロードする準備ができている TXFIFO の送信	TXE	TXEIE
RXFIFO で受信したデータ	RXNE	RXNEIE
マスタ･モード･フォールト･イベント	MODF	ERRIE
オーバーラン･エラー	OVR	
TI フレーム･フォーマット･エラー	FRE	
CRC プロトコル･エラー	CRCERR	

39.7 I²S の機能詳細

39.7.1 I²S の概要

I²S インタフェースのブロック図を図 588 に示します。

図 588. I²S ブロック図



1. MCK はMISO ピンに配置されます。

SPIx_I2SCFGR レジスタの I2SMOD ビットをセットして I²S 機能を有効にすると、SPI はオーディオI²S インタフェースとして機能することができます。このインタフェースは、主に SPI と同じピン、フラグ、および割込みを使用します。

I2S と SPI は、3 つのピンを共用します。

- SD：シリアル･データは（MOSI ピンに配置）、2 つの時間多重化データ･チャネルで送受信されます（半二重モードのみ）。
- WS：ワード･セレクトは（NSS ピンに配置）、マスタ･モードではデータ制御信号の出力、スレーブ･モードでは入力です。
- CK：シリアル･クロックは(SCK ピンに配置)、マスタ･モードではシリアル･クロック出力、スレーブ･モードではシリアル･クロック入力です。

外部オーディオデバイスにマスタクロック出力が必要な場合、追加のピンを使用できます。

- MCK: マスタ･クロックは（別々に配置）、I2S がマスタ･モードに設定されている（かつ、SPIx_I2SPR レジスタの MCKOE ビットがセットされている）とき、256 x f_sと等しい（f_s はオーディオ･サンプリング周波数）設定済みの周波数で生成されたこの追加クロックを出力するために使用されます。

I2S は、マスタモードに設定されているとき、専用のクロックジェネレータを使用して通信クロックを発生させます。このクロックジェネレータは、マスタクロック出力のソースでもあります。I2S モードでは、2 つの追加レジスタを使用できます。1 つはクロックジェネレータ設定 SPIx_I2SPR にリンクされ、もう 1 つは汎用 I2S 設定レジスタ SPIx_I2SCFGR（オーディオ標準、スレーブ／マスタモード、データフォーマット、パケットフレーム、クロック極性など）です。

SPIx_CR1 レジスタとすべての CRC レジスタは、I2S モードでは使用されません。同様に、SPIx_CR2 レジスタの SSOE ビットと SPIx_SRレジスタ の MODF および CRCERR ビットも使用されません。

I2S は、16 ビット幅モードでのデータ転送に同じ SPI レジスタ（SPIx_DR）を使用します。

39.7.2 サポートされるオーディオプロトコル

3 線バスでは、一般に 2 つのチャネル（右チャネルと左チャネル）で時間多重化されたオーディオデータのみを処理する必要があります。しかしながら、送受信用には 1 つの 16 ビットレジスタしかありません。したがって、各チャネルサイドに対応する適切な値をデータレジスタに書き込んだり、データレジスタからデータを読み出して SPIx_SR レジスタの CHSIDE ビットをチェックして対応するチャネルを識別したりすることは、ソフトウェアの責任です。左チャネルは常に最初に送信され、その後で右チャネルが送信されます（CHSIDE は PCM プロトコルには無関係です）。

4 つのデータおよびパケットフレームを使用できます。データは次のフォーマットで送信されます。

- 16 ビットフレームにパックされた 16 ビットデータ
- 16 ビットフレームにパックされた 32 ビットデータ
- 24 ビットフレームにパックされた 32 ビットデータ
- 32 ビット･フレームにパックされた 32 ビット･データ

32 ビットパケットに拡張された 16 ビットデータを使用するとき、最初の 16 ビット（MSB）は上位ビットであり、LSB の 16 ビットは、ソフトウェア動作や DMA リクエストを必要とせずに（1 回の読込み／書込み動作のみで）強制的に 0 にされます。

24 ビットと 32 ビットのデータフレームは、SPIx_DR レジスタとの間で 2 回の CPU 読出しまたは書込み動作（あるいはアプリケーションにとって DMA が望ましい場合は 2 回の DMA 動作）を必要とします。特に 24 ビットのデータフレームの場合、8 つの下位ビットはハードウェアによって 0 ビットで 32 ビットに拡張されます。

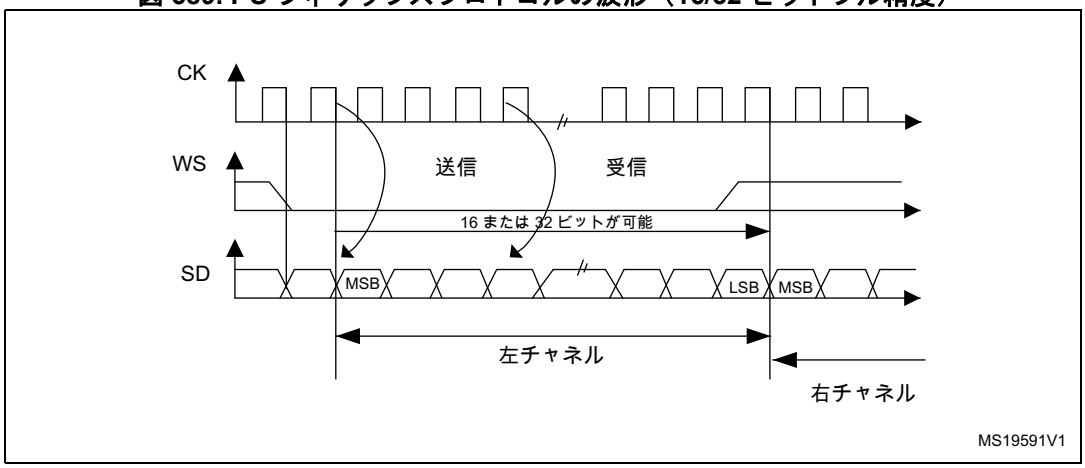
すべてのデータフォーマットと通信規格に対して、最上位ビットは常に最初に送信されます（MSB ファースト）。

I2S インタフェースは、SPIx_I2SCFGR レジスタの I2SSTD[1:0] と PCMSYNC ビットを使用して設定可能な 4 つのオーディオ規格をサポートします。

フィリップス I²S 規格

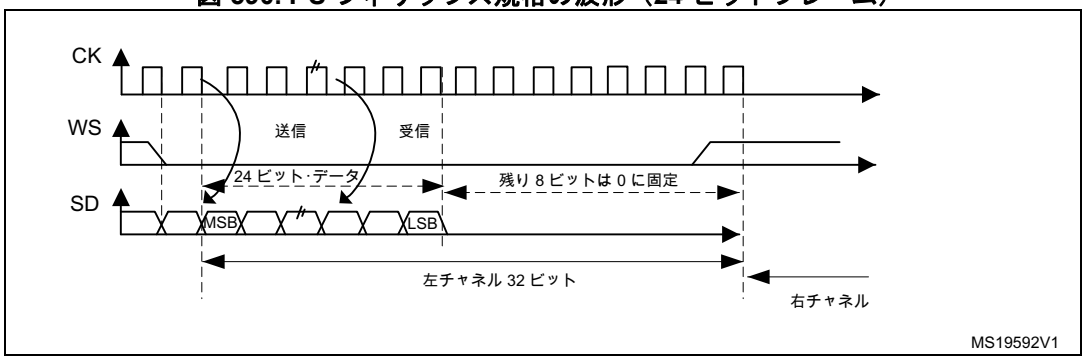
この規格では、どのチャネルが送信されているかを示すために WS 信号を使用します。この信号が有効になってから CKの1クロックサイクル後に最初のビット（MSB）が使用可能になります。

図 589. I²S フィリップスプロトコルの波形（16/32 ビットフル精度）



データは、CK の立ち下がりエッジでラッチされ（トランスミッタの場合）、立ち上がりエッジで読み出されます（レシーバの場合）。WS 信号も CK の立ち下がりエッジでラッチされます。

図 590. I²S フィリップス規格の波形（24 ビットフレーム）

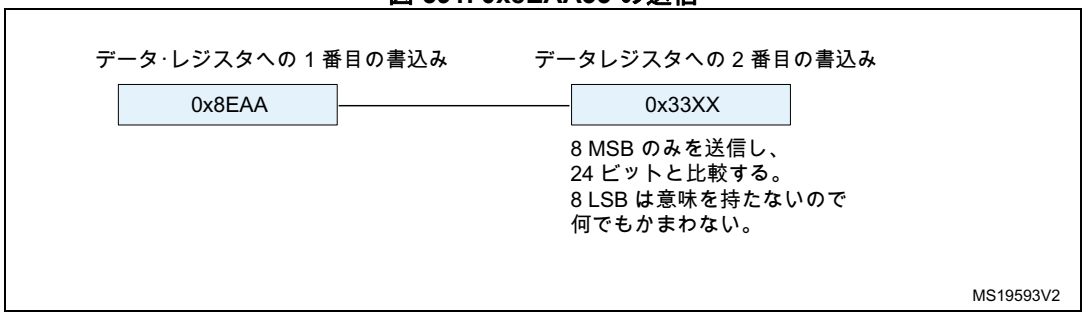


このモードでは、SPIx_DR レジスタに対して 2 回の書込みまたは読出し動作が必要です。

- 送信モードの場合

  0x8EAA33 を送信する必要がある場合（24 ビット）:

図 591. 0x8EAA33 の送信



- 受信モードの場合

  データ 0x8EAA33 が受信される場合：

図 592. 0x8EAA33 の受信

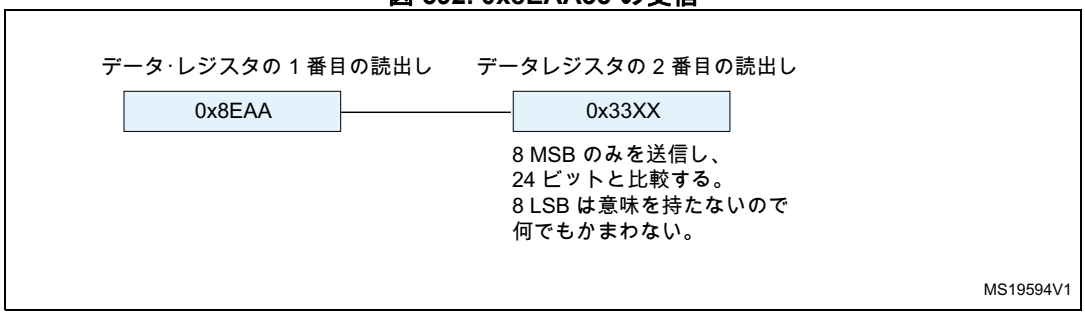
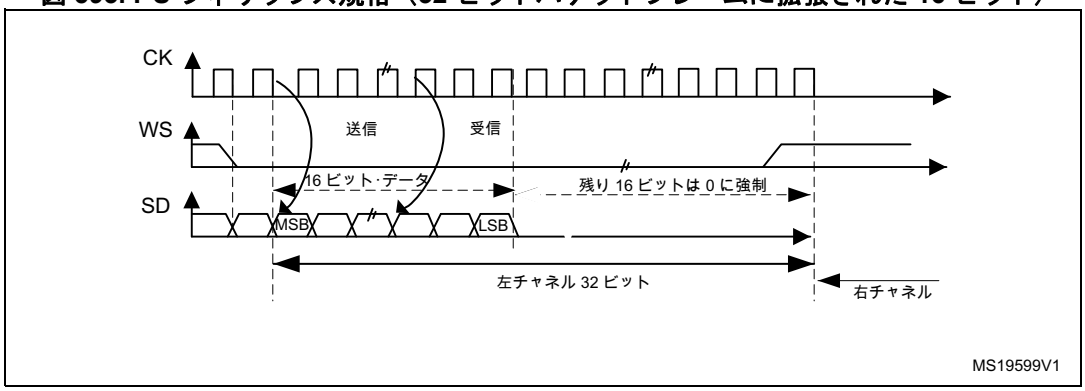


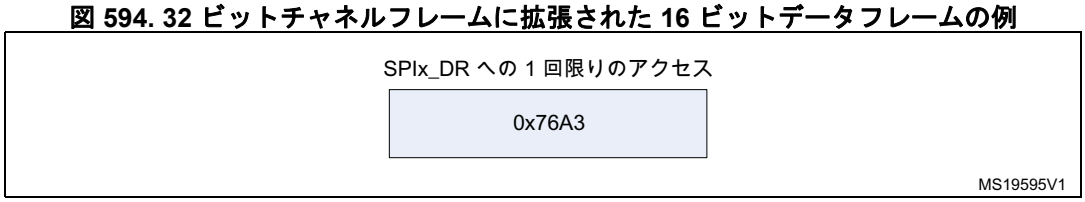
図 593. I²S フィリップス規格（32 ビットパケットフレームに拡張された 16 ビット）



I²S 設定フェーズで、32 ビットチャネルフレームに拡張された 16 ビットデータフレームが選択されたとき、SPIx_DR レジスタへのアクセスは 1 回のみ必要です。残りの 16 ビットは、データを 32 ビットフォーマットに拡張するために、ハードウェアによって強制的に 0x0000 にされます。

送信するデータまたは受信したデータが 0x76A3（32 ビットに拡張された 0x76A30000）であるとき、図 594 に示す動作が要求されます。

図 594. 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例



送信では、SPIx_DR に MSB が書き込まれるたびに、TXE フラグがセットされ、可能ならば、SPIx_DR レジスタに新しい送信値をロードするために割込みが生成されます。これはハードウェアによって行われるため、0x0000 がまだ送信されていなくても行われます。

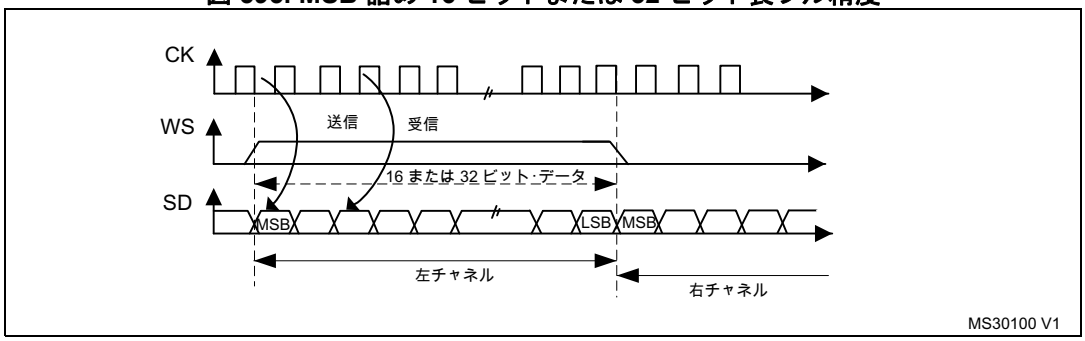
受信では、RXNE フラグがセットされ、可能ならば、最初の 16 MSB ハーフワードの受信時にその割込みが生成されます。

このように、2 回の書込みまたは読出し動作の間にはより多くの時間が設けられるため、アンダーランまたはオーバーラン状態（データ転送の方向に依存）を避けられます。

MSB 詰め規格

この規格では、WS 信号は最初のデータビット（最上位ビット）と同時に生成されます。

図 595. MSB 詰め 16 ビットまたは 32 ビット長フル精度



データは、CK の立ち下がりエッジでラッチされ（トランスミッタの場合）、立ち上がりエッジで読み出されます（レシーバの場合）。

図 596. MSB 詰め 24 ビットフレーム長

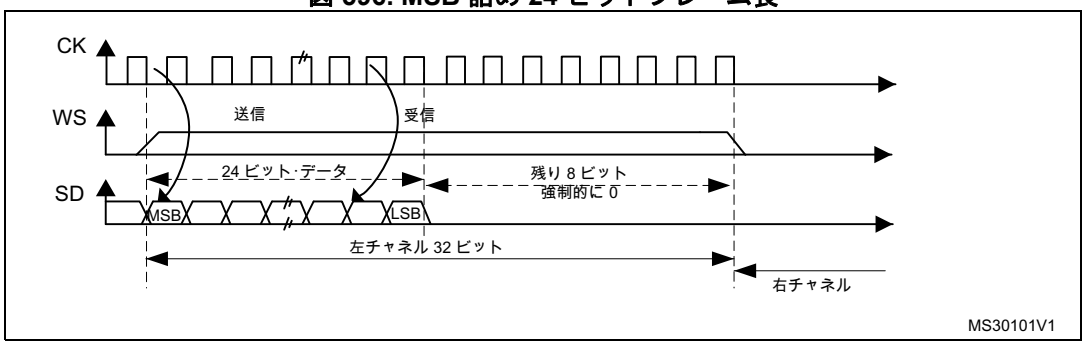
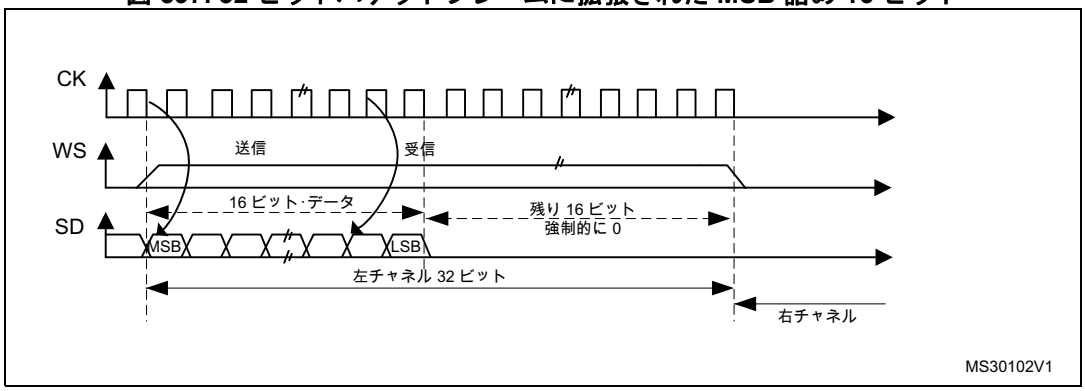


図 597. 32 ビットパケットフレームに拡張された MSB 詰め 16 ビット



LSB 詰め規格

この規格は、MSB 詰め規格と似ています（16 ビットと 32 ビットのフル精度フレームフォーマットに違いはありません）。

入出力信号のサンプリングは I^2S フィリップス規格と同じです。

図 598. LSB 詰め 16 ビットまたは 32 ビットフル精度

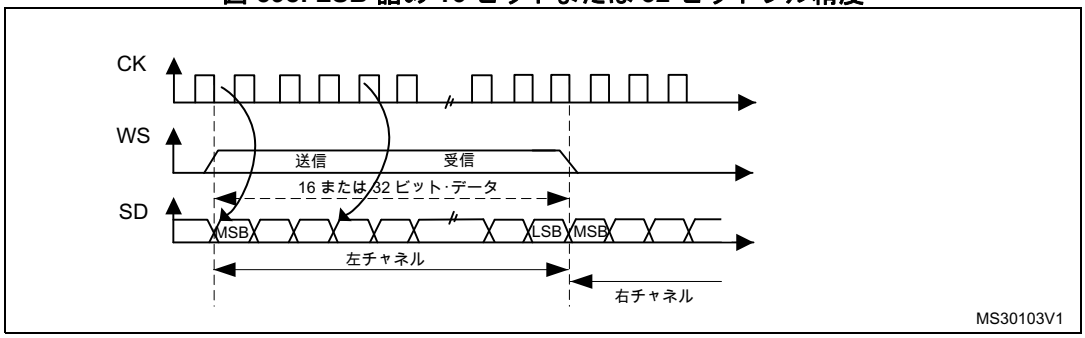
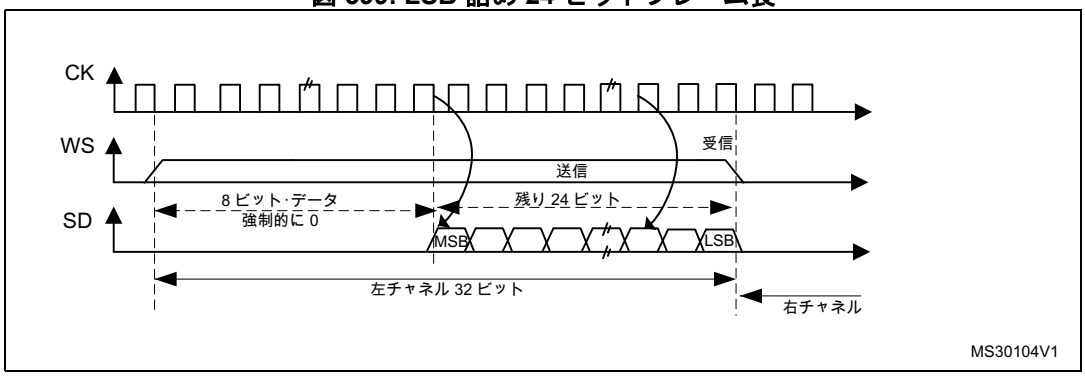


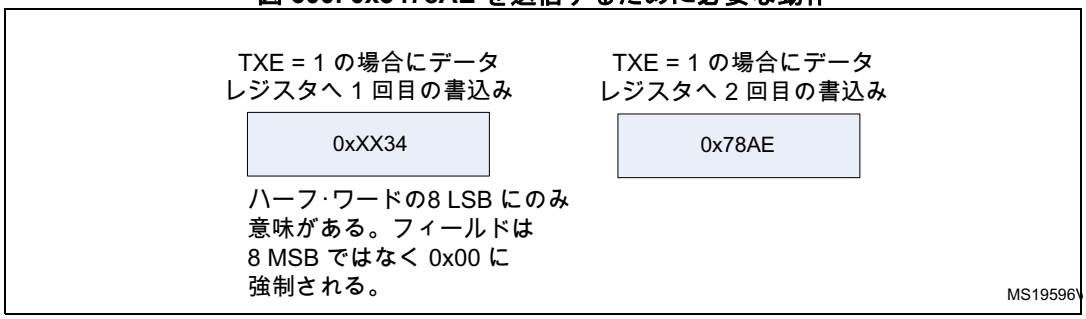
図 599. LSB 詰め 24 ビットフレーム長



- 送信モードの場合

  データ 0x3478AE を送信する必要がある場合、ソフトウェアまたは DMA によって SPIx_DR レジスタへの 2 回の書込み動作が必要です。この動作を次に示します。

図 600. 0x3478AE を送信するために必要な動作



- 受信モードの場合

  データ 0x3478AE が受信される場合、RXNE イベントごとに、SPIx_DR レジスタから連続する 2 回の読出し動作が必要です。

図 601. 0x3478AE の受信に必要な動作

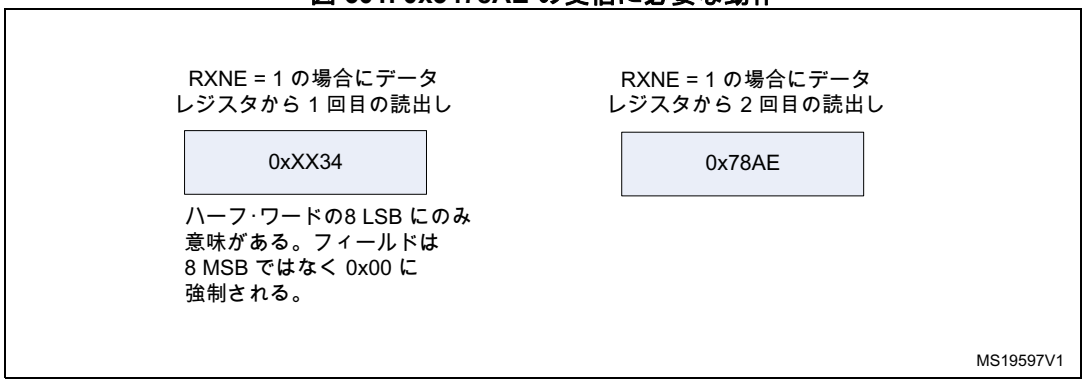
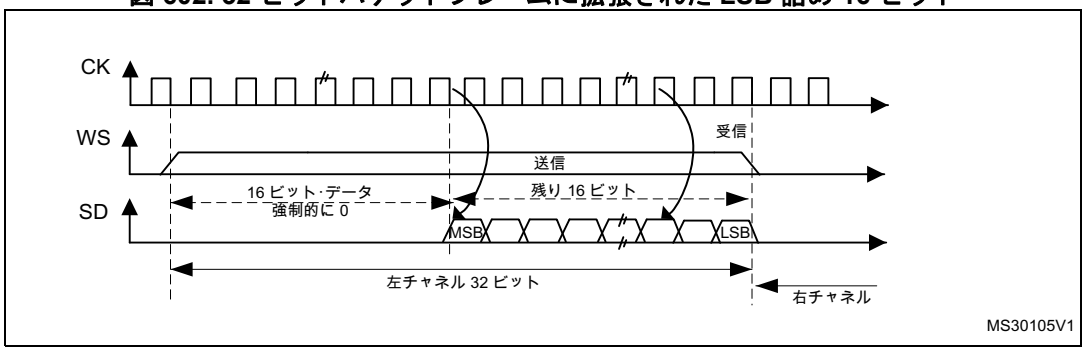


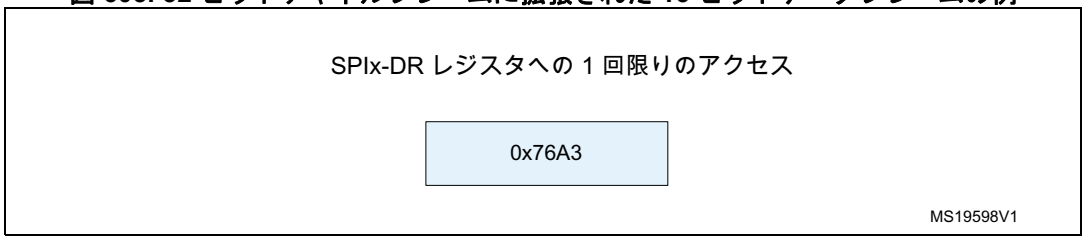
図 602. 32 ビットパケットフレームに拡張された LSB 詰め 16 ビット



I2S 設定フェーズで、32 ビットチャネルフレームに拡張された 16 ビットデータフレームが選択されたとき、SPIx_DR レジスタへのアクセスは 1 回のみ必要です。残りの 16 ビットは、データを 32 ビットフォーマットに拡張するために、ハードウェアによって強制的に 0x0000 にされます。この場合、それはハーフワード MSB に相当します。

送信するデータまたは受信したデータが 0x76A3（32 ビットに拡張された 0x0000 76A3）であるとき、図 603 に示す動作が要求されます。

図 603. 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例



送信モードでは、TXE イベントが発生すると、アプリケーションは送信するデータ（この場合は 0x76A3）を書き込む必要があります。0x000 フィールドが最初に送信されます（32 ビット拡張部）。有効なデータ（0x76A3）が SD に送信されると、すぐに TXEフラグ が再びセットされます。

受信モードでは、0x0000 フィールドではない有効なハーフワードが受信されると、すぐに RXNE がアサートされます。

このように、アンダーランやオーバーランの状態を防ぐために、2 回の書込みまたは読出し動作の間には、より多くの時間が設けられています。

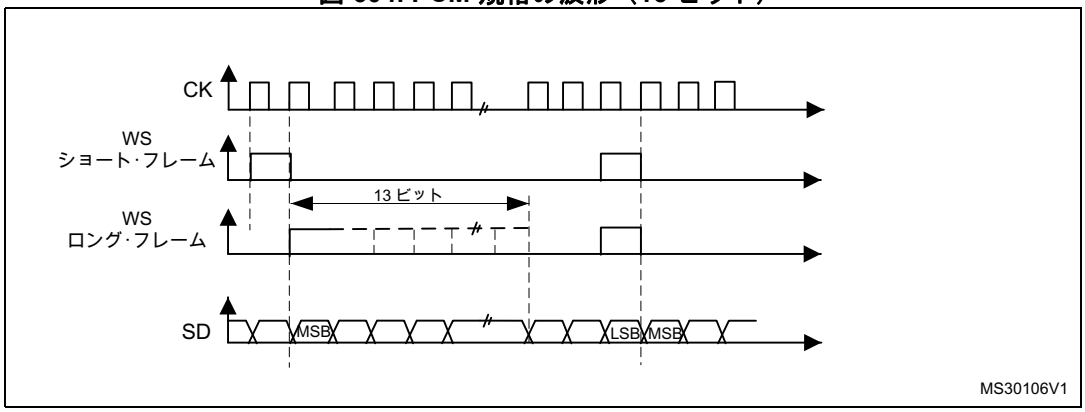
PCM 規格

PCM 規格では、チャネルサイド情報を使用する必要はありません。SPIx_I2SCFGR レジスタの PCMSYNC ビットを使用して、2 つの PCM モード（ショートおよびロングフレーム）の使用および設定が可能です。

PCM モードで、出力信号（WS、SD）は CK 信号の立ち上がりエッジでサンプリングされます。入力信号（WS、SD）は立ち下がりエッジでキャプチャされます。

CK および WS は、マスタモードで出力として設定されることに注意してください。

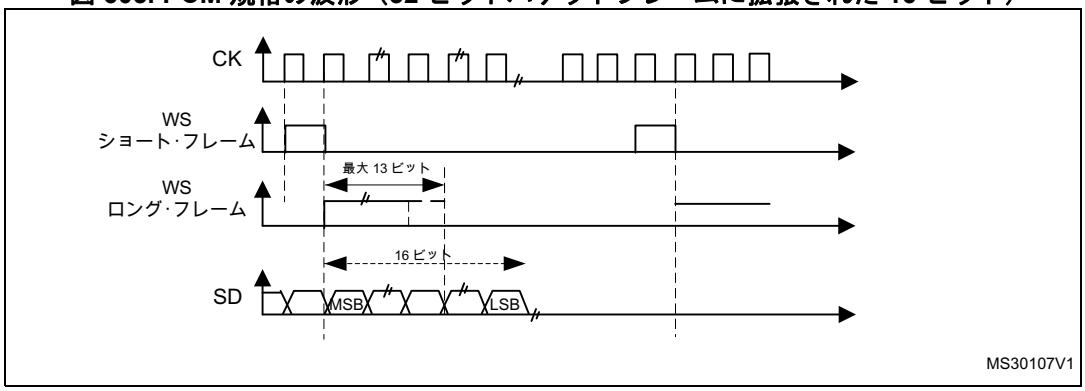
図 604. PCM 規格の波形（16 ビット）



ロングフレーム同期では、WS 信号のアサーション時間はマスタモードで 13 ビットに固定されています。

ショートフレーム同期では、WS 同期信号の長さは、わずか 1 サイクルです。

図 605. PCM 規格の波形（32 ビットパケットフレームに拡張された 16 ビット）

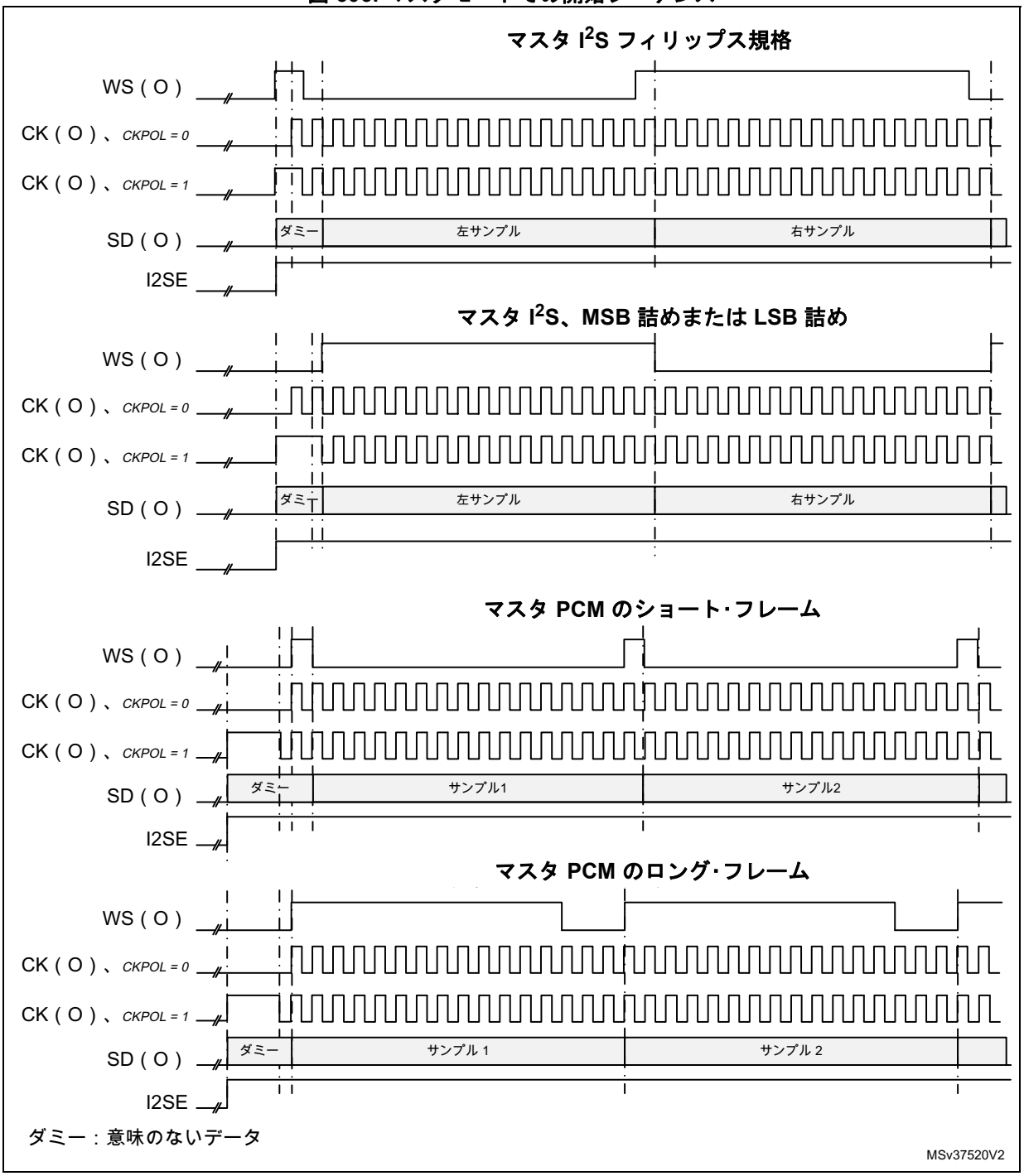


注： **2 つのモード（マスタとスレーブ）と 2 つの同期（ショートとロング）に関しては、スレーブモードでも、連続した 2 つのデータ（したがって 2 つの同期信号）間のビット数を（SPIx_I2SCFGR レジスタの DATLEN および CHLEN ビットで）指定する必要があります。**

39.7.3 起動に関する説明

図 606 は、SPI/I2S が有効化された場合（I2SE ビット）の MASTER モードでのシリアルインタフェースの処理について示しています。生成された信号での CKPOL の影響についても示します。

図 606. マスタモードでの開始シーケンス



スレーブモードで、フレームの同期の検出方法は ASTRTEN ビットの値に依存します。

ASTRTEN = 0 の場合、オーディオインタフェースを有効（I2SE = 1）にすると、ハードウェアは CK 信号を使用して受信 WS 信号で適切な遷移が行われるのを待ちます。

WS 信号での適切な遷移は、I^2S フィリップス規格が使用されている場合は立ち下がりエッジ、それ以外の規格の場合は立ち上がりエッジです。立ち下がりエッジは、最初の WS を 1 にサンプリングし、続いて 0 にサンプリングすることで検出されます。立ち上がりエッジの検出方法はこの逆です。

ASTRTEN = 1 の場合、ユーザは WS がアクティブになる前にオーディオインタフェースを有効にする必要があります。これは、I^2S フィリップス規格が WS = 1、またはその他の規格が WS = 0 の場合に、I2SE ビットを 1 にセットする必要があることを意味しています。

39.7.4 クロックジェネレータ

I^2S ビットレートにより、I^2S データライン上のデータフローと I^2S クロック信号周波数が決まります。

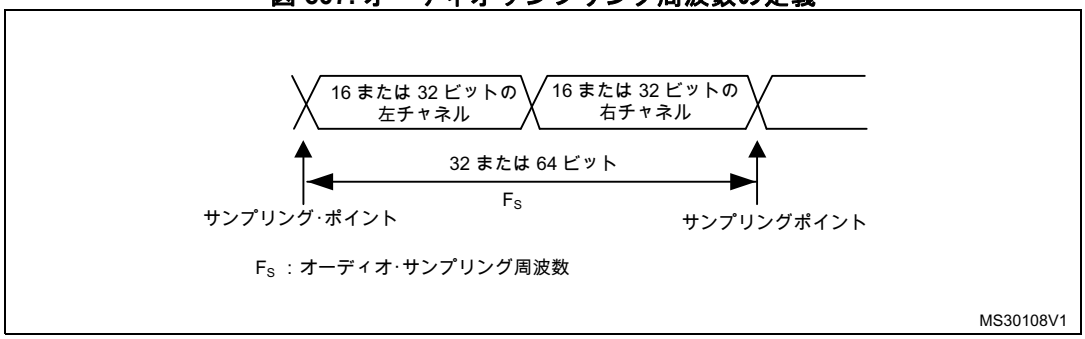
I^2S ビットレート = チャネルあたりのビット数×チャネル数×オーディオサンプリング周波数

16 ビットオーディオ、左チャネルおよび右チャネルの場合、I^2S ビットレートは次のように算出されます。

$$I^2S \text{ ビットレート} = 16 \times 2 \times f_S$$

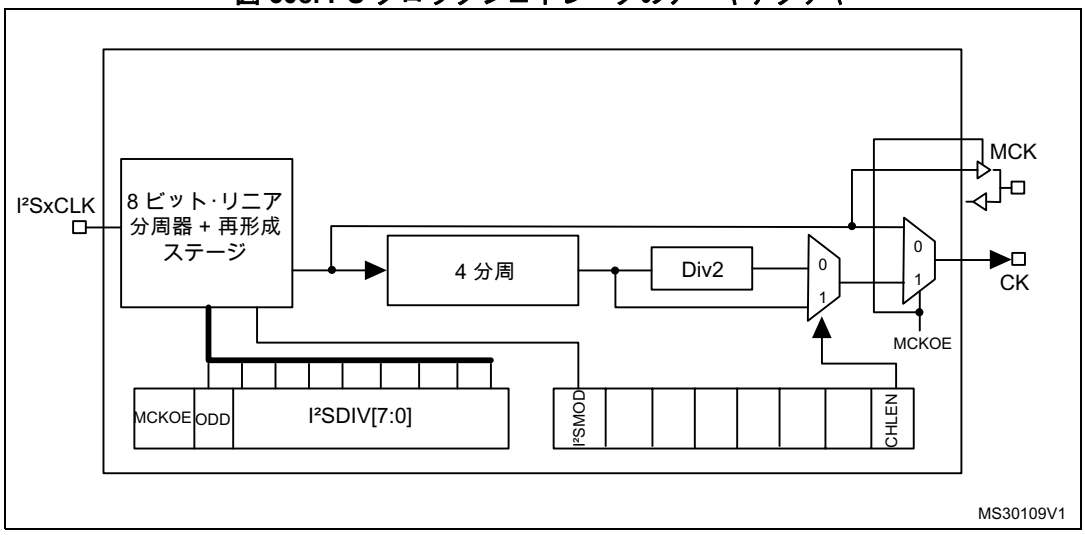
パケット長が 32 ビットの場合、I^2S ビット･レート = $32 \times 2 \times f_S$ となります。

図 607. オーディオサンプリング周波数の定義



マスタモードが設定された場合、希望するオーディオ周波数で通信するために、特定の措置を講じてリニアディバイダを適切にプログラムする必要があります。

図 608. I²S クロックジェネレータのアーキテクチャ



1. x は 2 または 3 です。

図 608 に、通信クロックのアーキテクチャを示します。I2SxCLK クロックは、製品のリセットおよびクロックコントローラ（RCC）によって供給されます。I2SxCLK クロックは、SPI/I2S APB クロックに対して非同期です。

警告： **さらに、I2SxCLK 周波数を SPI/I2S ブロックによって使用される APB クロック以上に保つ必要があります。この条件が満たされない場合、SPI/I2S は正しく動作しません。**

オーディオ·サンプリング周波数の取り得る値は、192 kHz、96 kHz、48 kHz、44.1 kHz、32 kHz、22.05 kHz、16 kHz、11.025 kHz、8 kHz（またはその他のこの範囲内にある値）。

希望する周波数を達成するには、次の式に基づいてリニアディバイダをプログラムする必要があります。

I²S モードの場合：

マスタクロックが生成される（SPIx_I2SPR レジスタの MCKOE ビットがセットされる）場合：

$$Fs = \frac{F_{I2SxCLK}}{256 \times ((2 \times I2SDIV) + ODD)}$$

マスタクロックが無効にされた場合（MCKOE ビットをクリア）

$$Fs = \frac{F_{I2SxCLK}}{32 \times (CHLEN + 1) \times ((2 \times I2SDIV) + ODD)}$$

チャネルフレームが 16 ビット幅のとき、CHLEN = 0

チャネルフレームが 32 ビット幅のとき、CHLEN = 1

PCM モードの場合：

マスタクロックが生成される（SPIx_I2SPR レジスタの MCKOE ビットがセットされる）場合：

$$Fs = \frac{F_{I2SxCLK}}{128 \times ((2 \times I2SDIV) + ODD)}$$

マスタクロックが無効にされた場合（MCKOE ビットをクリア）

$$Fs = \frac{F_{I2SxCLK}}{16 \times (CHLEN + 1) \times ((2 \times I2SDIV) + ODD)}$$

チャネルフレームが 16 ビット幅のとき、CHLEN = 0

チャネルフレームが 32 ビット幅のとき、CHLEN = 1

ここで、F_S はオーディオサンプリング周波数であり、$F_{I2SxCLK}$ は SPI/I2S ブロックに供給されるカーネルクロックの周波数です。

注： **I2SDIV は厳密に 1 より高くする必要があります。**

次の表に、さまざまなクロック設定に対する精度値の例を示します。

注： **最適クロック精度を可能にするその他の設定も可能です。**

表 362. HSE から派生した 48 MHz クロックを使用した場合のオーディオ周波数の精度(1)

SYSCLK（MHz）	データ長	I2SDIV	I2SODD	MCLK	ターゲット fs（Hz）	実際の fs (kHz)	誤差
48	16	8	0	不可	96000	93750	2.3438%
48	32	4	0	不可	96000	93750	2.3438%
48	16	15	1	不可	48000	48387.0968	0.8065%
48	32	8	0	不可	48000	46875	2.3438%
48	16	17	0	不可	44100	44117.647	0.0400%
48	32	8	1	不可	44100	44117.647	0.0400%
48	16	23	1	不可	32000	31914.8936	0.2660%
48	32	11	1	不可	32000	32608.696	1.9022%
48	16	34	0	不可	22050	22058.8235	0.0400%
48	32	17	0	不可	22050	22058.8235	0.0400%
48	16	47	0	不可	16000	15957.4468	0.2660%
48	32	23	1	不可	16000	15957.447	0.2660%
48	16	68	0	不可	11025	11029.4118	0.0400%
48	32	34	0	不可	11025	11029.412	0.0400%
48	16	94	0	不可	8000	7978.7234	0.2660%
48	32	47	0	不可	8000	7978.7234	0.2660%
48	16	2	0	可能	48000	46875	2.3430%

表 362. HSE から派生した 48 MHz クロックを使用した場合のオーディオ周波数の精度(1)（続き）

SYSCLK（MHz）	データ長	I2SDIV	I2SODD	MCLK	ターゲット fs（Hz）	実際の fs (kHz)	誤差
48	32	2	0	可能	48000	46875	2.3430%
48	16	2	0	可能	44100	46875	6.2925%
48	32	2	0	可能	44100	46875	6.2925%
48	16	3	0	可能	32000	31250	2.3438%
48	32	3	0	可能	32000	31250	2.3438%
48	16	4	1	可能	22050	20833.333	5.5178%
48	32	4	1	可能	22050	20833.333	5.5178%
48	16	6	0	可能	16000	15625	2.3438%
48	32	6	0	可能	16000	15625	2.3438%
48	16	8	1	可能	11025	11029.4118	0.0400%
48	32	8	1	可能	11025	11029.4118	0.0400%
48	16	11	1	可能	8000	8152.17391	1.9022%
48	32	11	1	可能	8000	8152.17391	1.9022%

1. この表の値は、さまざまなクロック設定に対するほんの一例です。最適クロック精度を可能にするその他の設定も可能です。

39.7.5 I^2S マスタモード

I2S はマスタモードで設定できます。つまり、シリアルクロックは、ワードセレクト信号 WS だけでなく、CK ピン上でも生成されます。マスタクロック（MCK）は、SPIx_I2SPR レジスタの MCKOE ビットで、出力するかしないかを制御できます。

手順

1. 適切なオーディオサンプリング周波数に到達するシリアルクロックボーレートを定義するため、SPIx_I2SPR レジスタの I2SDIV[7:0] ビットを選択します。SPIx_I2SPR レジスタの ODD ビットも定義する必要があります。
2. 通信クロックの一定したレベルを定義するために、CKPOL ビットを選択します。外部の DAC/ADC オーディオコンポーネントにマスタクロック MCK を供給する必要がある場合、SPIx_I2SPR レジスタの MCKOE ビットをセットします（I2SDIV と ODD の値は、MCK 出力の状態に応じて計算する必要があります。詳細については、セクション 39.7.4：クロックジェネレータを参照）。
3. SPIx_I2SCFGR レジスタのI2SMOD ビットをセットして I2S 機能を有効にし、I2SSTD[1:0] と PCMSYNC ビットにより I^2S 規格を、DATLEN[1:0] ビットによりデータ長を、CHLEN ビットを設定してチャネルあたりのビット数を、それぞれ選択します。SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットにより I^2S マスタモードと方向（トランスミッタまたはレシーバ）も選択します。
4. 必要な場合は、SPIx_CR2 レジスタに書き込むことによって、可能性のあるすべての割込みソースと DMA 機能を選択します。
5. SPIx_I2SCFGR レジスタの I2SE ビットをセットする必要があります。

WS と CK は出力モードに設定されます。SPIx_I2SPR の MCKOE ビットがセットされている場合、MCK も出力です。

送信シーケンス

送信シーケンスは、Txバッファにハーフワードが書き込まれたときに開始されます。

Txバッファに書き込まれる最初のデータは、左チャネルのデータに対応すると想定します。データがTxバッファからシフトレジスタに転送されると、TXE がセットされ、右チャネルに対応するデータをTxバッファに書き込む必要があります。CHSIDE フラグは、どのチャネルが送信されるかを示します。CHSIDE フラグは TXE がハイレベルになったときに更新されるため、このフラグは TXE フラグがセットされた場合に意味を持ちます。

完全なフレームとは、左チャネルのデータ送信と、それに続く右チャネルのデータ送信であるとみなす必要があります。左チャネルのみが送信される部分的フレームは実現できません。

データハーフワードは、最初のビット送信時に 16 ビットシフトレジスタに同時にロードされてから、MOSI/SD ピンに MSB ファーストで連続的にシフトアウトされます。TXE フラグは、Txバッファからシフトレジスタへの毎回の転送後にセットされ、SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は割込みが生成されます。

選択された I^2S 規格モードに応じた書込み動作の詳細については、セクション 39.7.2 : サポートされるオーディオプロトコルを参照してください。

連続したオーディオデータ送信を行うには、現在の送信が終了する前に、次の送信データを SPIx_DR レジスタに書きこむ必要があります。

I2SE をクリアすることによって I2S をスイッチオフするには、TXE = 1 および BSY = 0 になるまで待つ必要があります。

受信シーケンス

動作モードは、ポイント 3 を除いて、送信モードの場合と同じです（セクション 39.7.5 : I2S マスタモードに示す手順を参照）。ポイント 3 では、I2SCFG[1:0] ビットを通じてマスタ受信モードを設定する必要があります。

データやチャネルの長さに関係なく、オーディオデータは 16 ビットのパケットによって受信されます。つまり、Rxバッファがフルになるたびに RXNE フラグがセットされ、SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は割込みが生成されます。データ長とチャネル長の設定にもよりますが、Rxバッファへの 1 回または 2 回の受信によって、オーディオ値が右チャネルまたは左チャネルに受信されることがあります。

RXNE ビットは、SPIx_DR レジスタの読出しによってクリアされます。

CHSIDE は毎回の受信後に更新されます。CHSIDE は、I2S セルによって生成される WS 信号に反応します。

選択された I^2S 規格モードに応じた読出し動作の詳細については、セクション 39.7.2 : サポートされるオーディオプロトコルを参照してください。

前の受信データがまだ読み出されていない間にデータが受信された場合、オーバーランが生成され、OVR フラグがセットされます。SPIx_CR2 レジスタの ERRIE ビットがセットされている場合、割込みが生成されてエラーを知らせます。

I^2S をスイッチオフするには、I2S が新しいデータ転送を開始することなく転送サイクルを適切に完了できるように、特定の動作が要求されます。そのシーケンスは、データ長とチャネル長の設定、および選択したオーディオプロトコルモードに依存します。ケース別の説明

- 32 ビットチャネル長に拡張された 16 ビットデータ長(DATLEN = 00、CHLEN = 1)、LSB 詰めモードを使用（I2SSTD = 10）
  a) 最後から 2 番目の RXNE = 1（n – 1）を待ちます。
  b) 次に I2S の 17 クロックサイクルを待ちます（ソフトウェアループを使用）。

c) I2S を無効にします（I2SE = 0）。

- 32 ビットチャネル長に拡張された 16 ビットデータ長（DATLEN = 00 および CHLEN = 1）、MSB 詰め、I²S または PCM モード（それぞれ、I2SSTD = 00、I2SSTD = 01、または I2SSTD = 11）
  a) 最後の RXNE を待ちます。
  b) 次に I2S の 1 クロックサイクルを待ちます（ソフトウェアループを使用）。
  c) I2S を無効にします（I2SE = 0）。
- DATLEN と CHLEN のその他すべての組み合わせについては、I2SSTD ビットを通じて選択したオーディオモードが何であれ、次のシーケンスを実行して I2S をスイッチオフします。
  a) 最後から 2 番目の RXNE = 1（n – 1）を待ちます。
  b) 次に I2S の 1 クロックサイクルを待ちます（ソフトウェアループを使用）。
  c) I2S を無効にします（I2SE = 0）。

注： **転送時、BSY フラグはローレベルに保持されます。**

39.7.6 I²S スレーブモード

スレーブ設定の場合、I²S は送信または受信モードに設定できます。動作モードは、主として I²S マスタ設定に関して述べたものと同じ規則に従います。スレーブモードでは、I²S インタフェースによって生成されるクロックはありません。クロックと WS 信号は、I²S インタフェースに接続された外部マスタから入力されます。したがって、ユーザがクロックを設定する必要はありません。

設定ステップを次に示します。

1. SPIx_I2SCFGR レジスタの I2SMOD ビットをセットして I²S モードを選択し、I2SSTD[1:0] ビットにより I²S 規格を、DATLEN[1:0] ビットによりデータ長を、CHLEN ビットを設定してフレームのチャネルあたりのビット数を、それぞれ選択します。SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットにより、スレーブのモード（送信または受信）も選択します。
2. 必要な場合は、SPIx_CR2 レジスタに書き込むことによって、可能性のあるすべての割込みソースと DMA 機能を選択します。
3. SPIx_I2SCFGR レジスタの I2SE ビットをセットする必要があります。

送信シーケンス

送信シーケンスは、外部マスタデバイスがクロックを送信したときと、NSS_WS 信号がデータの転送を要求したときに開始されます。スレーブを有効にしなければ、外部マスタは通信を開始しません。I²S データレジスタにロードしなければ、マスタは通信を開始しません。

I²S、MSB 詰め、および LSB 詰めモードの場合、データレジスタに書き込まれる最初のデータは、左チャネルのデータに対応します。通信が開始されると、データはTxバッファからシフトレジスタに転送されます。次に、右チャネルのデータを I2S データレジスタに書き込むように要求するために、TXE フラグがセットされます。

CHSIDE フラグは、どのチャネルが送信されるかを示します。スレーブモードでは、マスタ送信モードに比べて、CHSIDE は外部マスタからの WS 信号に反応します。つまり、スレーブが最初のデータの送信準備をしなければ、マスタはクロックを生成できません。WS アサーションは、最初に送信される左チャネルに対応します。

注： **I2SE は、マスタの最初のクロックが CK ラインに到達するよりも、少なくとも 2 PCLK サイクル前に書き込まれる必要があります。**

データハーフワードは、最初のビット送信時に内部バスから 16 ビットシフトレジスタに同時にロードされてから、MOSI/SD ピンに MSB ファーストで連続的にシフトアウトされます。TXE フラグは、Txバッファからシフトレジスタへの毎回の転送後にセットされ、SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は割込みが生成されます。

なお、Txバッファへの書込みの前に、TXE フラグが 1 であることを確認する必要があります。

選択された I²S 規格モードに応じた書込み動作の詳細については、セクション 39.7.2 : サポートされるオーディオプロトコルを参照してください。

連続したオーディオデータ送信を行うには、現在の送信が終了する前に、次の送信データを SPIx_DR レジスタに書きこむ必要があります。次のデータ通信の最初のクロックエッジよりも前にデータが SPIx_DR レジスタに書き込まれない場合、アンダーランフラグがセットされ、割込みが生成されることがあります。これによって、転送データに誤りがあることがソフトウェアに知らされます。SPIx_CR2 レジスタの ERRIE ビットがセットされた場合、SPIx_SR レジスタの UDR フラグがハイレベルになると割込みが生成されます。この場合、I2S をスイッチオフし、左チャネルからデータ転送をリスタートする必要があります。

I2SE ビットをクリアすることによって I2S をスイッチオフするには、TXE = 1 および BSY = 0 になるまで待つ必要があります。

受信シーケンス

動作モードは、ポイント 1 を除いて、送信モードの場合と同じです（ セクション 39.7.6 : I2S スレーブモードに示す手順を参照）。ポイント 1 では、SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットを通じてマスタ受信モードを設定する必要があります。

データやチャネルの長さに関係なく、オーディオデータは 16 ビットのパケットによって受信されます。つまり、RXバッファがフルになるたびに SPIx_SR レジスタの RXNE フラグがセットされ、SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は割込みが生成されます。データ長とチャネル長の設定にもよりますが、RXバッファへの 1 回または 2 回の受信によって、オーディオ値が右チャネルまたは左チャネルに受信されることがあります。

CHSIDE フラグは、SPIx_DR レジスタから読み出されるデータが受信されるたびに更新されます。このフラグは、外部マスタコンポーネントによって管理される外部 WS ラインに反応します。

RXNE ビットは、SPIx_DR レジスタの読出しによってクリアされます。

選択された I²S 規格モードに応じた読出し動作の詳細については、セクション 39.7.2 : サポートされるオーディオプロトコルを参照してください。

前の受信データがまだ読み出されていない間にデータが受信された場合、オーバーランが生成され、OVR フラグがセットされます。SPIx_CR2 レジスタの ERRIE ビットがセットされた場合、エラーを示すために割込みが生成されます。

受信モードで I2S をスイッチオフするには、最後の RXNE = 1 を受信した直後に I2SE をクリアする必要があります。

注 : **外部マスタコンポーネントには、オーディオチャネルを介して 16 ビットまたは 32 ビットのパケットでデータを送受信する機能が必要です。**

39.7.7 I2S ステータスフラグ

アプリケーションが I2S バスの状態を完全に監視できるように、3 つのステータスフラグが用意されています。

ビジーフラグ（BSY）

BSY フラグは、ハードウェアによってセット／クリアされます（このフラグへの書込みは無効）。このフラグは I2S の通信層の状態を示します。

BSY がセットされると、I2S が通信中でビジーであることを示します。マスタ受信モード（I2SCFG= 11）には 1 つの例外があり、BSY フラグは受信時にローレベルに保持されます。

ソフトウェアが I2S を無効にする必要がある場合､BSY フラグは転送の終わりを検出するために役立ちます。これによって、最後の転送データの破壊を回避します。このため、下記の手順を厳守する必要があります。

BSY フラグは転送が開始されるとセットされます。ただし、I2S がマスタ受信モードにある場合を除きます。

BSY フラグをクリアするタイミング

- 転送が完了したとき（ただし、通信が連続的だと思われるマスタ送信モードの場合を除く）
- I^2S が無効にされたとき

通信が連続的な場合

- マスタ送信モードでは、BSY フラグはすべての転送期間を通じてハイレベルに保持されます。
- スレーブモードでは、BSY フラグは、各転送間で I2S の 1 クロックサイクルの間ローレベルになります。

注： **各データの送受信の処理には BSY フラグを使用しないでください。代わりに、TXE フラグと RXNE フラグを使用することをお勧めします。**

Txバッファエンプティフラグ（TXE）

このフラグがセットされると、Txバッファはエンプティ（空）であり、次に送信するデータをバッファにロードできることを示します。送信されるデータがすでにTxバッファに格納されているとき、TXE フラグはリセットされます。TXE フラグは、I2S が無効にされている（I2SE ビットがリセット）ときにもリセットされます。

RXバッファノットエンプティ（RXNE）

このフラグがセットされると、RXバッファに有効な受信データがあることを示します。このフラグは、SPIx_DR レジスタが読み出されるとリセットされます。

チャネルサイドフラグ（CHSIDE）

送信モードでは、このフラグは TXE がハイレベルになるとリフレッシュされます。このフラグは、SD 上の転送データが属するチャネルサイドを示します。スレーブ送信モードでアンダーランエラーイベントが発生した場合、このフラグは信頼できないため、通信を再開する前に、I2S をスイッチオフし、さらにスイッチオンする必要があります。

受信モードでは、このフラグは SPIx_DR にデータが受信されるとリフレッシュされます。このフラグは、どちらのチャネルサイドからデータが受信されたかを示します。なお、エラー（OVR など）が発生した場合、このフラグは無意味になるため、I2S を無効にし、さらに有効にすることによってリセットする必要があります（変更が必要な場合は設定します）。

このフラグは、PCM 規格では意味を持ちません（ショートとロングフレームの両モード）。

SPIx_SR の OVR または UDR フラグがセットされ、SPIx_CR2 の ERRIE ビットもセットされると、割込みが生成されます。この割込みをクリアするには、割込みソースをいったんクリアした後で、SPIx_SR ステータスレジスタを読み出します。

39.7.8 I²S エラーフラグ

I²S には 3つのエラーフラグがあります。

アンダーランフラグ（UDR）

スレーブ送信モードでは、ソフトウェアが SPIx_DR にまだ値をロードしていない間に、データ送信用の最初のクロックが現われると、このフラグがセットされます。このフラグは、SPIx_I2SCFGR レジスタの I2SMOD ビットがセットされると使用できます。SPIx_CR2 レジスタの ERRIE ビットがセットされている場合は、割込みを生成できます。
UDR ビットは、SPIx_SR レジスタの読出し動作によってクリアされます。

オーバーランフラグ（OVR）

このフラグがセットされるのは、データが受信され、前のデータが SPIx_DR レジスタからまだ読み出されていないときです。結果として、受信データは失われます。SPIx_CR2 の ERRIE ビットがセットされている場合、割込みが生成されることがあります。

この場合、受信バッファの内容は、送信側のデバイスからの新しい受信データによって更新されません。SPIx_DR レジスタへの読出し動作によって、前に正しく受信されたデータが返されます。それ以降に送信されたすべてのハーフワードは失われます。

OVR ビットをクリアするには、SPIx_DR レジスタを読み出し、続けて SPIx_SR レジスタに読出しアクセスを行います。

フレームエラーフラグ（FRE）

このフラグは、I²S がスレーブモードに設定された場合にのみハードウェアによってセットすることができます。このフラグは、スレーブが WS ラインの変化を想定していないときに外部マスタがWS ラインを変化させようとしている場合にセットされます。同期が失われた場合、この状態から回復し、外部マスタデバイスと I²S スレーブデバイスを再同期させるためには、次のステップに従います。

1. I²S を無効化します。
2. 正しいレベルが WS ラインで検出されたとき（WS ラインは I²S モードではハイレベル、MSB 詰めまたは LSB 詰めあるいは PCM モードではローレベル）、再度有効にします。

マスタデバイスとスレーブデバイスの間の同期外れは、CK 通信クロック上または WS フレーム同期ライン上のノイズの多い環境に起因する可能性があります。ERRIE ビットをセットすれば、エラー割込みを生成させることができます。同期外れフラグ（FRE）は、ステータスレジスタを読み出すときに、ソフトウェアでクリアされます。

39.7.9 DMA の機能

I²S モードでは、DMA は SPI モードとまったく同じように機能します。異なる点は、I²S モードではデータ転送保護システムがないため、CRC 機能を使用できない点のみです。

39.8 I²S 割込み

表 363 に I²S 割込みのリストを示します。

表 363. I²S 割込みリクエスト

割込みイベント	イベントフラグ	有効制御ビット
送信バッファエンプティフラグ	TXE	TXEIE
受信バッファ·ノット·エンプティ·フラグ	RXNE	RXNEIE
オーバーラン·エラー	OVR	ERRIE
アンダーラン·エラー	UDR	
フレーム·エラー·フラグ	FRE	

39.9 SPI および I²S レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。さらに、SPI_DR へは 8 ビット単位でアクセスできます。

39.9.1 SPI 制御レジスタ 1（SPIx_CR1）

アドレス･オフセット：0x00

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIDI MODE	BIDIOE	CRC EN	CRCNE XT	CRCL	RX ONLY	SSM	SSI	LSB FIRST	SPE	BR[2:0]			MSTR	CPOL	CPHA
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 **BIDIMODE**：双方向データモードイネーブル。

このビットは、共通の双方向データラインを 1 本使用して、半二重通信を有効にします。双方向モードがアクティブのときは、RXONLY ビットをクリアされたままにします。

0：2 線単方向データ･モードを選択します。

1：1 線双方向データ･モードを選択します。

注： このビットは I²S モードでは使用しません。

ビット 14 **BIDIOE**：双方向モードでの出力イネーブル

双方向モードでの転送方向は、このビットと BIDIMODE ビットを組み合わせて選択します。

0：出力は無効です（受信専用モード）。

1：出力は有効です（送信専用モード）。

注： マスタモードでは MOSI ピンが使用され、スレーブモードでは MISO ピンが使用されます。
このビットは I²S モードでは使用しません。

ビット 13 **CRCEN**：ハードウェア CRC 計算イネーブル

0：CRC 計算は無効です。

1：CRC 計算は有効です。

注： 正しい動作のためには、このビットへの書込みは、SPI が無効（SPE = 0）のときにのみ行います。
このビットは I²S モードでは使用しません。

ビット 12 **CRCNEXT**：送信 CRC Next

0：次の送信値は Tx バッファから送信されます。

1：次の送信値は Tx CRC レジスタから送信されます。

注： このビットは、最後のデータが SPIx_DR レジスタに書き込まれた直後に書き込む必要があります。
このビットは I²S モードでは使用しません。

ビット 11 **CRCL**：CRC 長

このビットは、CRC 長をセットするために、ソフトウェアによってセット／クリアされます。

0：8 ビットの CRC 長

1：16 ビットの CRC 長

注： 正しい動作のためには、このビットへの書込みは、SPI が無効（SPE = 0）のときにのみ行います。
このビットは I²S モードでは使用しません。

ビット 10 **RXONLY**：受信専用モード･イネーブル

このビットは、データ受信専用の単方向ラインを 1 本使用して、単方向通信を有効にします。受信専用モードがアクティブのときは、BIDIMODE ビットをクリアされたままにします。このビットはマルチスレーブシステムでも役立ちます。そのシステムでは、この特定のスレーブはアクセスされず、アクセスされたスレーブからの出力は破壊されません。

0：全二重（送受信）

1：出力は無効です（受信専用モード）。

注： このビットは I²S モードでは使用しません。

ビット 9 **SSM**：ソフトウェア･スレーブ管理

SSM ビットがセットされているとき、NSS ピンの入力は SSI ビットからの値に置き換えられます。

0：ソフトウェア･スレーブ管理は無効です。

1：ソフトウェア･スレーブ管理は有効です。

注： このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 8 **SSI**：内部スレーブ選択

このビットは、SSM ビットがセットされているときにのみ有効です。このビットの値は強制的に NSS ピンに設定され、NSS ピンの I/O 値は無視されます。

注： このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 7 **LSBFIRST**：フレームフォーマット

0：データは MSB から先に送信/受信されます。

1：データは LSB から先に送信/受信されます。

注： 1.このビットは、通信中には変更しないでください。

2.このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 6 **SPE**：SPI 有効化

0：ペリフェラルは無効です。

1：ペリフェラルは有効です。

注： SPI を無効にするときは、1714 ページのSPI を無効にする手順に記載されている手順に従ってください。

このビットは I²S モードでは使用しません。

ビット 5:3 **BR[2:0]**：ボー･レート制御

000：$f_{PCLK}/2$

001：$f_{PCLK}/4$

010：$f_{PCLK}/8$

011：$f_{PCLK}/16$

100：$f_{PCLK}/32$

101：$f_{PCLK}/64$

110：$f_{PCLK}/128$

111：$f_{PCLK}/256$

注： これらのビットは、通信中には変更しないでください。

これらのビットは I²S モードでは使用されません。

ビット 2 **MSTR**：マスタ選択

0：スレーブ設定

1：マスタ設定

注： このビットは、通信中には変更しないでください。

このビットは I²S モードでは使用しません。

ビット 1 **CPOL**：クロック極性

0：アイドル時に CK は 0 となります。

1：アイドル時に CK は 1 となります。

注： このビットは、通信中には変更しないでください。

このビットは、CRC が TI モードで適用される場合を除き、I²S モードおよび SPI TI モードでは使用されません。

ビット 0 **CPHA**：クロック位相

0：最初のクロック遷移が最初のデータキャプチャエッジです。

1：2 番目のクロック遷移が最初のデータキャプチャエッジです。

注： このビットは、通信中には変更しないでください。

このビットは、CRC が TI モードで適用される場合を除き、I²S モードおよび SPI TI モードでは使用されません。

39.9.2 SPI 制御レジスタ 2（SPIx_CR2）

アドレス･オフセット：0x04

リセット値：0x0700

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LDMA_TX	LDMA_RX	FRXTH	DS[3:0]				TXEIE	RXNEIE	ERRIE	FRF	NSSP	SSOE	TXDMAEN	RXDMAEN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **LDMA_TX**：送信の最後の DMA 転送

このビットは、DMA で送信するデータの合計数が奇数であるか偶数であるかを定義するために、データパッキングモードで使用されます。これは、SPIx_CR2 レジスタの TXDMAEN ビットがセットされている場合で、パッキングモードが使用されている場合にのみ意味を持ちます（データ長 =< 8 ビット、SPIx_DR への書込みアクセスは 16 ビット幅）。SPI が無効化された場合に書き込む必要があります（SPIx_CR1 レジスタで SPE = 0）。

0：転送データ項目の数は偶数です。

1：転送データ項目の数は奇数です。

注： CRCEN ビットがセットされている場合、1714 ページの SPI を無効にする手順を参照してください。

このビットは I²S モードでは使用しません。

ビット 13 **LDMA_RX**：受信の最後の DMA 転送

このビットは、DMA で受信するデータの合計数が奇数であるか偶数であるかを定義するために、データパッキングモードで使用されます。これは、SPIx_CR2 レジスタの RXDMAEN ビットがセットされている場合で、パッキングモードが使用されている場合にのみ意味を持ちます（データ長 =< 8 ビット、SPIx_DR への書込みアクセスは 16 ビット幅）。SPI が無効化された場合に書き込む必要があります（SPIx_CR1 レジスタで SPE = 0）。

0：転送データ項目の数は偶数です。

1：転送データ項目の数は奇数です。

注： CRCEN ビットがセットされている場合、1714 ページの SPI を無効にする手順を参照してください。

このビットは I²S モードでは使用しません。

ビット 12 **FRXTH**：FIFO 受信閾値

このビットは、RXNE イベントをトリガする RXFIFO の閾値をセットするために使用されます。

0：FIFO レベルが 1/2（16 ビット）以上になると RXNE イベントが生成されます。

1：FIFO レベルが 1/4（8 ビット）以上になると RXNE イベントが生成されます。

注： このビットは I²S モードでは使用しません。

ビット 11:8 **DS[3:0]**：データ･サイズ

以下のビットは SPI 転送のデータ長を設定します。

0000：未使用

0001：未使用

0010：未使用

0011：4 ビット

0100：5 ビット

0101：6 ビット

0110：7 ビット

0111：8 ビット

1000：9 ビット

1001：10 ビット

1010：11 ビット

1011：12 ビット

1100：13 ビット

1101：14 ビット

1110：15 ビット

1111：16 ビット

ソフトウェアが「未使用」値のいずれかの書込みを試みた場合、値は強制的に「0111」になります（8 ビット）

注： これらのビットは I^2S モードでは使用されません。

ビット 7 **TXEIE：**Tx バッファ･エンプティ割込みイネーブル

0：TXE 割込みはマスクされます。

1：TXE 割込みはマスクされません。TXE フラグがセットされたとき、割込みリクエストの生成に使用されます。

ビット 6 **RXNEIE：**RX バッファ･ノット･エンプティ割込みイネーブル

0：RXNE 割込みはマスクされます。

1：RXNE 割込みはマスクされません。RXNE フラグがセットされたとき、割込みリクエストの生成に使用されます。

ビット 5 **ERRIE：**エラー割込み有効化

このビットは、エラー状態が発生したとき（SPI モードでは CRCERR、OVR、MODF；TI モードでは FRE；I^2S モードでは UDR、OVR、FRE）、割込みの生成を制御します。

0：エラー割込みはマスクされます。

1：エラー割込みは有効です。

ビット 4 **FRF**：フレーム･フォーマット

0：SPI モトローラ･モード

1：SPI TI モード

注： このビットは、SPI が無効（SPE=0）のときにのみ書き込む必要があります。

このビットは I^2S モードでは使用しません。

ビット 3 **NSSP**：NSS パルス管理

このビットは、マスタモードでのみ使用されます。これにより、SPI は連続転送中に 2 つの連続したデータ間で NSS パルスを生成できます。単一のデータ転送の場合、転送後、NSS ピンは強制的にハイレベルになります。

CPHA = '1' の場合も FRF = '1' の場合も意味を持ちません。

0：NSS パルスなし

1：NSS パルスを生成

注： 1.このビットは、SPI が無効（SPE=0）のときにのみ書き込む必要があります。

2.このビットは I^2S モードおよび SPI TI モードでは使用しません。

ビット 2 **SSOE**：SS 出力イネーブル

0：マスタ･モードで SS 出力は無効にされ、SPI インタフェースをマルチマスタ設定で使用できます。

1：SPI インタフェースが有効であるとき、マスタ･モードで SS 出力は有効です。SPI インタフェースはマルチマスタ環境では機能できません。

注： このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 1 **TXDMAEN**：Txバッファ DMA イネーブル

このビットがセットされると、TXE フラグがセットされるたびに DMA リクエストが生成されます。

0：Tx バッファ DMA は無効です。

1：Tx バッファ DMA は有効です。

ビット 0 **RXDMAEN**：Rxバッファ DMA イネーブル

このビットがセットされると、RXNE フラグがセットされるたびに DMA リクエストが生成されます。

0：Rx バッファ DMA は無効です。

1：Rx バッファ DMA は有効です。

39.9.3 SPI ステータスレジスタ（SPIx_SR）

アドレスオフセット：0x08

リセット値：0x0002

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	FTLVL[1:0]		FRLVL[1:0]		FRE	BSY	OVR	MODF	CRCERR	UDR	CHSIDE	TXE	RXNE
			r	r	r	r	r	r	r	r	rc_w0	r	r	r	r

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:11 **FTLVL[1:0]**：FIFO 送信レベル

これらのビットは、ハードウェアによってセット／クリアされます。

00：FIFO エンプティ

01：1/4 FIFO

10：1/2 FIFO

11：FIFO フル（FIFO 閾値が 1/2 より大きい場合に FULL とみなします）

注： このビットは I²S モードでは使用しません。

ビット 10:9 **FRLVL[1:0]**：FIFO 受信レベル

これらのビットは、ハードウェアによってセット／クリアされます。

00：FIFO エンプティ

01：1/4 FIFO

10：1/2 FIFO

11：FIFO フル

注： これらのビットは、CRC 計算が有効なときは I²S モードや SPI 受信専用モードでは使用されません。

ビット 8 **FRE**：フレーム･フォーマット･エラー

このフラグは、TI スレーブモードと I²S スレーブモードの SPI で使用されます。セクション 39.5.11：SPI エラーフラグおよびセクション 39.7.8：I2S エラーフラグを参照してください。

このフラグは、ハードウェアによってセットされ、SPIx_SR がソフトウェアによって読み出されるとリセットされます。

0：フレーム･フォーマット･エラーはありません。

1：フレーム･フォーマット･エラーが発生しました。

ビット 7 **BSY**：ビジー･フラグ

0：SPI（または I2S）はビジー状態ではありません。

1：SPI（または I2S）が通信ビジー状態であるか、または Tx バッファが空ではありません。

このフラグはハードウェアによってセット／クリアされます。

注： BSY フラグを使用する際は注意が必要です。セクション 39.5.10：SPI ステータスフラグおよび 1714 ページのSPI を無効にする手順を参照してください。

ビット 6 **OVR**：オーバーラン･フラグ

0：オーバーランは発生していません。

1：オーバーランが発生しました。

このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、1746 ページのI2S エラーフラグを参照してください。

ビット 5 **MODF**：モード･フォールト

0：モード･フォールトは発生していません。

1：モード･フォールトが発生しました。

このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、セクション ：1724 ページのモードフォールト（MODF）を参照してください。

注： このビットは I^2S モードでは使用しません。

ビット 4 **CRCERR**：CRC エラー･フラグ

0：受信した CRC 値が SPIx_RXCRCR の値と一致しています。

1：受信した CRC 値が SPIx_RXCRCR の値と一致していません。

注： このフラグは、ハードウェアによってセットされ、ソフトウェアによって 0 を書き込むことでクリアされます。

このビットは I^2S モードでは使用しません。

ビット 3 **UDR**：アンダーラン･フラグ

0：アンダーランは発生していません。

1：アンダーランが発生しました。

このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、1746 ページのI2S エラーフラグを参照してください。

注： このビットは SPI モードでは使用しません。

ビット 2 **CHSIDE**：チャネル･サイド

0：左チャネルで送信する必要があるか、または受信が行われました。

1：右チャネルで送信する必要があるか、または受信が行われました。

注： このビットは SPI モードでは使用しません。PCM モードでは意味を持ちません。

ビット 1 **TXE**：送信バッファ･エンプティ

0：Tx バッファは空ではありません。

1：Tx バッファは空です。

ビット 0 **RXNE**：受信バッファ･ノット･エンプティ

0：Rx バッファは空です。

1：Rx バッファは空ではありません。

39.9.4 SPI データレジスタ（SPIx_DR）

アドレスオフセット：0x0C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **DR[15:0]**：データ･レジスタ

受信したデータまたは送信されるデータ

このデータレジスタは、Rx および Tx FIFO 間のインタフェースとして使用できます。データレジスタが読み出されると、データレジスタへの書込みが TxFIFO にアクセスしている間に、RxFIFO にアクセスされます（セクション 39.5.9：データの送受信手順を参照）。

注： データは常に右詰めです。未使用のビットは、レジスタへの書込み時に無視され、レジスタの読出し時にゼロとして読み出されます。Rx 閾値設定は、常に現在使用中の読出しアクセスに対応している必要があります。

39.9.5 SPI CRC 多項式レジスタ（SPIx_CRCPR）

アドレスオフセット：0x10

リセット値：0x0007

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRCPOLY[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:0 **CRCPOLY[15:0]**：CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を格納します。

CRC 多項式（0x0007）は、このレジスタのリセット値です。必要に応じて、別の多項式を設定することができます。

注： 多項式の値は必ず奇数でなければなりません。偶数の値はサポートされていません。

39.9.6 SPI Rx CRC レジスタ（SPIx_RXCRCR）

アドレスオフセット：0x14

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCRC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15:0 **RXCRC[15:0]**：Rx CRC レジスタ

CRC 計算が有効なとき、RXCRC[15:0] ビットには、その後に受信したバイトから算出された CRC 値が格納されています。このレジスタは、SPIx_CR1 レジスタの CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPIx_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

CRC フレームフォーマットが 8 ビット長に設定された場合（SPIx_CR1 の CRCL ビットがクリアされている）、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビット CRC フレームフォーマットが選択された場合（SPIx_CR1 レジスタの CRCL ビットがセットされている）、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

注： BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。

これらのビットは I^2S モードでは使用されません。

39.9.7 SPI Tx CRC レジスタ（SPIx_TXCRCR）

アドレスオフセット：0x18

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXCRC[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15:0 **TXCRC[15:0]**：Tx CRC レジスタ

CRC 計算が有効なとき、TXCRC[7:0] ビットには、その後に送信されたバイトから算出された CRC 値が格納されます。このレジスタは、SPIx_CR1 の CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPIx_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

CRC フレームフォーマットが 8 ビット長に設定された場合（SPIx_CR1 の CRCL ビットがクリアされている）、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビット CRC フレームフォーマットが選択された場合（SPIx_CR1 レジスタの CRCL ビットがセットされている）、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

注： BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。

これらのビットは I^2S モードでは使用されません。

39.9.8 SPIx_I2S 設定レジスタ（SPIx_I2SCFGR）

アドレス･オフセット：0x1C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	ASTRTEN	I2SMOD	I2SE	I2SCFG[1:0]		PCMSYNC	Res.	I2SSTD [1:0]		CKPOL	DATLEN[1:0]		CHLEN
			rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **ASTRTEN**：非同期開始イネーブル

0：非同期開始は無効です。

I^2S がスレーブモードで有効な場合、ハードウェアは I^2S クロックを受信し、WS 信号で適切な遷移を検出すると、転送を開始します。

1：非同期開始は有効です。

I^2S がスレーブモードで有効な場合、ハードウェアは I^2S クロックを受信し、WS 信号で適切なレベルを検出すると、転送を開始します。

注： WS 信号での適切な遷移は、I^2S フィリップス規格が使用されている場合は立ち下がりエッジ、それ以外の規格の場合は立ち上がりエッジです。

WS 信号での適切なレベルは、I^2S フィリップス規格が使用されている場合はローレベル、それ以外の規格の場合はハイレベルです。

詳細については、セクション 39.7.3：起動に関する説明を参照してください。

ビット 11 **I2SMOD**：I^2S モード選択

0：SPI モードが選択されます。

1：I^2S モードが選択されます。

注： このビットは SPI が無効なときに設定してください。

ビット 10 **I2SE**：I^2S 有効化

0：I^2S ペリフェラルは無効です。

1：I^2S ペリフェラルは有効です。

注： このビットは SPI モードでは使用しません。

ビット 9:8 **I2SCFG[1:0]**：I2S 設定モード

00：スレーブ - 送信

01：スレーブ - 受信

10：マスタ - 送信

11：マスタ - 受信

注： これらのビットは、I^2S が無効なときに設定してください。

これらは SPI モードでは使用しません。

ビット 7 **PCMSYNC**：PCM フレーム同期

0：ショート･フレーム同期

1：ロング･フレーム同期

注： このビットは、I2SSTD = 11（PCM 規格使用）の場合にのみ意味を持ちます。

SPI モードでは使用しません。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **I2SSTD[1:0]**：I2S 規格選択
00：フィリップス I^2S 規格
01：MSB 詰め規格（左詰め）
10：LSB 詰め規格（右詰め）
11：PCM 規格
I^2S 規格に関する詳細は、1730ページのセクション 39.7.2 を参照してください。
注： 正しい動作のためには、これらのビットは、I^2S が無効のときに設定してください。
これらは SPI モードでは使用しません。

ビット 3 **CKPOL**：インアクティブ状態のクロック極性
0：I^2S クロックのインアクティブ状態はロー･レベルです。
1：I^2S クロックのインアクティブ状態はハイ･レベルです。
注： 正しい動作のためには、このビットは、I^2S が無効のときに設定してください。
SPI モードでは使用しません。
CKPOL ビットは、SD 信号および WS 信号を受信または送信するために使用される CK エッジ感度には影響を与えません。

ビット 2:1 **DATLEN[1:0]**：転送されるデータ長
00：16 ビットデータ長
01：24 ビットデータ長
10：32 ビットデータ長
11：設定禁止
注： 正しい動作のためには、これらのビットは、I^2S が無効のときに設定してください。
これらは SPI モードでは使用しません。

ビット 0 **CHLEN**：チャネル長（オーディオ･チャネルごとのビット数）
0：16 ビット幅
1：32 ビット幅
ビット書込み動作は、DATLEN=00 のときにのみ意味を持ちます。そうでない場合、書き込まれた値とは無関係に、チャネル長はハードウェアによって 32 ビットに固定されます。
注： 正しい動作のためには、このビットは、I^2S が無効のときに設定してください。
SPI モードでは使用しません。

39.9.9 SPIx_I2S プリスケーラレジスタ（SPIx_I2SPR）

アドレス･オフセット：0x20

リセット値：0x0002

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	MCKOE	ODD	I2SDIV[7:0]							
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **MCKOE**：マスタ･クロック出力イネーブル
0：マスタクロック出力は無効です。
1：マスタクロック出力は有効です。
注： このビットは I^2S が無効なときに設定してください。このビットは、I^2S がマスタモードのときにのみ使用されます。
SPI モードでは使用しません。

ビット 8 **ODD**：プリスケーラの奇数分周比

0：真の分周値 = I2SDIV *2

1：真の分周値 = （I2SDIV * 2）+ 1

1737ページのセクション 39.7.3 を参照してください。

注： このビットは I²S が無効なときに設定してください。このビットは、I²S がマスタモードのときにのみ使用されます。

SPI モードでは使用しません。

ビット 7:0 **I2SDIV[7:0]**：I²S リニアプリスケーラ

I2SDIV [7:0] = 0 または I2SDIV [7:0] = 1 は禁止されている値です。

1737ページのセクション 39.7.3 を参照してください。

注： これらのビットは、I²S が無効なときに設定してください。これらのビットは、I²S がマスタモードのときにのみ使用されます。

これらは SPI モードでは使用しません。

39.9.10 SPI/I2S レジスタマップ

表 364 に、SPI/I2S レジスタマップとリセット値を示します。

表 364. SPI/I2S レジスタマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**SPIx_CR1**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BIDIMODE	BIDIOE	CRCEN	CRCNEXT	CRCL	RXONLY	SSM	SSI	LSBFIRST	SPE	BR [2:0]			MSTR	CPOL	CPHA
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**SPIx_CR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LDMA_TX	LDMA_RX	FRXTH	DS[3:0]				TXEIE	RXNEIE	ERRIE	FRF	NSSP	*SSOE*	TXDMAEN	RXDMAEN
	リセット値																		0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0x08	**SPIx_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FTLVL[1:0]		FRLVL[1:0]		FRE	BSY	OVR	MODF	CRCERR	UDR	CHSIDE	TXE	RXNE
	リセット値																				0	0	0	0	0	0	0	0	0	0	0	1	0
0x0C	**SPIx_DR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DR[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**SPIx_CRCPR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CRCPOLY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x14	**SPIx_RXCRCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXCRC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**SPIx_TXCRCR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXCRC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**SPIx_I2SCFGR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ASTRTEN	I2SMOD	I2SE	I2SCFG[1:0]		PCMSYNC	Res.	I2SSTD		CKPOL	DATLEN[1:0]		CHLEN
	リセット値																				0	0	0	0	0	0		0	0	0	0	0	0
0x20	**SPIx_I2SPR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	MCKOE	ODD	I2SDIV[7:0]							
	リセット値																							0	0	0	0	0	0	0	0	1	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

40 シリアル･オーディオ･インタフェース（SAI）

40.1 概要

この SAI インタフェース（シリアル･オーディオ･インタフェース）は、柔軟性があり幅広い構成が可能なため、多様なオーディオ･プロトコルに対応できます。多くのステレオまたはモノラルのオーディオ･アプリケーションが対象になります。例えば、I^2S 規格、LSB/MSB 詰め、PCM/DSP、TDM、AC'97 の各プロトコルに対応することができます。オーディオブロックがトランスミッタとして設定された場合、SPDIF 出力が提供されます。

このレベルの柔軟性と再設定機能を実現するため、SAI は独立した 2 つのオーディオサブブロックを内蔵しています。各ブロックには専用のクロックジェネレータと I/O ラインコントローラが備わっています。

この SAI は、マスタまたはスレーブの設定で動作します。このオーディオサブブロックはレシーバにもトランスミッタにもなり、他の一方に対して同期または非同期で動作します。

40.2 SAI の主な機能

- 個別に FIFO を内蔵し、トランスミッタまたはレシーバとなることができる 2 つの独立したオーディオサブブロック。
- 各オーディオサブブロックに 8 ワード FIFO を内蔵。
- オーディオサブブロック間での同期モードまたは非同期モードが可能。
- 両オーディオサブブロックに対して独立してマスタ設定またはスレーブ設定が可能。
- 両オーディオサブブロックをマスタモードに設定した場合に、個別のオーディオサンプリング周波数に対応する各オーディオブロック用クロックジェネレータ。
- 設定可能なデータサイズ：8、10、16、20、24、32 ビット
- オーディオプロトコル：I^2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97
- 最大 4 台のマイクロフォンのペアをサポートする PDM インタフェース。
- 必要に応じて、SPDIF 出力を利用可能。
- サイズ設定が可能な最大 16 個のスロットが利用可能。
- フレームごとのビット数が設定可能。
- フレーム同期アクティブレベルが設定可能（オフセット、ビット長、レベル）。
- スロット内の先頭アクティブビット位置が設定可能。
- データ転送における LSB ファーストまたは MSB ファーストが可能。
- ミュートモード。
- ステレオ／モノラルオーディオフレーム機能。
- 通信クロックストローブエッジが設定可能（SCK）。
- 対応する割込み付きのエラーフラグ（それぞれイネーブル付き）。
  - オーバーラン検出およびアンダーラン検出、
  - スレーブモードにおける予測フレーム同期信号検出、
  - スレーブモードにおけるレイトフレーム同期信号検出、
  - 受信時 AC'97 モードに対するコーデックノットレディ

- 割込みソース（有効時）:
  - エラー
  - FIFO リクエスト
- 2 チャネル DMA インタフェース

40.3 SAI の実装

表 365. STM32G4 シリーズ SAI の機能 (1)

SAI の機能	SAI1
I^2S、LSB/MSB 詰め、PCM/DSP、TDM、AC'97	X
FIFO サイズ	8 ワード
SPDIF	X
PDM	X(2)

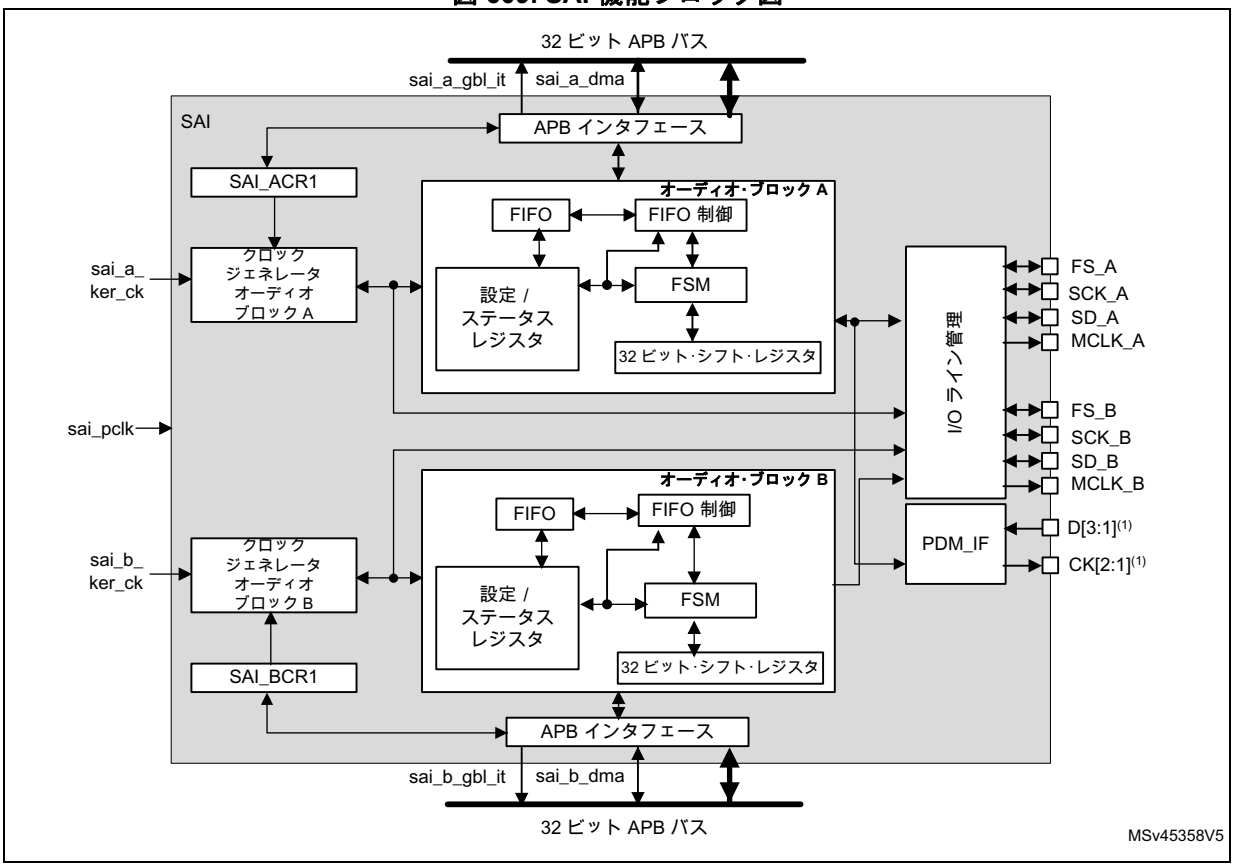
1. “X” はサポートされています。“-” はサポートされていません。
2. D[3:1] および CK[2:1] の信号のみ使用可能です。

40.4 SAI の機能詳細

40.4.1 SAI ブロック図

図 609 に SAI ブロック図を、表 366 および表 367 に SAI 内部および外部信号のリストを示します。

図 609. SAI 機能ブロック図



1. これらの信号は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3 : SAI の実装を参照してください。

この SAI は大部分が 2つのオーディオサブブロックから構成されており、それぞれに固有のクロックジェネレータを装備しています。各オーディオブロックは、それぞれの機能ステートマシンによって制御される 32 ビットシフトレジスタを内蔵しています。データは、専用 FIFO に保存されたり、そこから読み出されたりします。FIFO は CPU からアクセスすることもできれば、通信時に CPU を開放するため DMA からアクセスすることもできます。各オーディオブロックは独立しています。互いに同期することもできます。

I/O ラインコントローラは、SAI 内の指定オーディオブロックに対する 4 本の専用ピン（SD、SCK、FS、MCLK）を管理します。この 2 つのサブブロックを同期するように指定すると、いくつかのピンを共用していくつかのピンを開放し、汎用 I/O として使用することができます。アプリケーション、デコーダ要件、オーディオブロックがマスタとして設定されているか否かによって、MCLK ピンを出力にしたり、それ以外にしたりすることができます。

1 つの SAI が別の SAI と同期して動作するように設定されている場合、より多くの I/O を開放することができます（SD_x ピンを除く）。

この機能ステートマシンは、広範囲なオーディオプロトコルを処理するように設定することができます。所望のプロトコルを設定するため、いくつかのレジスタがあります（オーディオフレーム波形ジェネレータ）。

オーディオサブブロックは、マスタモードまたはスレーブモードでトランスミッタまたはレシーバになることができます。マスタモードでは、SCK_x ビットクロックとフレーム同期信号が SAI で生成されますが、スレーブモードでは、これらの信号を別の外部マスタまたは内部マスタから受け取ります。FS 信号の方向がマスタモードまたはスレーブモードの定義に直接結びつかない特別なケースがあります。AC'97 プロトコルでは、SAI （リンク･コントローラ）が SCK クロックを入力するように設定されている場合でも （つまりスレーブ･モードでも）、FS 信号は SAI の出力になります。

注： **読みやすくするために、このセクションでは SAI_A または SAI_B を SAI_x と表記しています。「x」とは SAI A サブブロックまたは SAI B サブブロックのことです。**

40.4.2 SAI ピンおよび内部信号

表 366. SAI 内部入力／出力信号

内部信号名	信号タイプ	説明
sai_a_gbl_it/ sai_b_gbl_it	出力	オーディオブロック A および B グローバル割込み
sai_a_dma、sai_b_dma	入力／出力	オーディオブロック A および B DMA 確認応答およびリクエスト
sai_a_ker_ck/ sai_b_ker_ck	入力	オーディオブロック A/B カーネルクロック
sai_pclk	入力	APB クロック

表 367. SAI の入出力ピン

名前	信号タイプ	コメント
SAI_SCK_A/B	入力／出力	オーディオブロック A/B ビットクロック
SAI_MCLK_A/B	出力	オーディオブロック A/B マスタクロック
SAI_SD_A/B	入力／出力	ブロック A/B のデータライン
SAI_FS_A/B	入力／出力	オーディオブロック A/B のフレーム同期ライン
SAI_CK[4:1]	出力	PDM ビットストリーム･クロック(1)
SAI_D[4:1]	入力	PDM ビットストリーム･データ (1)。

1. これらの信号は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3 : SAI の実装を参照してください。

40.4.3 主要な SAI モード

SAI の各オーディオサブブロックは、選択したオーディオブロックの SAI_xCR1 レジスタの MODE ビットを使ってマスタまたはスレーブに設定することができます。

マスタモード

マスタモードでは、SAI は接続された外部デバイスへタイミング信号を送り出します。

- ビットクロックとフレーム同期はそれぞれ SCK_x ピンと FS_x ピンに出力されます。
- 必要であれば、SAI は MCLK_x ピンにマスタクロックを生成することもできます。

SCK_x、FS_x、および MCLK_x のいずれも出力として設定されます。

スレーブモード

SAI は、外部デバイスからのタイミング信号の受信を待ちます。

- SAI サブブロックが非同期モードに設定されている場合、SCK_x ピンおよび FS_x ピンは入力として設定されます。
- SAI サブブロックが 2 番目のオーディオサブブロックと同期して動作するよう設定されている場合、該当する SCK_x および FS_x ピンは空きとなり汎用 I/O として使用できます。

MCLK_x ピンはスレーブモードでは使用されず、別の機能に割り当てることができます。

スレーブデバイスを有効にした後でマスタを有効にすることをお勧めします。

SAI モードの設定と有効化

各オーディオサブブロックは、該当するオーディオブロックにある SAI_xCR1 レジスタの MODE ビットを使って個別にトランスミッタまたはレシーバに指定することができます。そのため、SAI_SD_x ピンは出力または入力としてそれぞれ設定されます。

同じ SAI にある 2 個のマスタ･オーディオ･ブロックは、2 つの異なる MCLK および SCK クロック周波数で設定することができます。この場合、オーディオブロックは非同期モードに設定する必要があります。

SAI 内の各オーディオブロックは、SAI_xCR1 レジスタの SAIEN ビットにより有効化されます。このビットがアクティブになると直ちに、トランスミッタまたはレシーバはスレーブモードでクロックライン、データライン、同期ラインの動作に反応するようになります。

マスタ TX モードでは、オーディオブロックを有効化すると直ちに、FIFO にデータがなくとも、外部スレーブに対するビットクロックを生成しますが、FS 信号の生成は、FIFO 内のデータの存在が条件となります。FIFO が先頭の送信データを受け取った後、このデータが外部スレーブに出力されます。FIFO に送信データがない場合、オーディオフレームにおいては 0 値が送信されて、アンダーランフラグが生成されます。

スレーブモードでは、オーディオブロックが有効化されて、フレームの開始が検出されたときに、オーディオフレームが開始されます。

スレーブ TX モードでは、オーディオブロックを有効化した後の最初のフレームでアンダーランイベントが生成されることはありません。理由は、次の動作シーケンスが必要となるためです。

1. ソフトウェアまたは DMA により SAI_xDR へ書込みを行います。
2. FIFO 閾値（FLH）フラグが 0b000（FIFO エンプティ）でなくなるまで待ちます。
3. スレーブ送信ミッタモードでオーディオブロックを有効化します。

40.4.4 SAI 同期モード

SAI サブクロック A と B は同期できます。

内部同期

オーディオサブブロックは、同じ SAI 内の 2 番目のオーディオサブブロックと同期して動作できるよう設定できます。この場合、ビットクロック信号とフレーム同期信号が共用され、通信に使用する外部ピン数を少なくすることができます。同期モードに設定されたオーディオブロックでは、そのSCK_x、FS_x、および MCLK_x ピンは GPIO として開放状態に戻されています。非同期モードに設定されたオーディオブロックのほうが、FS_x、SCK_x、MCLK_x の I/O ピンに適したブロックとなります（オーディオブロックをマスタと考えた場合）。

一般に、同期モードのオーディオブロックを使って、SAI を全二重モードに設定することができます。2 個のオーディオブロックの片方をマスタとして設定し、他方をスレーブとして設定することができま

す。あるいは、両方をスレーブとして設定し、1 つは非同期ブロック(SAI_xCR1 の対応する SYNCEN[1:0] ビットを 00 にセット）に、もう一方を同期ブロック（SAI_xCR1 の対応する SYNCEN[1:0] ビットを 01 にセット）にすることもできます。

注： **内部再同期ステージのため、PCLK APB 周波数はビットレートクロック周波数の 2 倍以上である必要があります。**

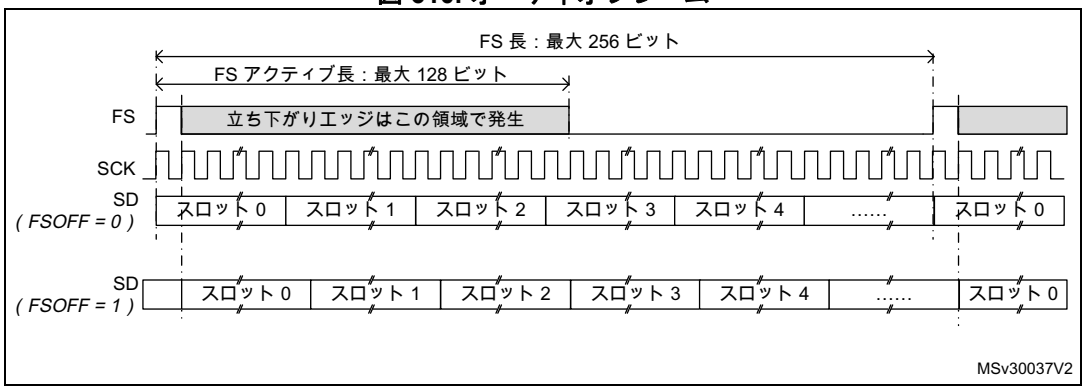
40.4.5 オーディオデータサイズ

オーディオフレームは、SAI_xCR1 レジスタの DS[2:0] ビットを設定することにより、様々なデータサイズを対象とすることができます。データサイズとしては、8、10、16、20、24、32 ビットが可能です。転送時、SAI_xCR1 レジスタの LSBFIRST ビットの設定によってデータの MSB を先頭にして送信することもできれば LSB を先頭にすることもできます。

40.4.6 フレーム同期

FS 信号は、オーディオフレームにおいてフレーム同期信号として機能します（フレーム開始）。この信号の波形は、このフレーム同期動作に関する独自の仕様を持つ様々なオーディオプロトコルに対応するため、全面的に設定可能となっています。この再設定は SAI_xFRCR レジスタを使って実行されます。図 610 にこの柔軟性について図示します。

図 610. オーディオフレーム



AC’97 モードまたは SPDIF モード（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01）では、フレーム同期波形を強制的に AC’97 プロトコルに適合させます。SAI_xFRCR レジスタ値は無視されます。

各オーディオブロックは独立であるため、それぞれに固有の設定が必要となります。

フレーム長

- マスタモード

  オーディオフレーム長は、SAI_xFRCR レジスタの FRL[7:0] ビットを設定して、最大 256 ビットクロックに設定することができます。

  フレーム長がフレームに対して指定されたスロット数より大きい場合、SAI_xCR2 レジスタの TRIS ビットの状態に応じて、残りの送信ビットが 0 に拡張されるか、または SD ラインがハイ･インピーダンス（HI-Z）状態になります（FS 信号の機能を参照）。受信モードでは、残りのビットは無視されます。

  NODIV ビットがクリアされている場合、(FRL + 1) は 8 から 256 のうちの 2 の累乗の数に調整する必要があります。これは、オーディオフレームがビットクロックサイクル当たり整数個の MCLK パルスを含むようにするためです。

NODIV ビットがセットされている場合、(FRL + 1) フィールドは 8 から 256 のうちの任意の値をとることができます。セクション 40.4.8 : SAI クロックジェネレータを参照してください。

- スレーブモード

  オーディオフレーム長は主に、外部マスタから送信されるオーディオフレーム当たりのビットクロック数をスレーブに対して指定するために使用されます。おもに処理中のオーディオ･フレームでマスタからの想定されるフレーム同期信号または遅延したフレーム同期信号の発生を検出するために使用されます。この場合、エラーが生成されます。詳細については、セクション 40.4.14 : エラーフラグを参照してください。

  スレーブモードでは、SAI_xFRCR レジスタの FRL[7:0] ビット設定に制約はありません。

フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオフレーム内で転送する最小ビット数は 8 です。

フレーム同期極性

SAI_xFRCR レジスタの FSPOL ビットにより、フレームが開始される FS ピンのアクティブ極性が設定されます。フレーム開始は、エッジに反応します。

スレーブモードにおいて、オーディオブロックは有効なフレームを待って、送信または受信を開始します。フレーム開始はこの信号に同期します。これは、通信中のフレーム開始が検出されず、予測のフレーム開始と一致しない場合にのみ有効です（セクション 40.4.14 : エラーフラグを参照）。

マスタモードでは、SAI_xCR1 レジスタの SAIEN ビットがクリアされるまで、オーディオフレームが完了するごとにフレーム同期が毎回送信されます。前のオーディオ･フレームの終了時に FIFO にデータがない場合、アンダーラン状態は セクション 40.4.14 : エラーフラグの説明のように扱われますが、オーディオ通信フローが中断されることはありません。

フレーム同期アクティブ･レベル長

SAI_xFRCR レジスタの FSALL[6:0] ビットにより、フレーム同期信号のアクティブ･レベル長が設定されます。この長さは、1 から 128 のビット･クロック･サイクルに設定することができます。

例えば、I^2S、LSB 詰めまたは MSB 詰めモードでは、アクティブ長をフレーム長の 1/2 に、PCM/DSP、TDM では 1 ビット幅に、それぞれ設定することができます。

フレーム同期オフセット

アプリケーションで対象とするオーディオプロトコルに応じて、オーディオフレームの最終ビットまたは先頭ビットを送信したときフレーム同期信号をアサートすることができます（例えば、それぞれ I^2S 規格プロトコルおよび MSB 詰めプロトコルの場合）。SAI_xFRCR レジスタの FSOFF ビットにより、2 つの設定のいずれかを選択することができます。

FS 信号の機能

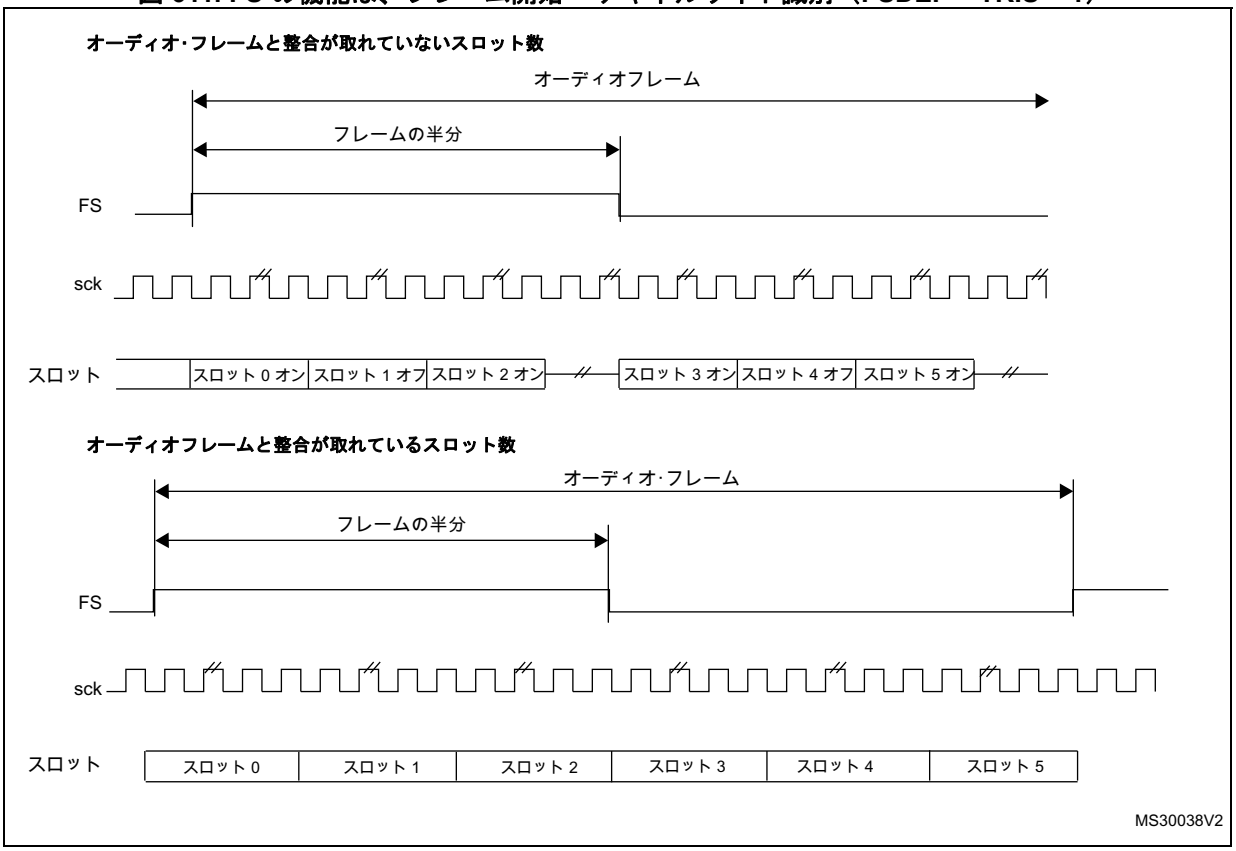
FS 信号は、FS の機能によって意味が異なります。SAI_xFRCR レジスタの FSDEF ビットによりどちらの意味を持たせるかを選択します。

- 0 : 例えば PCM/DSP、TDM、AC'97 オーディオ･プロトコルのようなフレーム開始
- 1 : 例えば I2S、MSB 詰めまたは LSB 詰めプロトコルのような、オーディオフレーム内のチャネルサイド識別およびフレーム開始

FS 信号をフレーム開始およびフレーム内チャネルサイド識別と見なす場合、指定スロット数は半分を左チャネル用、もう半分を右チャネル用と見なす必要があります。ハーフオーディオフレームのビットクロック数がチャネルサイド専用スロット数より大きい場合、TRIS = 0 であれば、SAI_xCR2 レジスタの残りのビットクロックに対して 0 が送信されます。そうしないと、TRIS = 1 の場合、SD

ラインがハイインピーダンスに開放されます。受信モード時、残りのビットクロックサイクルはチャネルサイドが変化するまで無視されます。

図 611. FS の機能は、フレーム開始 + チャネルサイド識別（FSDEF = TRIS = 1）

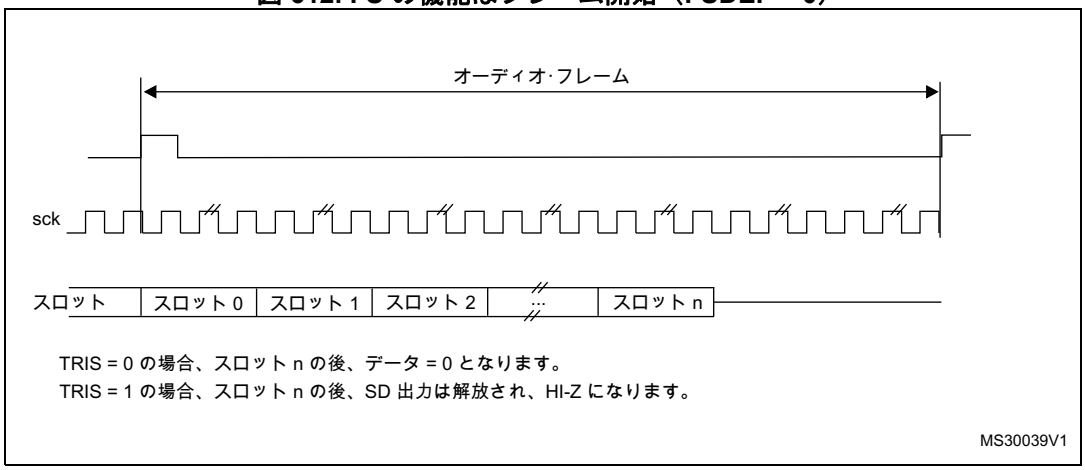


1. フレーム長は偶数である必要があります。

SAI_xFRCR の FSDEF ビットがクリアされたままの場合、FS 信号はフレーム開始に相当し、さらに SAI_xSLOTR の NBSLOT[3:0] ビットで指定されたスロット数に SAI_xSLOTR の SLOTSZ[1:0] ビットで設定されたスロット当たりのビット数を乗算した値が、フレームサイズ（SAI_xFRCR レジスタの FRL[7:0] ビット）より小さい場合、

- SAI_xCR2 レジスタで TRIS = 0 のとき、送信の場合はフレームの終わりまで、最終スロット後の残りのビットは強制的に 0 に設定されます。
- TRIS = 1 のとき、これら残りのビットの転送中、ラインはハイ･インピーダンス（HI-Z）になります。受信モードでは、これらのビットは破棄されます。

図 612. FS の機能はフレーム開始（FSDEF = 0）



送信モードでオーディオブロックが SD ラインの SPDIF 出力を取得するように設定されている場合は、FS 信号は使用されません。対応する FS の I/O は開放され、他の目的のために空けられます。

40.4.7 スロットの設定

スロットは、オーディオフレームの基本要素です。オーディオフレーム内のスロット数は NBSLOT[3:0] + 1 に等しくなります。

オーディオフレーム当たりの最大スロット数は 16 に固定されています。

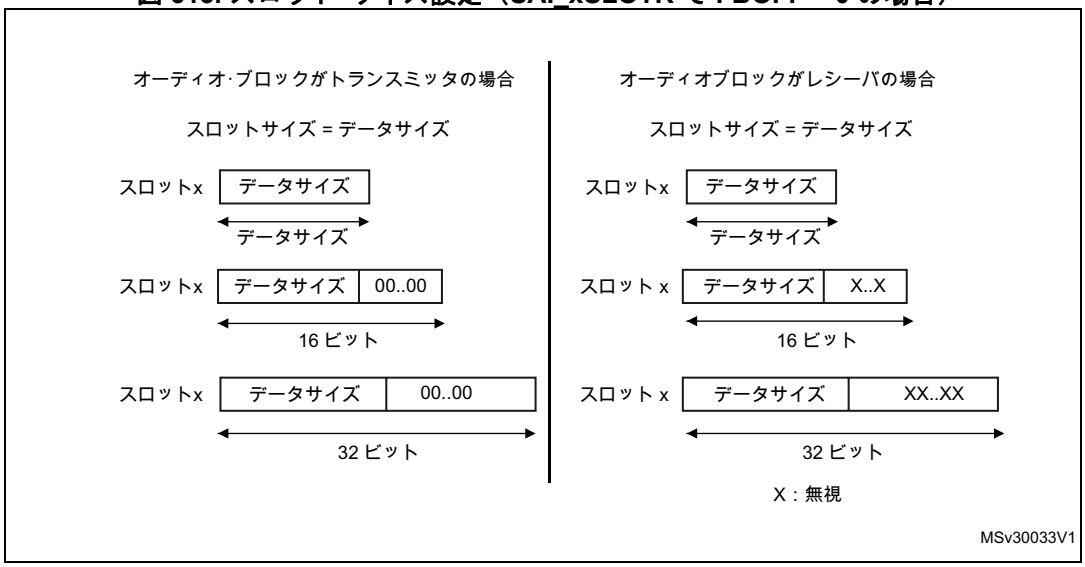
AC'97 プロトコルまたは SPDIF（PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01の場合）に対しては、スロット数は、そのプロトコル仕様を対象とするよう自動的に設定され、NBSLOT[3:0] の値は無視されます。

各スロットは、SAI_xSLOTR レジスタの SLOTEN[15:0] ビットにより有効/無効を指定することができます。

送信モードでは、無効なスロットが転送されると、SD データラインは TRIS ビットの設定に応じて強制的に 0 に設定されるか、または開放されてハイインピーダンス（HI-Z）状態になります（非アクティブ･スロットでの出力データ･ライン管理を参照）。受信モードでは、このスロットの終わりからの受信した値は無視されます。結果として、FIFO アクセスがないため、この非アクティブなスロット状態にリンクした FIFOからの読出しリクエストや FIFO への書込みリクエストは起こりません。

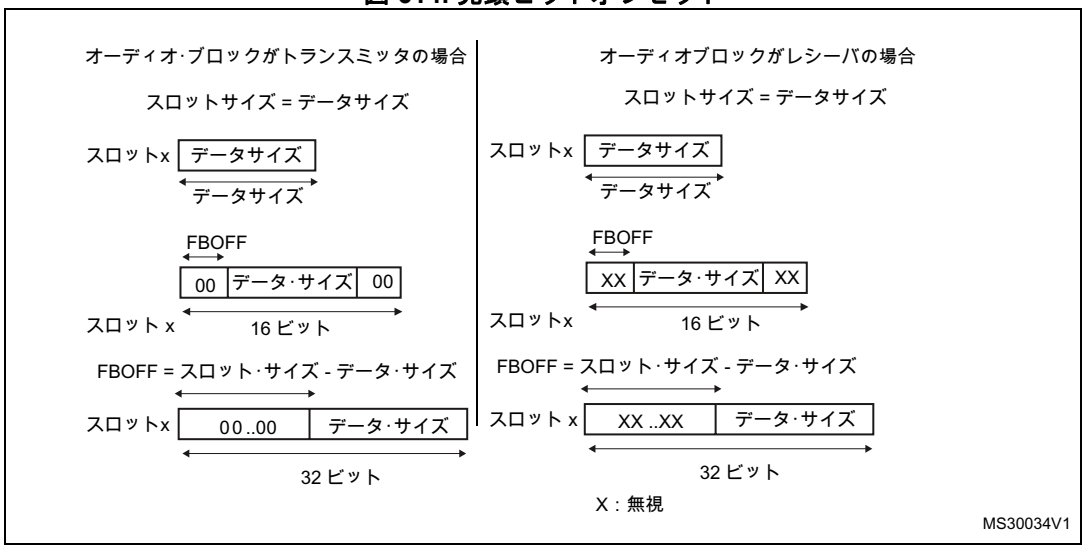
また、スロットのサイズも 図 613 に示すように設定可能です。スロットのサイズは、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビットを設定することで選択できます。サイズは、オーディオフレームの各スロットに同じように適用されます。

図 613. スロット･サイズ設定（SAI_xSLOTR で FBOFF = 0 の場合）



スロット内で転送する先頭データ･ビットの位置を選択することができます。このオフセットは、SAI_xSLOTR レジスタの FBOFF[4:0] ビットによって設定されます。送信モードでは、スロットの開始からこのオフセット位置に到達するまでの間 0 値が挿入されます。受信時、オフセット･フェーズ内のビットは無視されます。この機能は LSB 詰めプロトコルを対象としています （オフセットが、スロットサイズ - データサイズに等しい場合）。

図 614. 先頭ビットオフセット



SAI の誤動作を避けるためには、次の条件を満足する必要があります。

FBOFF ≤(SLOTSZ - DS)、

DS ≤SLOTSZ、

NBSLOT x SLOTSZ ≤FRL（フレーム長）、

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合でも、スロット数は偶数でなければなりません。

AC’97 および SPDIF プロトコル（PRTCFG[1:0] ビット = 10 または PRTCFG[1:0] ビット = 01）では、スロットサイズは、セクション 40.4.11 : AC’97 リンク･コントローラで示されるように自動的に設定されます。

40.4.8 SAI クロックジェネレータ

各オーディオブロックには専用のクロックジェネレータが備わっています。クロックジェネレータは、sai_x_ker_ck からマスタクロック（MCLK_x）およびビットクロック（SCK_x）信号を作成します。sai_x_ker_ck クロックは、製品のクロックコントローラ（RCC）によって供給されます。

マスタクロック（MCLK_x）の生成

クロックジェネレータは、オーディオブロックがマスタまたはスレーブとして定義されている場合にマスタクロック（MCLK_x）を供給します。マスタクロックは、該当ブロックの SAIEN ビットが 0 にセットされていても、MCKEN ビットを 1 にセットするとすぐに、生成されます。この機能は、オーディオ･ストリームをアクティブにする前に MCLK_x を生成できるため、MCLK_x クロックが外部オーディオ･デバイスのシステム･クロックとして使用される場合に役立ちます。

オーディオ･サンプルを転送する前に MCLK_x 出力にマスタ･クロックを生成するには、ユーザ･アプリケーションで以下のシーケンスに従う必要があります。

1. SAIEN = 0 であることを確認します。
2. MCKDIV[5:0] 分周器を必要な値にプログラムします。
3. MCKEN ビットを 1 にセットします。
4. 後に、アプリケーションは SAI の他の部分を構成し、SAIEN ビットを 1 に設定してオーディオサンプルの転送を開始できます。

MCLK_x 出力に生成されたクロックの乱れを避けるため、次の操作は推奨されません。

- MCKEN = 1 のときに、MCKDIV を変更すること
- SAIEN = 1 の場合に、MCKEN を 0 にセットすること

SAI では、MCLK_x が（SAIEN = 0 の状態で）MCKEN ビットによってオンおよびオフに切り替えられたときに、MCLK_x 出力にスプールがないことを保証します。

表 368 に、MCLK_x の有効化条件を示します。

表 368. MCLK_x の有効化条件

MCLKEN	NODIV	ブロック x の SAIEN	MCLK_x
0	X	0	無効
1			有効
0	1	1	無効
1			有効
X	0		有効

注 : MCLKEN が 1 にセットされている場合、MCLK_x は AC’97 モードで生成することもできます。

ビットクロック（SCK_x）の生成

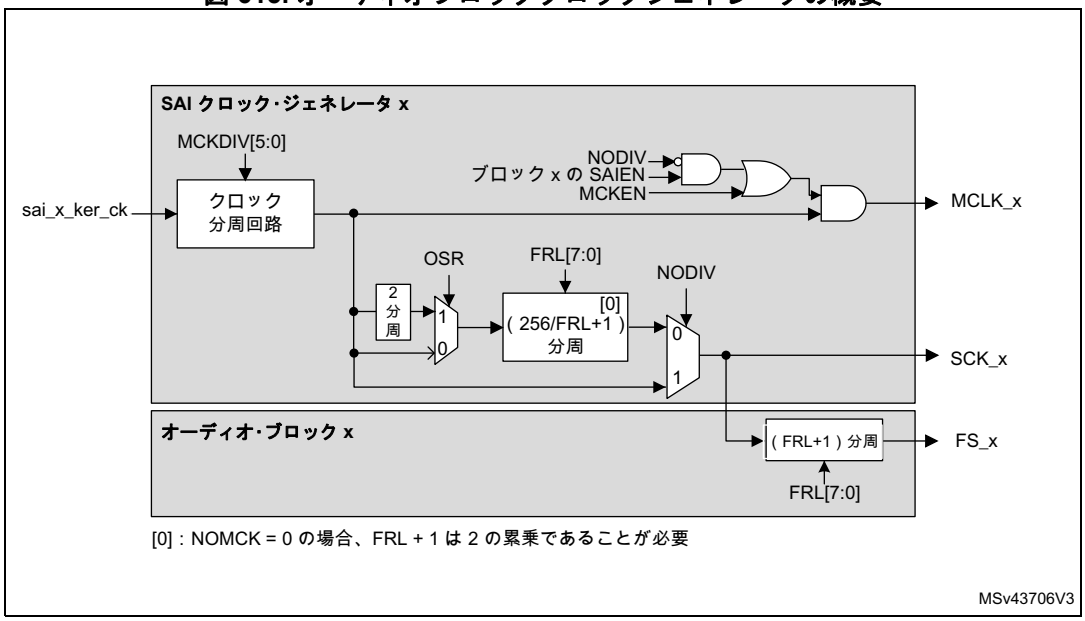
クロックジェネレータは、オーディオブロックがマスタとして定義されている場合にビットクロック（SCK_x）を供給します。フレーム同期（FS_x）は、クロックジェネレータから供給される信号からも得られます。

スレーブモードでは、NODIV および OSR フィールドの値は無視され、SCK_x クロックは生成されません。

SCK_x のビットクロックストローブエッジは、CKSTR フィールドで設定することができ、これはマスタモードとスレーブモードの両方で機能します。

図 615 に、オーディオブロッククロックジェネレータのアーキテクチャを示します。

図 615. オーディオブロッククロックジェネレータの概要



マスタクロック（MCLK_x）とフレーム同期（FS_x）周波数の比率を 256 または 512 に強制するために、NODIV ビットを使用する必要があります。

- NODIV を 0 にセットした場合、フレーム同期とマスタクロックとの周波数比は、OSR 値に従って 512 または 256 に固定されますが、フレーム長は 2 の累乗値でなければなりません。詳細については、以下を参照してください。
- NODIV を 1 にセットした場合、ビットクロック（SCK_x）の周波数は、アプリケーションによって、MCKDIV を介して調整できます。さらに、フレーム長が 8 以上（すなわち、FRL[7:0] > 6）であるかぎり、フレーム長の値に何ら制約はありません。フレーム同期周波数は MCKDIV およびフレーム長（FRL[7:0]）に依存します。この場合、MCLK_x の周波数は SCK_x に等しくなります。

NODIV、MCKEN、SAIEN、OVR、CKSTR、および MCKDIV[5:0] ビットは SAI_xCR1 レジスタに属しますが、FRL[7:0] ビットは SAI_xFRCR レジスタに属します。

NODIV = 0 のときのクロックジェネレータのプログラミング

この場合、MCLK_x 周波数は次のようになります。

- $F_{MCLK_x} = 256 \times F_{FS_x}$（OSR = 0 の場合）
- $F_{MCLK_x} = 512 \times F_{FS_x}$（OSR = 1 の場合）

MCKDIV が 0 ではない場合、MCLK_x 周波数は下の計算式によって与えられます。

$$F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期周波数は次のようになります。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV \times (OSR + 1) \times 256}$$

ビットクロック周波数（SCK_x）は次の計算式で与えられます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck} \times (FRL + 1)}{MCKDIV \times (OSR + 1) \times 256}$$

注： **NODIV が 0 に等しい場合、（FRL+1）は 2 の累乗値でなければなりません。加えて、（FRL+1）は 8～256 でなければなりません。（FRL+1）はオーディオフレーム内のビットクロック数を表します。**

MCKDIV 分周比が奇数である場合には、MCLK のデューティ･サイクルは 50% にはなりません。MCKDIV が奇数、OSR が 0 に等しく、かつ（FRL+1）= 2^8 であると、ビット･クロック信号（SCK_x）についてもデューティ･サイクルが 50% とは異なることがあります。

MCKDIV を偶数か大きな値（10 を超える値）にプログラミングすることを推奨します。

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

NODIV = 1 のときのクロックジェネレータのプログラミング

MCKDIV が 0 ではない場合、ビットクロック（SCK_x）周波数は下の計算式によって与えられます。

$$F_{SCK_x} = F_{MCLK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

フレーム同期（FS_x）周波数は次の計算式によって与えられます。

$$F_{FS_x} = \frac{F_{sai_x_ker_ck}}{(FRL + 1) \times MCKDIV}$$

注： **NODIV が 1 にセットされている場合、（FRL+1）は 8 から 256 のうちの任意の値をとることができます。**

MCKDIV = 0 であると、MCKDIV = 1 のときと同じ結果となることに注意してください。

クロックジェネレータのプログラミング例

表 369 に、48、96、192 kHz に対するプログラミング例を示します。

表 369. クロックジェネレータのプログラミング例

入力 sai_x_ker_ck クロック周波数	MCLK	F_{MCLK}/ F_{FS}	FRL (1)	OSR	NODIV	MCKEN	MCKDIV[5:0]	オーディオ サンプリング周波数 (F_{FS})
98.304 MHz	Y	512	2^N-1	1	0	1	0 または 1	192 kHz
		512	2^N-1	1	0	1	2	96 kHz
		512	2^N-1	1	0	1	4	48 kHz
		256	2^N-1	0	0	1	2	192 kHz
		256	2^N-1	0	0	1	4	96 kHz
		256	2^N-1	0	0	1	8	48 kHz
	N	-	63	-	1	0	8	192 kHz
		-	63	-	1	0	16	96 kHz
		-	63	-	1	0	32	48 kHz

1. N は 3 ~ 8の整数値です。

40.4.9 内部 FIFO

SAI 内の各オーディオブロックには固有の FIFO を搭載しています。そのブロックがトランスミッタに指定されるかレシーバに指定されるかによって、FIFO にそれぞれ書き込んだり読み出したりすることができます。したがって、SAI_xSR レジスタの FREQ ビットに対応した FIFO リクエストは 1 つしかありません。

SAI_xIM レジスタの FREQIE ビットが有効化されている場合、割込みが生成されます。これは次の条件によって変わります：

- FIFO 閾値設定（SAI_xCR2 の FLVL ビット）
- 通信方向（トランスミッタまたはレシーバ）送信モードでの割込み生成および受信モードでの割込み生成を参照してください。

送信モードでの割込み生成

送信モードでの割込みの生成は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDR レジスタに有効なデータがない（SAI_xSR の FLVL[2:0] ビット < 001b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティ（空）でなくなった場合（SAI_xSR の FLVL[2:0] ビット≠0b000）、すなわち FIFO に 1 個以上のデータが格納されている状態になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合、データが FIFO の 1/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 0b010）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 0b010）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 0b010）に設定されている場合、データが FIFO の 1/2 より少ない（SAI_xSR の FLVL[2:0] ビット < 011b）という条件で、

割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 1/2 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧ 011b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合、データが FIFO の 3/4 より少ない（SAI_xSR の FLVL[2:0] ビット < 0b100）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO の 3/4 以上にデータがある状態（SAI_xSR の FLVL[2:0] ビット≧0b100）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフル（FTH[2:0] = 0b100）に設定されている場合、FIFO がフルでない（SAI_xSR の FLVL[2:0] ビット < 101b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフル（SAI_xSR の FLVL[2:0] ビット = 101b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

受信モードでの割込み生成

受信モードでの割込みの発生は、次のように FIFO の設定によって変わります：

- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO エンプティ（FTH[2:0] = 0b000）に設定されている場合、SAI_xDR レジスタに 1 個以上のデータがある（SAI_xSR の FLVL[2:0] ビット ≧ 001b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO がエンプティになる（SAI_xSR の FLVL[2:0] ビット = 0b000）、すなわち FIFO にデータがなくなると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/4（FTH[2:0] = 001b）に設定されている場合､FIFO データロケーションの 1/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット ≧ 0b010）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）｡使用可能な FIFO データロケーションが 1/4 未満（SAI_xSR の FLVL[2:0] ビット < 0b010）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 1/2（FTH[2:0] = 0b010）に設定されている場合、FIFO データロケーションの 1/2 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧011b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）｡使用可能な FIFO データロケーションが 1/2 未満（SAI_xSR の FLVL[2:0] ビット < 011b）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFOがフルの 3/4（FTH[2:0] = 011b）に設定されている場合､FIFO データロケーションの 3/4 以上が使用可能（SAI_xSR の FLVL[2:0] ビット≧0b100）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）｡使用可能な FIFO データロケーションが 3/4 未満（SAI_xSR の FLVL[2:0] ビット < 0b100）になると、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。
- SAI_xCR2 レジスタの FIFO 閾値ビットが FIFO がフル（FTH[2:0] = 0b100）に設定されている場合、FIFO がフル（SAI_xSR の FLVL[2:0] ビット = 101b）という条件で、割込みが生成されます（SAI_xSR レジスタの FREQ ビットがハードウェアによって 1 にセットされます）。FIFO データがフルでなくなると（SAI_xSR の FLVL[2:0] ビット < 101b）、この割込み（SAI_xSR レジスタの FREQ ビット）はハードウェアによってクリアされます。

SAI_xCR1 レジスタの DMAEN ビットがセットされている場合、SAI は、割込み生成と同様に DMA を使用することができます｡FREQ ビットアサーションメカニズムは、上記 FREQIE について説明した割込み生成メカニズムと同じです。

各 FIFO は 8 ワードの FIFO です。FIFO に対する書込み操作や読出し操作は、そのアクセスサイズによらず、1 回につき FIFO の 1 ワード分を対象とします。FIFO の 1 ワードには 1 オーディオスロットが格納されます。FIFO ポインタは、SAI_xDR レジスタに対する各アクセスの後に 1 ワードずつインクリメントされます。

SAI_xDR に書き込む際、データは右詰めである必要があります。

受信データは SAI_xDR で右詰めとなっています。

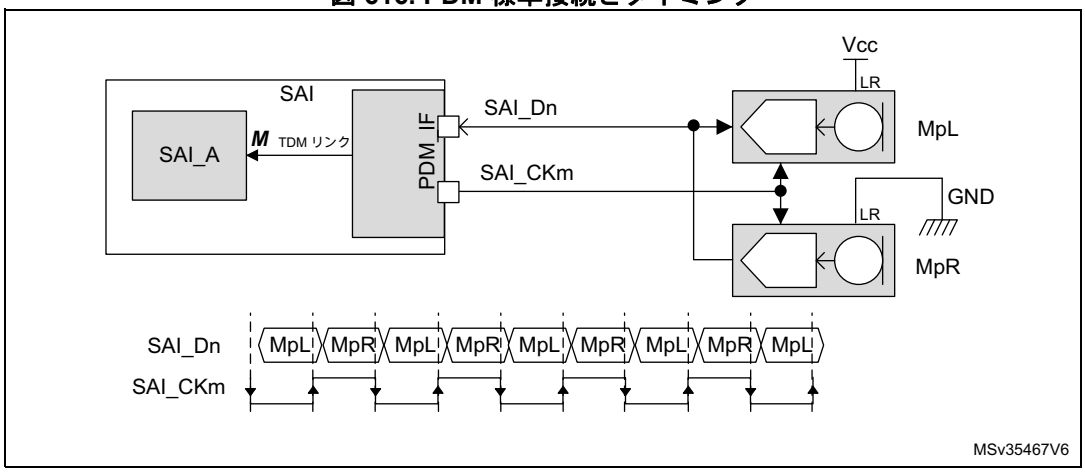
SAI_xCR2 レジスタの FFLUSH ビットをセットして SAI を無効化すると、FIFO ポインタを再初期化することができます。SAI が有効なときに FFLUSH をセットすると、FIFO 内にあるデータは自動的に失われます。

40.4.10 PDM インタフェース

PDM（パルス密度変調）インタフェースは、デジタル･マイクロフォン対応のために提供されています。最大 4 組のデジタルマイクロフォンのペアに並列接続可能です。製品の実装方法によっては、サポートできるマイクロフォンの数が減る場合があります（セクション 40.3：SAI の実装参照）。

図 616 に、PDM インタフェースを介したデジタル･マイクロフォンのペアの標準的な接続を示します。どちらのマイクロフォンも同じビットストリームクロックとデータのラインを共有しています。設定ピン（LR）によって、片方のマイクロフォンが SAI_CK[m] の立ち上がりエッジで有効なデータを供給するのと同時に、もう一方が SAI_CK[m] の立ち下がりエッジで有効なデータを供給します（m はクロックライン数）。

図 616. PDM 標準接続とタイミング



1. **n** はデータライン数、**p** はマイクロフォンのペア数を表します。

PDM 機能は､TDM マスタモードで設定された SAI_A サブクロックとともに使用されるためのものです。SAI_B サブクロックとは使用できません。PDM インタフェースは、SAI_A の TDM インタフェースから供給されるタイミング信号を使用して、それを適合させてビットストリームクロック（SAI_CK[m]）を生成します。

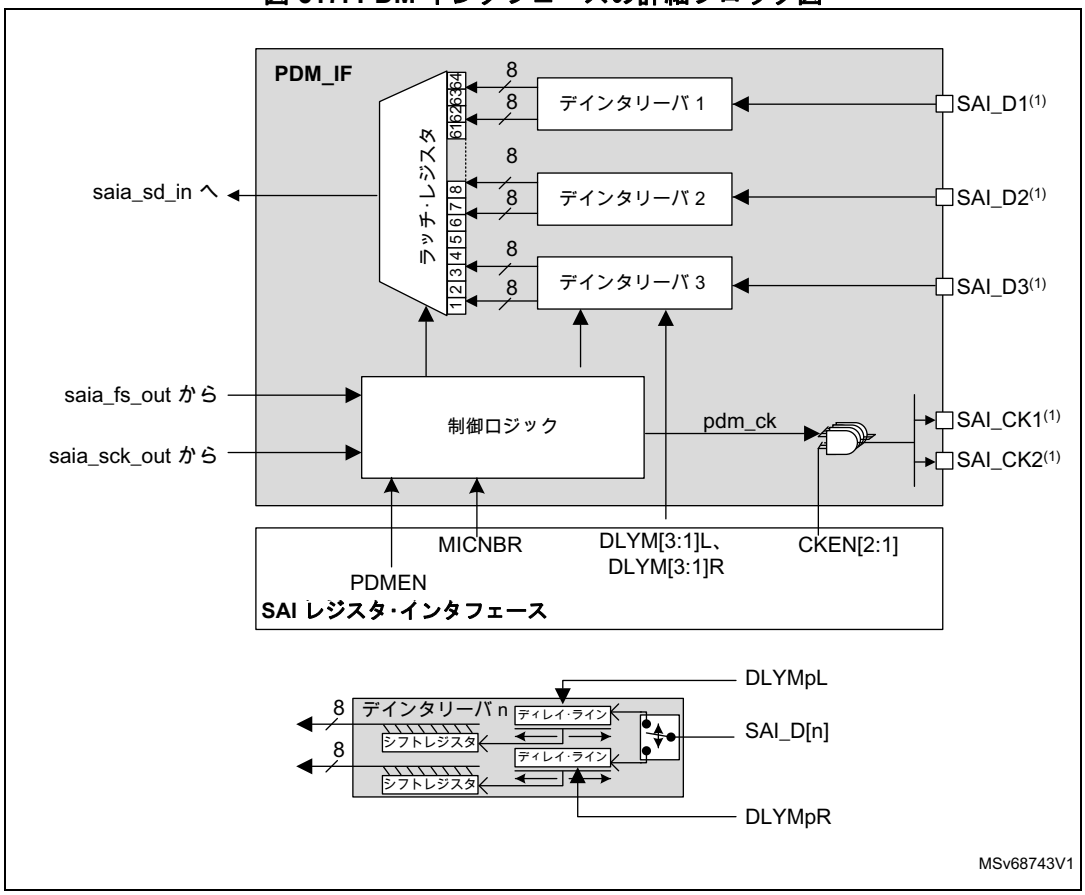
PDM へのデータ処理シーケンスは、次のとおりです。

1. PDM インタフェースは、SAI_A の TDM インタフェースから受信したビットクロックからビットストリームクロックを構築します。
2. マイクロフォンから受信したビットストリームデータ（SAI_D[n]）はインタリーブを解かれ、ビットストリームクロック精度で各マイクロフォンの遅延を微調整するために、7 ビット遅延ラインを通ります。
3. シフトレジスタによって、それぞれのシリアルビットストリームがバイト列に変換されます。
4. 最後の操作では、TDM インタフェースのシリアルデータラインを通じて、結果として得られたバイト列が SAI_A にシフトアウトされます。

以下の図 617 に、PDM インタフェースのブロック図をデインタリーバの詳細図とともに示します。

注： **PDM インタフェースには、ビットストリームから PCM オーディオサンプルを構築するために必要なデシメーションフィルタは組み込まれていません。この操作を行うかどうかは、アプリケーションソフトウェア次第です。**

図 617. PDM インタフェースの詳細ブロック図



1. これらの信号は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3 : SAI の実装を参照してください。
2. **n** はデータ･ライン数、**p** はマイクロフォンのペア数を表します。

PDM インタフェースは、SAI_PDMCR レジスタの PDMEN ビットを通じて有効にできます。ただし、SAI_A ブロックを有効にする前に PDM インタフェースを有効にする必要があります。

メモリのフットプリントを削減するため、アプリケーションが必要とするマイクロフォンの本数を選択できます。この選択は、MICNBR[1:0] ビットで行うことができます。マイクロフォンの本数は、2

本、4 本、6 本、8 本から選択可能です。たとえば、アプリケーションがマイクロホンを 3 本使用するのであれば、4 本を選択する必要があります。

PDM インタフェースの有効化

PDM インタフェースを有効にするには、以下のシーケンスに従います。

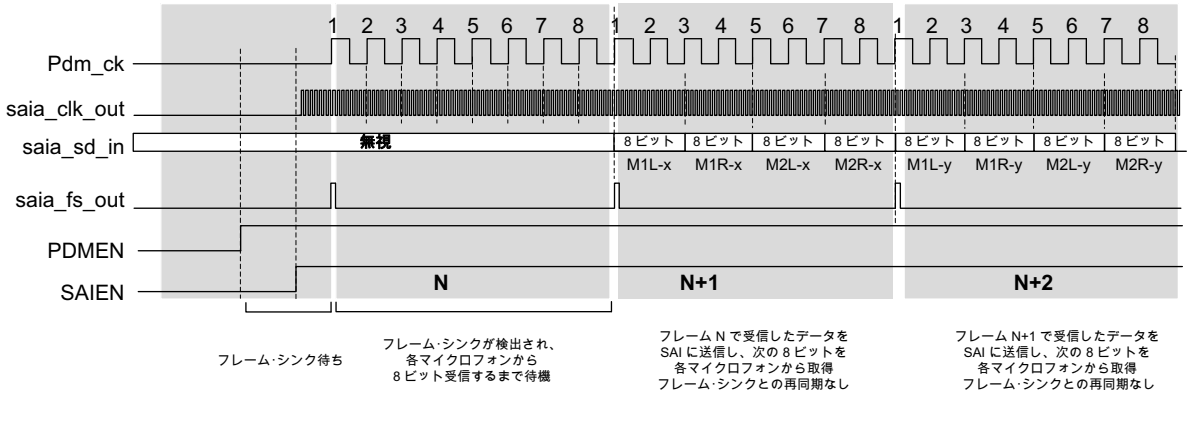
1. SAI_A を TDM マスタモードに設定します（表 370 参照）。
2. 次のように PDM インタフェースを設定します。
   a) MICNBR にデジタルマイクロフォンの本数を設定します。
   b) CKEN の対応するビットを 1 にセットして、アプリケーションで必要なビットストリームクロックを有効にします。
3. PDMEN ビットで PDM インタフェースを有効にします。
4. SAI_A を有効にします。

注： **PDM インタフェースと SAI_A が有効になると、SAI_ADR で受信する最初の 2 つの TDMA フレームは無効になるので、破棄する必要があります。**

起動シーケンス

図 618 に起動シーケンスを示します。PDM インタフェースが有効になると、マイクロフォンサンプルの取得を始める前に、フレーム同期イベントを待ちます。SAI_CK クロック 8 周期後、各マイクロフォンから入力されるデータバイトが利用可能となり、TDM インタフェースを通じて SAI に転送されます。

図 618. 起動シーケンス



SAI_ADR データフォーマット

マイクロフォンから SAI_ADR レジスタに入力されるデータの配列は、次のパラメータに依存しています。

- マイクロフォンの本数
- 選択されたスロット幅
- LSBFIRST ビット

スロット幅によって、SAI_ADR に入力可能な各ワードの有効ビット数が設定されます。

32 ビットのスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 32 ビットの有効なビットが含まれます。これによって、メモリに格納されるワード数が減ります。ただし、対応するには、ソフトウェアが各マイクロフォンのデータのインタリーブを解くための何らかの操作を行う必要があります。

その一方で、8 ビットのスロット幅が選択された場合には、SAI_ADR に入力可能な各データには 8 ビットの有効ビットが含まれます。これによって、メモリに格納されるワード数が増えます。ただし、各ワードにはマイクロフォン 1 本からの情報が含まれているため、これには余計な処理を避けるという利点があります。

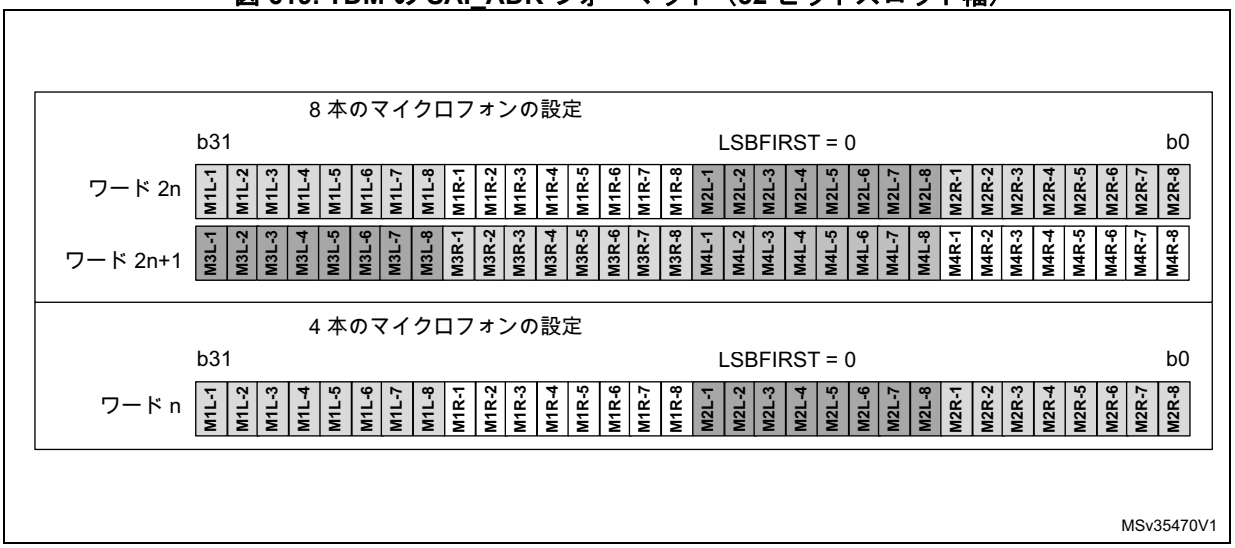
SAI_ADR データフォーマット例

- **32 ビットスロット幅**（DS = 0b111 かつ SLOTSZ = 0）。図 619 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 2 ワードに、各マイクロフォンからの 1 データバイトが含まれています。

  マイクロフォン 4 本の設定では、SAI_ADR レジスタから読み込まれた各ワードに、各マイクロフォンからの 1 データバイトが含まれています。

図 619. TDM の SAI_ADR フォーマット（32 ビットスロット幅）

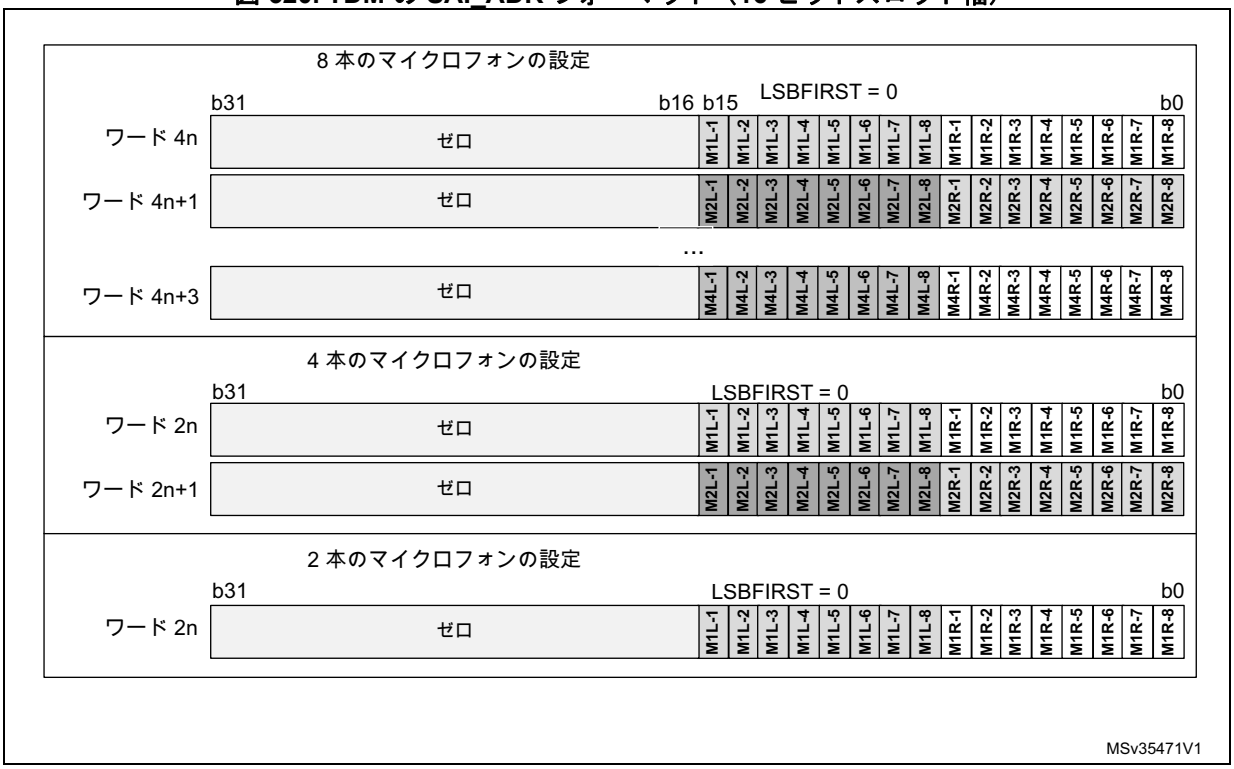


- **16 ビットスロット幅**（DS = 0b100 かつ SLOTSZ = 0）。図 620 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 4 ワードに、各マイクロフォンからの 1 データバイトが含まれています。SAI_ADR の 16 ビットデータは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 16 ビットワードが最大で 2 ワード必要であり、マイクロフォン 2 本からであれば 1 ワード必要です。

図 620. TDM の SAI_ADR フォーマット（16 ビットスロット幅）

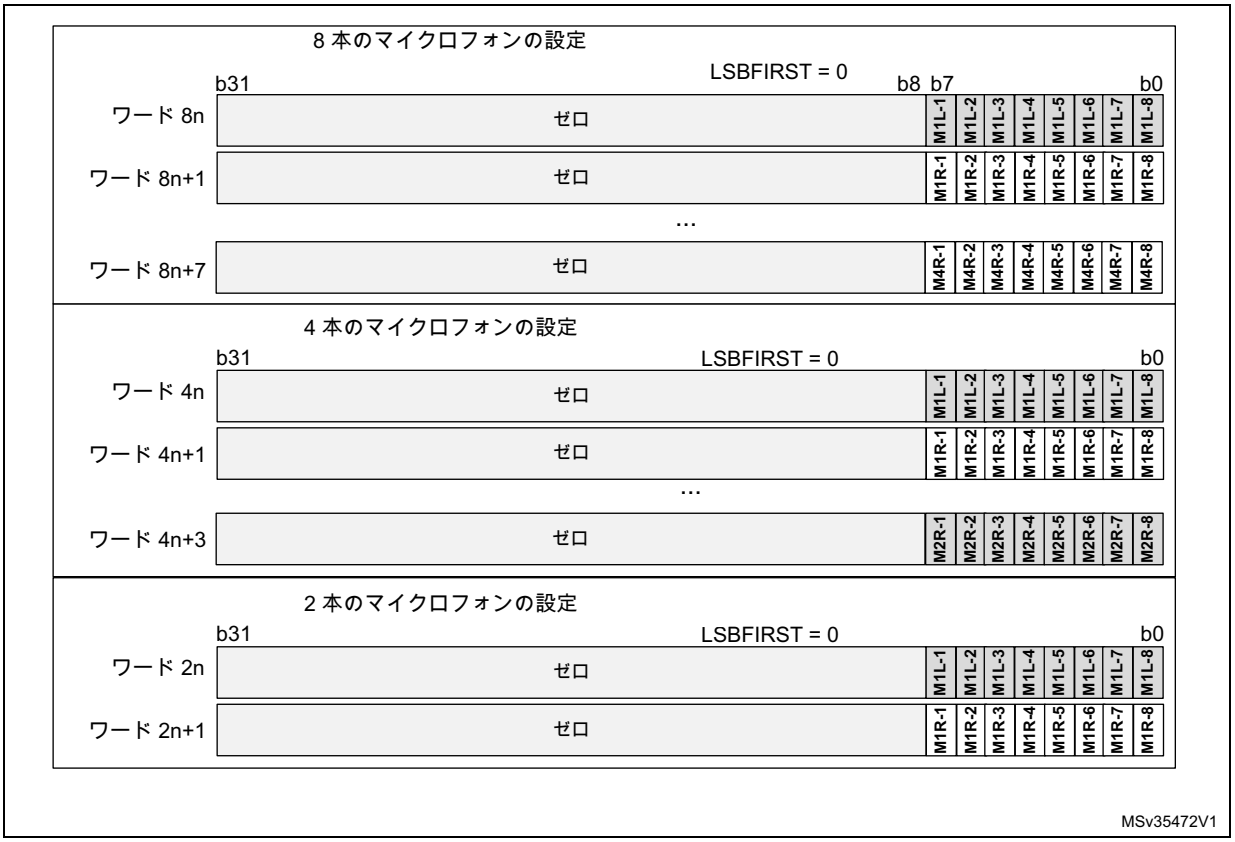


- **8 ビットスロット幅を使用**（DS = 0b010 かつ SLOTSZ = 0）。図 621 を参照してください。

  マイクロフォン 8 本の設定では、SAI_ADR レジスタから読み込まれた連続する 8 ワードに、各マイクロフォンからのデータが 1 バイト含まれています。SAI_ADR の 8 ビットデータは右詰めされていることに注意してください。

  マイクロフォン 4 本または 2 本の設定では、SAI の動作はマイクロフォン 8 本の設定とほぼ同じです。マイクロフォン 4 本からバイトを取得するには 8 ビットワードが最大で 4 ワード必要であり、マイクロフォン 2 本からであれば 2 ワード必要です。

図 621. TDM の SAI_ADR フォーマット（8 ビットスロット幅）



PDM インタフェースの TDM 設定

SAI_A TDM インタフェースは、マイクロフォンのサンプルを取得するために、内部で PDM インタフェースに接続されています。ユーザアプリケーションは、PDM インタフェースを表 370 に示すように設定して、PDM インタフェースとの良好な接続を保証する必要があります。

表 370. TDM 設定

ビット フィールド	値	コメント
MODE	0b01	モードは MASTER レシーバである必要があります。
PRTCFG	0b00	TDM 用フリー･プロトコル
DS	X	フレーム長とスロット数（FRL と NBSLOT）に基づいて、必要なデータフォーマットに応じて調整します。表 371 を参照してください。
LSBFIRST	X	このパラメータは必要なデータフォーマットに応じて使用します。
CKSTR	0	信号の遷移は、SCK_A ビットクロックの立ち上りエッジで発生します。信号はビットクロックの立ち下がりでは安定しています。
MONO	0	ステレオモード
FRL	X	マイクロフォンの本数（MICNBR）に応じて調整します。表 371 を参照してください。
FSALL	0	ハルス幅は 1 ビットクロックサイクルです。
FSDEF	0	FS 信号がフレームの開始です。

表 370. TDM 設定 （続き）

ビット フィールド	値	コメント
FSPOL	1	FS はアクティブハイです。
FSOFF	0	FS はスロット 0 の先頭ビットでアサートされます。
FBOFF	0	スロットにはオフセットはありません。
SLOTSZ	0	スロットサイズ = データサイズ
NBSLOT	X	スロットサイズとフレーム長（FRL と DS）に基づいて、必要なデータフォーマットに応じて調整します。表 371 を参照してください。
SLOTEN	X	NBSLOT に応じて設定します。
NODIV	1	マスタクロック MCLK の生成は不要です。
MCKDIV	X	sai_a_ker_ck 入力に供給される周波数に依存します。 このパラメータは、適切なビットストリーム･クロック周波数を生成するために調整する必要があります。表 371 を参照してください。

ビットストリームクロックレートの調整

SAI TDM インタフェースを適切にプログラムするために､ユーザ･アプリケーションでは 表 370 に記載された設定を考慮し、次のシーケンスに従う必要があります。

1. 次の式を使用して、PDM ビットストリーム･クロックに必要な周波数に応じてビット･クロック周波数（F_{SCK_A}）を調整します。

$$F_{SCK_A} = F_{PDM_CK} \times (MICNBR + 1) \times 2$$

MICNBR は､0､1､2､または 3 の値を取ります（0 = マイクロフォン 2 本､セクション 40.6.17 : 参照）。

2. 次の式を使用して、フレーム長（FRL）を設定します。

$$FRL = (16 \times (MICNBR + 1)) - 1$$

3. スロットサイズ（DS）を（FRL+1）の倍数に設定します。

表 371. TDM フレームの設定例(1)(2)

マイクロフォンのサンプリングレート	マイクロフォンの数	必要な SAI_CKn 周波数 (3)	ビットクロック（SCK_A）周波数	フレームシンク（FS_A）周波数	FRL	DS	NBSLOT	コメント
48 kHz	最大 8	3.072 MHz	24.576 MHz	384 kHz	63	0b111	1	フレームあたり 32 ビット 2 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b100	3	フレームあたり 16 ビット 4 スロット
		3.072 MHz	24.576 MHz	384 kHz	63	0b010	7	フレームあたり 8 ビット 8 スロット
	最大 6	3.072 MHz	18.432 MHz	384 kHz	47	0b110	1	フレームあたり 24 ビット 2 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b100	2	フレームあたり 16 ビット 3 スロット
		3.072 MHz	18.432 MHz	384 kHz	47	0b010	5	フレームあたり 8 ビット 6 スロット
	最大 4	3.072 MHz	12.288 MHz	384 kHz	31	0b111	0	フレームあたり 32 ビット 1 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b100	1	フレームあたり 16 ビット 2 スロット
		3.072 MHz	12.288 MHz	384 kHz	31	0b010	3	フレームあたり 8 ビット 4 スロット
	最大 2	3.072 MHz	6.144 MHz	384 kHz	15	0b100	0	フレームあたり 16 ビット 1 スロット
		3.072 MHz	6.144 MHz	384 kHz	15	0b010	1	フレームあたり 8 ビット 2 スロット
16 kHz	最大 8	1.024 MHz	8.192 MHz	128 kHz	63	0b111	1	フレームあたり 32 ビット 2 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b100	3	フレームあたり 16 ビット 4 スロット
		1.024 MHz	8.192 MHz	128 kHz	63	0b010	7	フレームあたり 8 ビット 8 スロット
	最大 6	1.024 MHz	6.144 MHz	128 kHz	47	0b110	1	フレームあたり 24 ビット 2 スロット
		1.024 MHz	6.144 MHz	128 kHz	47	0b010	5	フレームあたり 8 ビット 6 スロット
	最大 4	1.024 MHz	4.096 MHz	128 kHz	31	0b111	0	フレームあたり 32 ビット 1 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b100	1	フレームあたり 16 ビット 2 スロット
		1.024 MHz	4.096 MHz	128 kHz	31	0b010	3	フレームあたり 8 ビット 4 スロット
	最大 2	1.024 MHz	2.048 MHz	128 kHz	15	0b100	0	フレームあたり 16 ビット 1 スロット
		1.024 MHz	2.048 MHz	128 kHz	15	0b010	1	フレームあたり 8 ビット 2 スロット

1. TDM 設定の詳細については、表 370：TDM 設定を参照してください。SAI に供給される sai_a_ker_ck クロック周波数は、SCK_A 周波数の倍数である必要があり、MCKDIV はそれに応じてプログラムされる必要があります。
2. 上記の sai_a_ker_ck の周波数は一例として示したものにすぎません。使用するデバイスで、これらの周波数を生成できるかどうかを確認するには、リセットおよびクロック制御（RCC）のセクションを参照してください。
3. 上表は、デシメーション比が 64 である場合に許容される設定です。

ディレイラインの調整

PDM インタフェースが有効になると、アプリケーションは、SAI_PDMDLY レジスタを通じて各マイクロフォン入力のディレイセルを実行中に調整できます。

新たなディレイ値は、TDM フレームの 2 周期後に有効になります。

40.4.11 AC’97 リンク･コントローラ

SAI は AC’97 リンクコントローラとしても機能することができます。このプロトコルの仕様は次のとおりです：

- スロット番号とスロットサイズは固定です。
- フレーム同期信号は完全に定義されており波形は固定です。

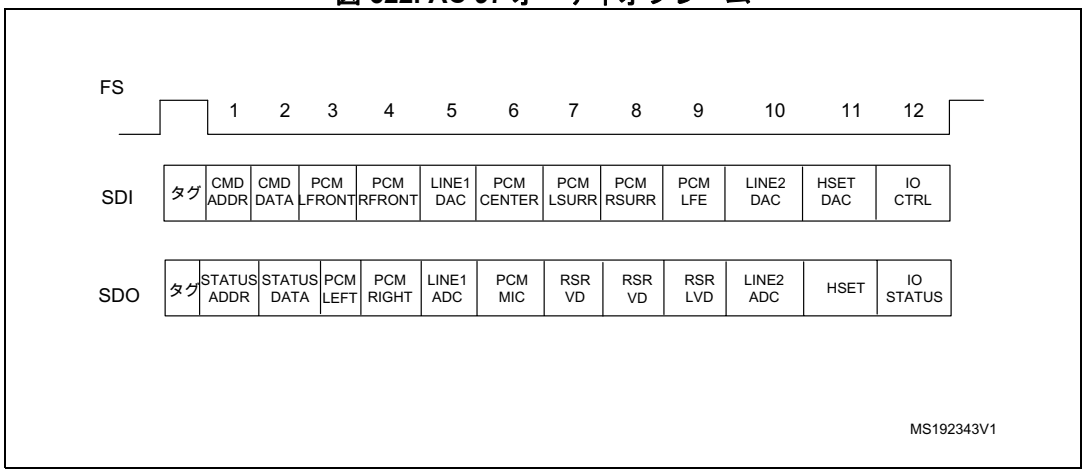
このプロトコルを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットを 10 に設定します。AC’97 モードを選択した場合、使用可能なデータサイズは 16 ビットまたは 20 ビットのみであり、その他の場合、SAI の動作は保証されません。

- したがって、NBSLOT[3:0] ビットと SLOTSZ[1:0] ビットは無視されます。
- スロット数は 13 スロットに固定されています。最初のスロットは 16 ビット幅で、他はすべて 20 ビット幅（データスロット）です。
- SAI_xSLOTR レジスタの FBOFF[4:0] ビットは無視されます。
- SAI_xFRCR レジスタは無視されます。
- MCLK は使用されません。

非同期指定されたブロックからの FS 信号は、自動的に出力として設定されます。これは、マスタ/スレーブの設定にかかわらず AC’97 コントローラリンクが FS 信号を駆動するためです。

図 622 に、AC’97 オーディオフレーム構造を示します。

図 622. AC’97 オーディオフレーム



注： **AC’97 プロトコルでは、タグのビット 2 は予約済みです（常に 0）。したがって、SAI FIFO に書き込まれる値によらず、タグのビット 2 は 0 レベルに固定されます。**

タグ 表示についての詳細は、AC’97 プロトコル 規格を参照してください。

1 つの SAI を AC’97 の 2 点間通信に的を絞って使用することができます。

受信モードでは、スロット 0 のコーデックレディビットがローにデコードされた場合、AC’97 リンクコントローラとして機能する SAI は、FIFO リクエストを必要としないため FIFO 内のデータストレージも不要です。SAI_xIM レジスタの CNRDYIE ビットが有効化されている場合、SAI_xSR レジスタの CNRDY フラグがセットされると割込みが生成されます。このフラグは、AC’97 プロトコル専用です。

AC’97 モードでのクロックジェネレータのプログラミング

AC’97 モードでは、フレーム長は 256 ビットに固定され、その周波数は 48 kHz に設定する必要があります。セクション 40.4.8 : SAI クロックジェネレータに示す式は、適切なフレーム･レート（F_{FS_x}）を生成するために、FRL = 255 で使用する必要があります。

40.4.12 SPDIF 出力

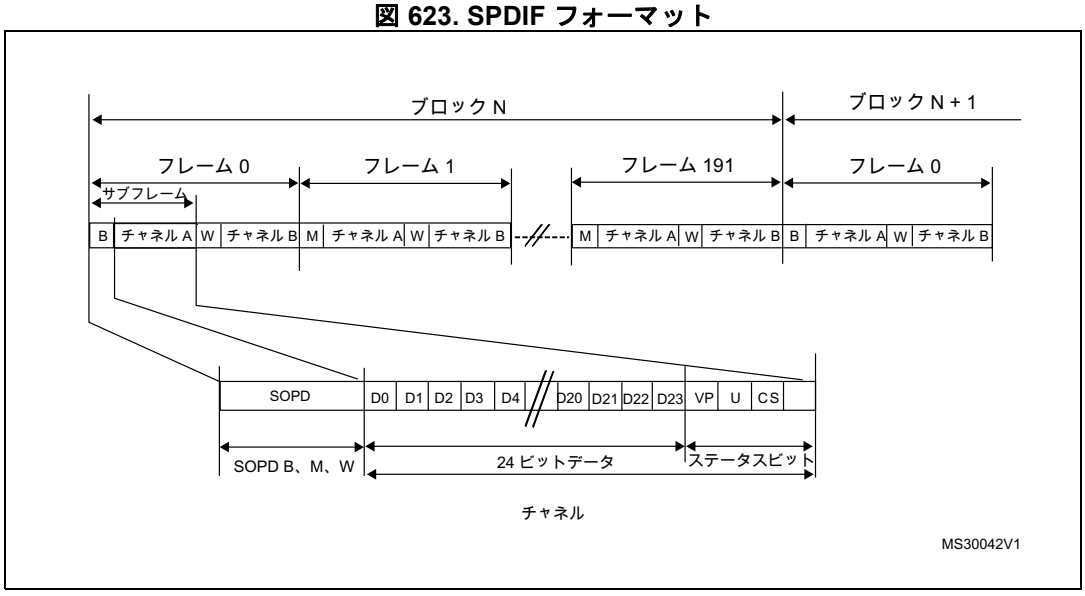
SPDIF インタフェースは送信モードでのみ使用できます。このインタフェースはオーディオ IEC60958 をサポートします。

SPDIF モードを選択するには、SAI_xCR1 レジスタの PRTCFG[1:0] ビットに 01 をセットします。

SPDIF プロトコルを使用する場合

- SD データラインのみが有効です。
- I/O ピンの FS、SCK、MCLK は空けてあります。
- MODE[1] ビットは、SAI のクロックジェネレータを有効にしたり SD ラインでのデータレートを管理するために強制的に 0 にして、マスタモードを選択します。
- データサイズは強制的に 24 ビットに設定されます。SAI_xCR1 レジスタの DS[2:0] ビットにセットされた値は無視されます。
- クロックジェネレータのシンボルレートを、ビットクロックがシンボルレートの 2 倍でなければならないことを考慮して、定義する必要があります。データはマンチェスタープロトコルでコード化されます。
- SAI_xFRCR レジスタおよび SAI_xSLOTR レジスタは無視されます。SAI は、SPDIF プロトコル要件（図 623 を参照）に適合するように内部的に設定されます。

図 623. SPDIF フォーマット



SPDIF ブロックには 192 個のフレームが含まれています。各フレームは、2 個の 32 ビットサブフレーム（一般に左チャネル用と右チャネル用に 1 つずつ）で構成されています。各サブフレームは SOPD パターン（4 ビット）で構成されています。このパターンを使用して、サブフレームがブロックの先頭である（つまり、チャネル A を識別している）、ブロックのどこかでチャネル A を識別している、またはチャネル B を参照しているのいずれかを指定します（表 372 を参照）。チャネル情報の次の 28 ビットは 24 ビットのデータと 4 ビットのステータス情報から成ります。

表 372. SOPD パターン

SOPD	プリアンブルのコード化		説明
	最終ビットが 0	最終ビットが 1	
B	11101000	00010111	ブロックの始めにあるチャネル A データ
W	11100100	00011011	ブロックのどこかにあるチャネル B データ
M	11100010	00011101	チャネル A データ

SAI_xDR に格納されているデータは次のような内容でなければなりません。

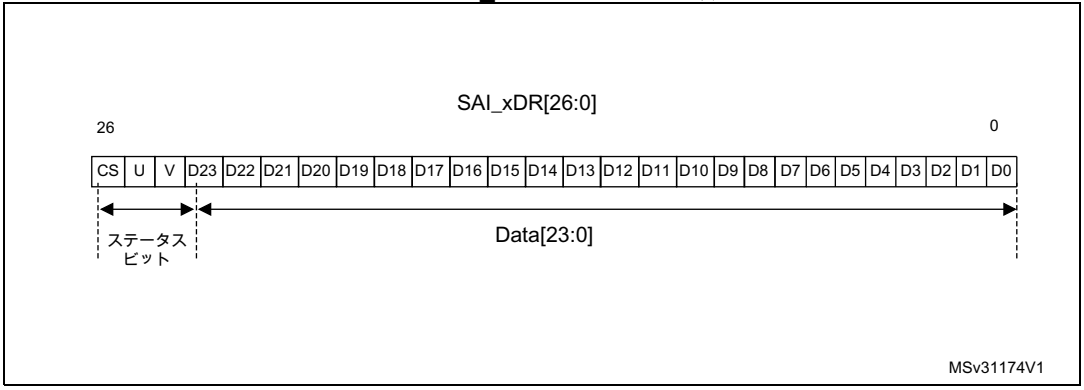
- SAI_xDR[26:24] は、チャネルのステータスビット、ユーザビット、有効性ビットを含みます。
- SAI_xDR[23:0] は、検討されたチャネルに関するデータ 24 ビットを含みます。

データ･サイズが 20 ビットの場合、データは SAI_xDR[23:4] に配置する必要があります。

データ･サイズが 16 ビットの場合、データは SAI_xDR[23:8] に配置する必要があります。

SAI_xDR[23] は常に MSB を表します。

図 624. SAI_xDR レジスタの順序



注： **必ず LSB（最下位ビット）から転送されます。**

SAI はまず最初にブロック内の各サブフレームの適切なプリアンブル部を送信します。次に、SAI_xDR が SD ラインに送信されます（マンチェスターコード化）。SAI は、表 373 の記載通りに計算されたパリティビットを転送することで、サブフレームを終了させます。

表 373. パリティビットの計算

SAI_xDR[26:0]	転送されるパリティビット P の値
奇数個の”0 ”	0
奇数個の”1 ”	1

SAI は送信モードでしか動作できないため、アンダーランは SPDIF モードの SAI_xSR レジスタで有効な唯一のエラーフラグです。そのため、アンダーラン割込みまたはアンダーランステータスビットによって検出されたアンダーランエラーから復帰するためには、次のシーケンスを実行する必要があります。

1. DMA が使用されている場合は、DMA ストリームを無効にします（DMA ペリフェラルを使用）。
2. SAI を無効にし、SAI_xCR1 レジスタの SAIEN ビットをポーリングすることでペリフェラルが物理的に無効化されていることを確認します。

3. SAI_xCLRFR レジスタの COVRUNDR フラグをクリアします。
4. FIFO を一掃するために、SAI_xCR2 の FFLUSH ビットをセットします。

   ソフトウェアで、新しいブロックの始めに今後格納するデータ（プリアンブル部 B 用のデータ）のアドレスを指し示す必要があります。DMA が使用されている場合、DMA ソースベースアドレスのポインタは適宜更新してください。
5. DMA を使用して、新しいソースベースアドレスに従ってデータ転送を管理する場合は、DMA ストリーム（DMA ペリフェラル）を再び有効にします。
6. SAI_xCR1 レジスタの SAIEN ビットをセットして、再び SAI を有効にします。

SPDIF ジェネレータモードでのクロックジェネレータのプログラミング

SPDIF ジェネレータ･モードの場合、SAI はシンボル･レートの 2 倍の速度のビット･クロックを供給します。下表では、オーディオ･サンプリング･レートに対するシンボル･レートの一般例を示します。

表 374. オーディオサンプリング周波数とシンボルレート

オーディオ･サンプリング周波数（F_S）	シンボル･レート
44.1 kHz	2.8224 MHz
48 kHz	3.072 MHz
96 kHz	6.144 MHz
192 kHz	12.288 MHz

より一般的には、オーディオサンプリング周波数（F_S）とビットクロックレート（F_{SCK_X}）の関係は、次の式で求められます。

$$F_S = \frac{F_{SCK_x}}{128}$$

ビットクロックレートは次のように求められます。

$$F_{SCK_x} = \frac{F_{sai_x_ker_ck}}{MCKDIV}$$

注： **上式は、SAI_ACR1 レジスタの NODIV が 1 に設定されている場合にのみ有効です。**

40.4.13 特有の機能

SAI インタフェースには、選択したオーディオプロトコルに応じて役立つ特定の機能が組み込まれています。これらの機能へは、SAI_xCR2 レジスタの特定ビットを使ってアクセスすることができます。

ミュートモード

ミュートモードは、オーディオサブブロックがトランスミッタまたはレシーバであるとき使用することができます。

送信モード時のオーディオサブブロック

送信モード時、ミュートモードはいつでも選択可能です。ミュートモードは、オーディオフレーム全体に対して有効です。処理中のフレーム内で SAI_xCR2 レジスタの MUTE ビットがセットされると、ミュートモードが有効になります。

ミュートモードビットは、フレームの終わりでのみ認識されます。この時点でミュートモードビットがセットされた場合は、ミュートモードは新しいオーディオフレームの先頭から次に来るフレームの

終了までフレーム全体に対して有効になります。その後、このビットを認識して、次のフレームも引き続きミュート･フレームであるかどうかを決定します。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 以下の場合、ミュート･モード中に送信される値が 0 なのか、または各スロットの最後の値なのかを指定することができます。この選択は、SAI_xCR2 レジスタの MUTEVAL ビットを使って行います。

SAI_xSLOTR レジスタの NBSLOT[3:0] ビットに設定されたスロット数が 2 より大きい場合、各スロットの各ビットで 0 値が送信されるため、SAI_xCR2 の MUTEVAL ビットは無意味となります。

ミュートモード時は、FIFO ポインタのインクリメントは続きます。つまり、FIFO にデータは供給され続けますが、ミュートモードの間は破棄されます。

受信モード時のオーディオサブブロック

受信モードでは、指定されかつ有効なオーディオフレームのすべてのスロットが、指定された連続した数のオーディオフレームに対して 0 を受信したとき、外部 トランスミッタから送信されたミュートモードを検出することが可能です（ SAI_xCR2 レジスタの MUTECNT[5:0] ビット）。

ミュートフレーム数が検出されると、SAI_xSR レジスタの MUTEDET フラグがセットされて、SAI_xCR2 の MUTEDETIE ビットがセットされている場合は割込みを生成することができます。

オーディオサブブロックが無効化されたとき、または有効なスロットがオーディオフレーム内で 1 つ以上のデータを受信したとき、ミュートフレームカウンタがクリアされます。カウンタが MUTECNT[5:0] ビットに指定された値に到達すると、割込みが 1 回だけ生成されます。次に、カウンタがクリアされると割込みイベントが再初期化されます。

注： **ミュートモードは SPDIF オーディオブロックには使用できません。**

モノラル／ステレオモード

送信モードでは、スロット数が 2 と想定し（SAI_xSLOTR で NBSLOT[3:0] = 0001）、メモリ内でデータを前処理することなくモノラルモードを指定できます。このような場合、スロット 0 のデータがデータ･スロット 1 へ複製されるため、FIFO へのアクセス･タイムは半減されます。

モノラル･モードを有効にするには、

1. SAI_xCR1 レジスタの MONO ビットに 1 をセットします。
2. SAI_xSLOTR レジスタの NBSLOT ビットに 1、SLOTEN ビットに 3 をセットします。

受信モードでは、MONO ビットをセットすることができますが、送信モードの場合と同様にスロット数が 2 のときだけ意味があります。このビットをセットした場合、スロット 0 のデータだけが FIFO に格納されます。この場合、スロット 1に属するデータは前のスロットと同じと見なされるため破棄されます。受信時のデータの流れが左右でデータが異なる真のステレオ･オーディオ･フローである場合、MONO ビットは意味を成しません。出力ステレオファイルから等価なモノラルファイルへの変換は、ソフトウェアで行われます。

コンパンディングモード

通信アプリケーションでは、データコンパンディングアルゴリズムにより送信データまたは受信データの処理が必要となることがあります。

図 625 に示すように、SAI_xCR2 レジスタの COMP[1:0] ビットに応じて（フリープロトコルモードを選択した場合のみ使用）、SD シリアル出力ラインに送信する前に処理（圧縮）するか否か、あるいは SD シリアル入力ラインから受信した後にデータを処理（伸長）するか否かをソフトウェアで選択することができます。サポートされている 2 つのコンパンディングモードは μ-Law と A-Law(l対数)で、これらは CCITT G.711 勧告の一部となっています。

米国と日本で採用されているコンパンディング規格は μ-Law であり、14 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 10）。

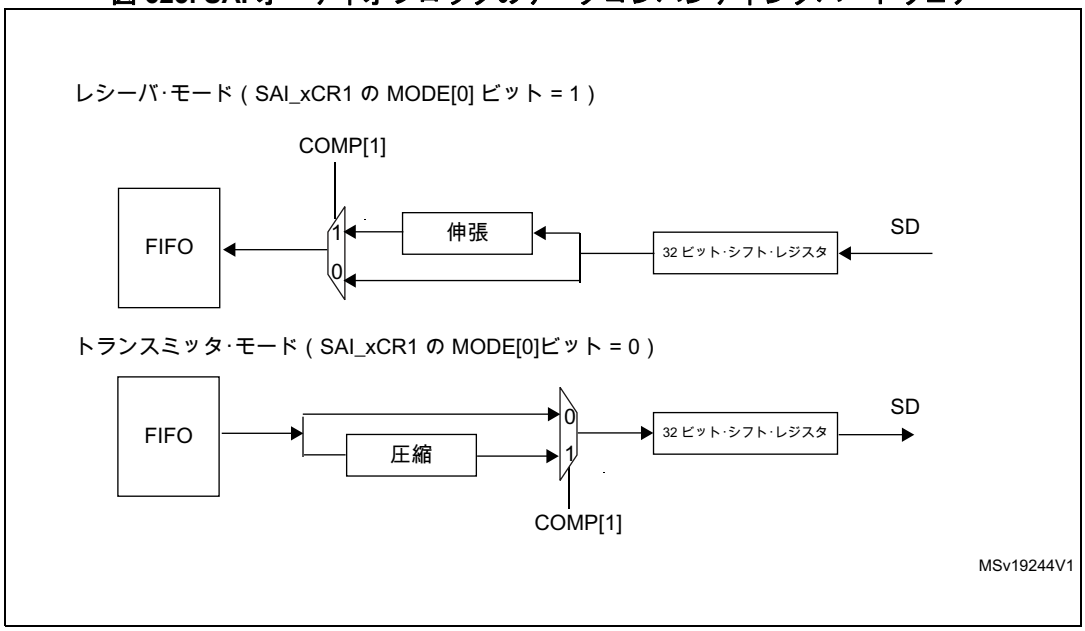
ヨーロッパのコンパンディング規格は A-Law であり、13 ビットのダイナミックレンジが確保できます（SAI_xCR2 レジスタで COMP[1:0] = 11）。

コンパンディング規格（μ-Law または A-Law）は、SAI_xCR2 レジスタの CPL ビットの設定に応じて、1 の補数表現または 2 の補数表現に基づいて計算することができます。

μ-Law および A-Law 規格では、データを MSB 詰めの 8 ビットコードにコード化します。コンパンドされたデータは常に 8 ビット幅です。このため、SAI オーディオ･ブロックが有効化されていて（SAI_xCR1 レジスタで SAIEN ビット = 1）、さらに、COMP[1:0] ビットによりこれら 2 つのコンパンディング･モードのいずれかを選択している場合、SAI_xCR1 レジスタの DS[2:0] ビットは 強制的に010 に設定されます。

コンパンディングが不要な場合は、COMP[1:0] ビットをクリアしたままにします。

図 625. SAI オーディオブロックのデータコンパンディングハードウェア



1. AC'97 または SPDIF を選択した場合は適用されません。

伸張モードまたは圧縮モードは SAI_xCR2 により自動的に選択されます。

- SAI オーディオ･ブロックがトランスミッタに設定され、かつ SAI_xCR2 レジスタの COMP[1] ビットがセットされている場合、圧縮モードが適用されます。
- SAI オーディオ･ブロックがレシーバに指定された場合、伸長アルゴリズムが適用されます。

非アクティブ･スロットでの出力データ･ライン管理

送信モードでは、非アクティブスロットがデータラインに送信されたときに出力となっている SDラインの動作を選択することができます（TRIS ビットを使用）。

- 非アクティブスロットが送信されるとき、SAI が SD 出力ラインを 0 に固定することもできれば、
- 送信データの最終ビットの終わりでラインを HI-Z 状態にして、このノードに接続された他のトランスミッタへラインを開放することもできます。

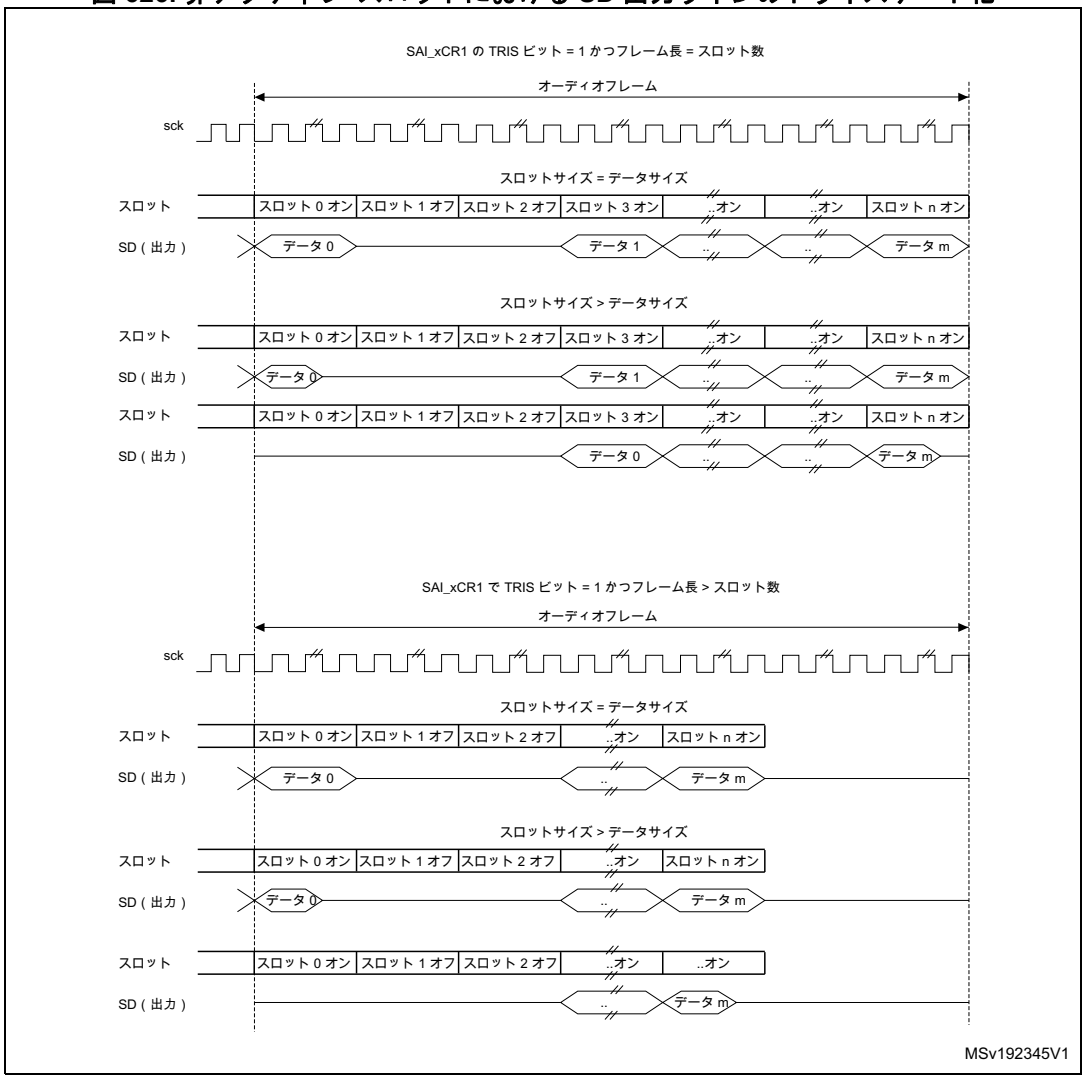
2 つのトランスミッタが同じ SD 出力ピンを同時に駆動しないように注意することが重要です。同時駆動した場合、短絡の原因となります。送信間の間隔を確保するため、データが 32 ビットより短い場合は、SAI_xSLOTR レジスタの SLOTSZ[1:0] ビット = 10 に設定することにより、データを 32 ビットに拡張することができます。すると、次のスロットが非アクティブに指定された場合、アクティブ･ス

ロットの LSB の終わりで（データを 32 ビットに拡張するため 0 を詰めるフェーズで）SD 出力ピンがトライステートになります。

さらに、スロット数とスロット･サイズの積がフレーム長より小さい場合、0 を詰めてオーディオ･フレームを完成させるときに SD 出力ラインがトライステートになります。

これらの動作を図 626 に示します。

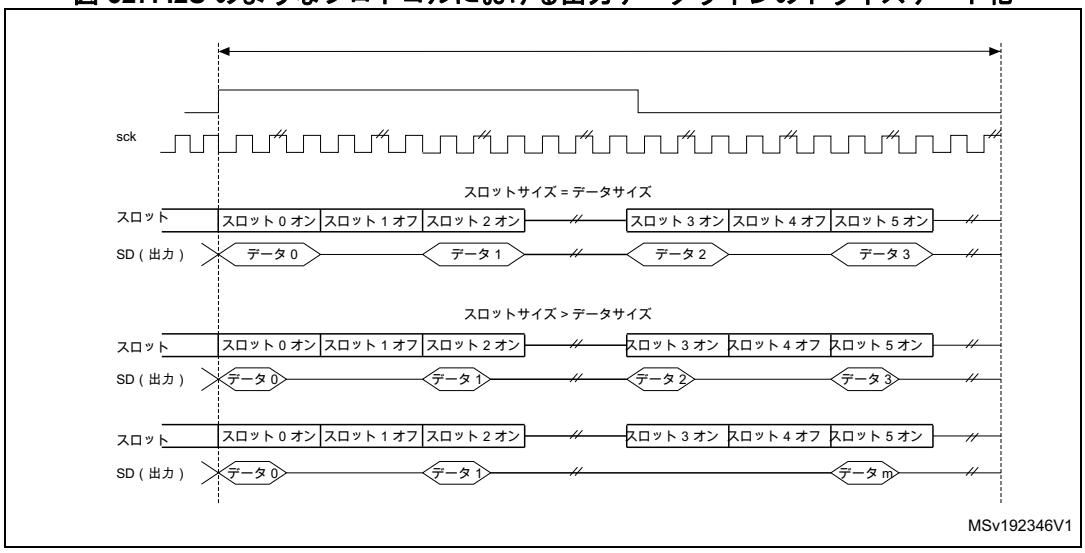
図 626. 非アクティブ･スロットにおける SD 出力ラインのトライステート化



MSv192345V1

選択したオーディオ･プロトコルが FS 信号をフレーム開始とチャネル･サイド識別として使用する場合（SAI_xFRCR レジスタでビット FSDEF = 1）、図 627 に従ってトライステート･モードが管理されます（この図では、SAI_xCR1 レジスタのビット TRIS = 1、FSDEF = 1、ハーフフレーム長 > スロット数 / 2、NBSLOT = 6）。

図 627. I2S のようなプロトコルにおける出力データラインのトライステート化



SAI_xCR2 レジスタの TRIS ビットがクリアされている場合、図 626 と図 627 に示す SD 出力ライン上のすべてのハイインピーダンス状態は 0 値での駆動として置き換えます。

40.4.14 エラーフラグ

SAI は次のエラーフラグを実装しています。

- FIFO オーバーラン／アンダーラン
- 予測フレーム同期検出
- レイトフレーム同期検出
- コーデックノットレディ（AC'97 専用）
- マスタモードにおける不正クロック設定

FIFO オーバーラン/アンダーラン（OVRUDR）

FIFO オーバーラン/アンダーランビットは、SAI_xSR レジスタで OVRUDR と呼ばれます。

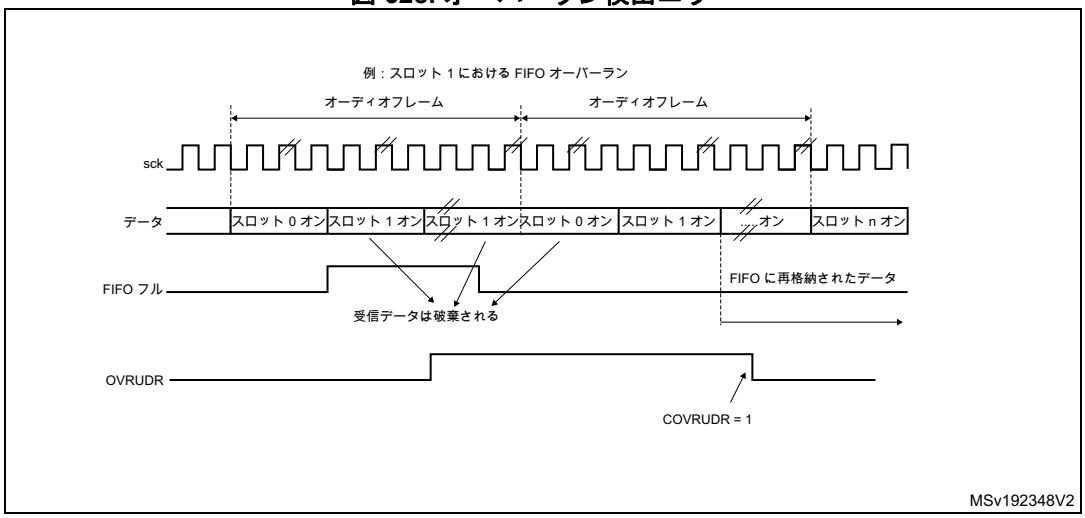
オーディオブロックはレシーバまたはトランスミッタのいずれかであり、SAI 内の各オーディオブロックに固有の SAI_xSR レジスタを搭載しているため、オーバーランエラーとアンダーランエラーは、同じビットを使用しています。

オーバーラン

オーディオブロックがレシーバに設定されていて、FIFO がフルで受信データを保存できないときにオーディオフレームでデータが受信された場合に、オーバーラン状態が発生します。この場合、受信データが失われ、SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割込みが生成されます。オーバーランが発生したスロット番号が、内部に保存されます。FIFO に新しいデータを格納する空きができるまで、データは FIFO に保存されません。FIFO に 1 データ分以上の空きがある場合、SAI オーディオ･ブロック･レシーバは、オーバーラン状態が検出されたとき、内部に保存されたスロット番号からの（新しいオーディオ･フレームの）新しいデータを保存します。これにより、転送先メモリでのデータ･スロットの不整合を回避します（図 628 を参照）。

SAI_xCLRFR レジスタの COVRUDR ビットがセットされると、OVRUDR フラグがクリアされます。

図 628. オーバーラン検出エラー



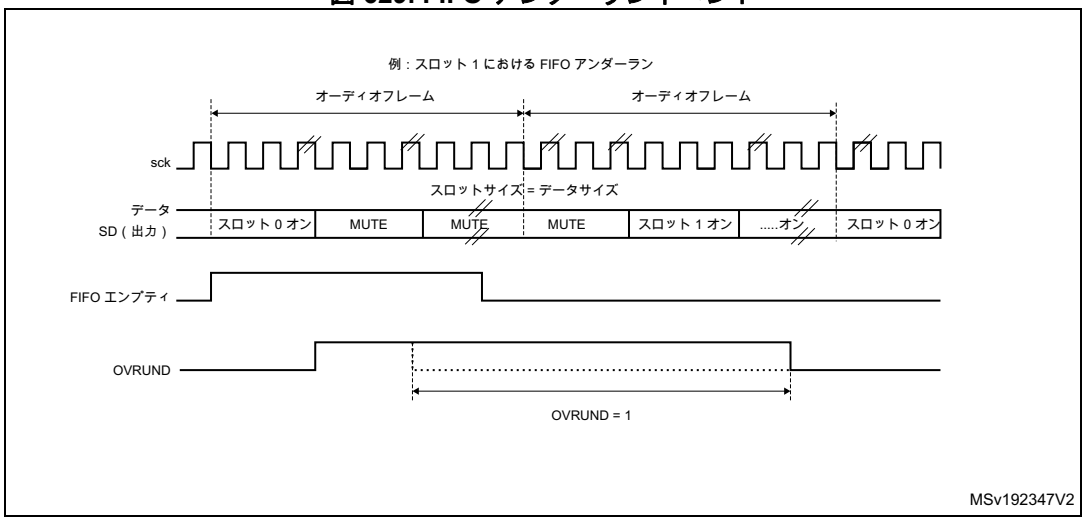
アンダーラン

SAI 内のオーディオブロックがトランスミッタで、かつデータの送信が必要なとき FIFO がエンプティ（空）の場合に、アンダーランが発生します。アンダーランが検出されると、イベントが発生したスロット番号が保存され、FIFO がアンダーランが検出されたスロットに対応するデータを送信する準備ができるまで MUTE 値（00）が送信されます（図 629 を参照）。これにより、メモリポインタとオーディオフレーム内のスロットの同期はずれを回避します。

アンダーランイベントにより SAI_xSR レジスタの OVRUDR フラグがセットされ、SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合、割込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの COVRUDR ビットをセットします。

アンダーランイベントは、オーディオサブブロックがマスタまたはスレーブとして設定されている場合に発生します。

図 629. FIFO アンダーランイベント



予測フレーム同期検出（AFSDET）

AFSDET フラグはスレーブモードでのみ使用されます。マスタモードでは決してアサートされません。フレーム長、フレーム極性、フレームオフセットが設定されていて既知であるため、フレーム同期 （FS）が予測より早く検出されることを示します。

予測フレームの検出により、SAI_xSR レジスタの AFSDET フラグがセットされます。

この検出は、現在のオーディオフレームに影響を与えません。現在のオーディオフレームは予測 FS に影響されません。つまり、現在のオーディオフレームを乱すことなく、FS 信号に関する「寄生」イベントフラグが立つということです。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は、割込みが生成されます。AFSDET フラグをクリアするには、SAI_xCLRFR レジスタの CAFSDET ビットをセットする必要があります。

予測フレーム検出エラーの後にマスタと再同期させるには、次の 4 ステップが必要です。

1. SAI_xCR1 レジスタの SAIEN ビットをリセットして、SAI ブロックを無効化します。SAI が無効であることを確認するには、SAIEN ビットを読み返し、それが 0 にセットされていることを確認します。
2. SAI_xCR2 レジスタの FFLUS ビットを使って FIFO を一掃します。
3. SAI ペリフェラルを再度有効化します（SAIEN ビットを 1 に設定）。
4. SAI ブロックは FS のアサートを待ってマスタとの同期を再開します。

注： **AFSDET フラグは AC’97 モードではアサートされません。これは SAI オーディオ･ブロックがリンク･コントローラとして機能し、スレーブに指定された場合でも FS 信号を生成するためです。SPDIF モードでは FS 信号は使用されないため、このフラグは意味を持ちません。**

レイトフレーム同期検出

SAI オーディオブロックがスレーブとして動作する場合のみ、SAI_xSR レジスタの LFSDET フラグをセットすることができます。フレーム長、フレーム極性、フレームオフセット設定は、SAI_xFRCR レジスタから知ることができます。

外部マスタが FS 信号を予想タイミングで送信しない（信号の発生が遅すぎる）場合、LFSDET フラグがセットされ､SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

SAI_xCLRFR レジスタの CLFSDET ビットがセットされると、LFSDET フラグがクリアされます。

レイトフレーム同期検出フラグは、対応するエラーが検出されるとセットされます。SAI はマスタと再同期させる必要があります（シーケンスは、予測フレーム同期検出（AFSDET）を参照）。

ノイズの多い環境では、オーディオブロックステートマシンが SCK クロック上のグリッチを間違って検出し、SAI データが間違ったフレーム位置にシフトされることがあります。このようなイベントは SAI によって検出され、レイトフレーム同期検出エラーとして報告することができます。

外部マスタが連続モードでオーディオデータフレーム転送を管理していない場合は破壊は起こりませんが、ほとんどの場合、アプリケーションでこの様な使われ方はされません。この場合、LFSDET フラグがセットされます。

注： **LFSDET フラグは AC’97 モードではアサートされません。これは SAI オーディオ･ブロックがリンク･コントローラとして機能し、スレーブと指定された場合でも FS 信号を生成するためです。SPDIF モードでは FS 信号はプロトコルで使用されないため、このフラグは意味を持ちません。**

コーデックノットレディ（CNRDY AC’97）

SAI オーディオブロックが AC’97 モードで動作するように設定されている場合にのみ SAI_xSR レジスタの CNRDY フラグが関係します（SAI_xCR1 レジスタの PRTCFG[1:0] ビット = 10）。SAI_xIM レジスタの CNRDYIE ビットがセットされている場合、CNRDY フラグがセットされたときに割込みが生成されます。

AC’97 オーディオフレームのタグ 0（スロット 0）の受信中にコーデックが通信の準備ができていないとき、CNRDY フラグがアサートされます。この場合、コーデックが動作可能であることをタグ 0 が示すまではコーデックは動作可能状態にはないため、FIFO に自動的に保存されるデータはありません。コーデックが動作可能状態になると、SAI_xSLOTR レジスタで指定されたすべてのアクティブ･スロットがキャプチャされます。

CNRDY フラグをクリアするには、SAI_xCLRFR レジスタの CCNRDY ビットをセットする必要があります。

マスタモードでの不正なクロック設定（NODIV = 0 の場合）

オーディオブロックがマスタとして動作し（MODE[1] = 0）、NODIV ビットが 0 に等しい場合、次の条件が満たされて SAI が有効になるとすぐに、WCKCFG フラグがセットされます。

- (FRL + 1) は 2の累乗値ではない、かつ
- (FRL + 1) は 8～256の範囲外である。

MODE、NODIV、SAIEN ビットは SAI_xCR1 レジスタに、FRL ビットは SAI_xFRCR レジスタに 属します。

WCKCFGIE ビットがセットされている場合、SAI_xSR レジスタで WCKCFG フラグがセットされると割込みが生成されます。このフラグをクリアするには、SAI_xCLRFR レジスタの CWCKCFG ビットをセットします。

WCKCFG ビットがセットされると、オーディオブロックが自動的に無効化され、ハードウェアによって SAIEN ビットがクリアされます。

40.4.15 SAI の無効化

SAI_xCR1 レジスタの SAIEN ビットをクリアすることによって、SAIオーディオブロックをいつでも無効化することができます。すでに開始しているすべてのフレームは、SAI が動作を停止する前に自動的に完了します。現在のオーディオフレーム転送の最後に SAI が完全にオフするまで、SAIEN ビットはハイを保持します。

他の SAI と同期している SAI 内にオーディオブロックがある場合、マスタになっている SAI を先に無効化する必要があります。

40.4.16 SAI DMA インタフェース

CPU を開放し、バスの帯域幅を最適化するため、各 SAI オーディオブロックには SAI_xDR レジスタ（内部 FIFO へのアクセス用）に対して読み書きを行うための独立した DMA インタフェースを搭載しています。オーディオサブブロック毎に 1 つの DMA チャネルがあり、基本的な DMA リクエスト／確認応答プロトコルに対応しています。

DMA 転送を行うようにオーディオサブブロックを設定するには、SAI_xCR1 レジスタの DMAEN ビットをセットします。DMA リクエストは、FIFO 閾値レベルに応じて FIFO コントローラによって直接管理されます（詳細は、セクション 40.4.9 : 内部 FIFO を参照してください）。DMA 転送方向は、SAI オーディオサブブロック設定と次の様に関連付けされています。

- オーディオブロックがトランスミッタとして動作する場合、オーディオブロックの FIFO コントローラが DMA リクエストを出力して SAI_xDR レジスタに書き込まれたデータを FIFO にロードします。
- オーディオブロックがレシーバとして動作する場合、DMA リクエストは SAI_xDR レジスタからの読出し動作に関わります。

SAI インタフェースを DMA モードに設定するには、次のシーケンスに従ってください。

1. SAI と FIFO 閾値レベルを設定して、DMA リクエストを開始するタイミングを指定します。
2. SAI DMA チャネルを設定します。
3. DMA を有効にします。
4. SAI インタフェースを有効にします。

注： **SAI ブロックを設定する前に、SAI DMA チャネルを無効化する必要があります。**

40.5 SAI 割込み

SAI は表 375 に示す 7 つの割込みソースに対応しています。

表 375. SAI 割込みソース

項目（割込みの略称）	割込みソース	割込みグループ	オーディオブロックモード	割込みイネーブル	割込みクリア
SAI	FREQ	FREQ	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの FREQIE	下記に応じてクリアします。 – FIFO 閾値設定（SAI_xCR2 の FLVL ビット） – 通信方向（トランスミッタ／レシーバ） 詳細については、セクション 40.4.9：内部 FIFOを参照してください。
	OVRUDR	ERROR	マスタ／スレーブ 受信または送信モード	SAI_xIM レジスタの OVRUDRIE	SAI_xCLRFR レジスタの COVRUDR = 1
	AFSDET	ERROR	スレーブ （AC'97 モードおよび SPDIF モードでは使用しません。）	SAI_xIM レジスタの AFSDETIE	SAI_xCLRFR レジスタで CAFSDET = 1
	LFSDET	ERROR	スレーブ （AC'97 モードおよび SPDIF モードでは使用しません。）	SAI_xIM レジスタの LFSDETIE	SAI_xCLRFR レジスタで CLFSDET = 1
	CNRDY	ERROR	スレーブ （AC'97 モードの場合のみ）	SAI_xIM レジスタの CNRDYIE	SAI_xCLRFR レジスタで CCNRDY = 1
	MUTEDET	MUTE	マスタ／スレーブ 受信モードのみ	SAI_xIM レジスタの MUTEDETIE	SAI_xCLRFR レジスタで CMUTEDET = 1
	WCKCFG	ERROR	マスタ（SAI_xCR1 レジスタで NODIV = 0 の場合）	SAI_xIM レジスタの WCKCFGIE	SAI_xCLRFR レジスタで CWCKCFG = 1

割込みを有効にするには、次のシーケンスに従ってください。

1. SAI 割込みを無効化します。
2. SAI を設定します。
3. SAI 割込みソースを設定します。
4. SAI を有効化します。

40.6 SAI レジスタ

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

40.6.1 SAI 設定レジスタ 1（SAI_ACR1）

アドレス･オフセット：0x004

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **MCKEN**：マスタ･クロック生成有効化

0：マスタクロックは生成されません。

1：マスタクロックが SAIEN ビットから独立して生成されます。

ビット 26 **OSR**：マスタ･クロックのオーバーサンプリング比

このビットは、NODIV ビットが 0 にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット 25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比 1 のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタ･クロック周波数は セクション 40.4.8 : SAI クロックジェネレータで指定された式で計算されます。

オーディオ･ブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 19 **NODIV**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が 256 または 512 に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率が FRL[7:0] に依存します。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA 有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。

ビット 16 **SAIEN**：オーディオ･ブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 に設定する前に、それが 0 に設定されていることを確認します。そうしないと、有効化のコマンドが考慮されないことになります。

このビットを使って、SAI オーディオ･ブロックの状態を制御できます。オーディオ･フレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオ･フレーム転送の転送後にセル全体が完全に無効化されます。

0：SAI オーディオブロックは無効です。

1：SAI オーディオブロックは有効です。

注： SAI ブロック（A または B）をマスタモードに設定した場合、SAIEN ビットをセットする前に SAI の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラル･モード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]**：同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：予約済み

11：予約済みです。

注： オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR**：クロック･ストローブ･エッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST**：LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC’97 ではデータは常に MSB 先頭で転送されるため、このビットは AC’97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]**：データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 ビットに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済みです。

001：予約済みです。

010：8 ビット

011：10 ビット

100：16 ビット

101：20 ビット

110：24 ビット

111：32 ビット

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **PRTCFG[1:0]**：プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリー･プロトコルを選択すると、オーディオ･ブロックの強力な設定機能を使用し、大部分の設定レジスタ･ビットとフレーム設定レジスタを設定して、特定のオーディオ･プロトコル（例えば、I2S、LSB/MSB 詰め、TDM、PCM/DSP）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済みです。

ビット 1:0 **MODE[1:0]**：SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。

40.6.2 SAI 設定レジスタ 1（SAI_BCR1）

アドレス･オフセット：0x024

リセット値：0x0000 0040

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DS[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
		rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27 **MCKEN**：マスタ･クロック生成有効化

0：マスタクロックは生成されません。

1：マスタクロックが SAIEN ビットから独立して生成されます。

ビット 26 **OSR**：マスタ･クロックのオーバーサンプリング比

このビットは、NODIV ビットが 0 にセットされているときにのみ意味を持ちます。

0：マスタクロック周波数 = F_{FS} x 256

1：マスタクロック周波数 = F_{FS} x 512

ビット 25:20 **MCKDIV[5:0]**：マスタクロック分周回路

これらのビットは、ソフトウェアによってセット／クリアされます。

000000：分周比 1 のカーネルクロック入力（sai_x_ker_ck）。

その他の場合、マスタ･クロック周波数は セクション 40.4.8 : SAI クロックジェネレータで指定された式で計算されます。

オーディオ･ブロックがスレーブの場合、これらのビットに意味はありません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 19 **NODIV**：分周器なし

このビットは、ソフトウェアによってセット／クリアされます。

0：マスタクロックジェネレータとフレーム同期の比率が 256 または 512 に固定されます。

1：マスタクロックジェネレータとフレーム同期の比率が FRL[7:0] に依存します。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17 **DMAEN**：DMA 有効化

このビットは、ソフトウェアによってセット／クリアされます。

0：DMA は無効です。

1：DMA は有効です。

注： オーディオブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されていますので、受信モードでの DMA リクエストを回避するために、MODE[1:0] ビットは DMAEN をセットする前に設定する必要があります。

ビット 16 **SAIEN**：オーディオ･ブロック有効化

このビットはソフトウェアでセットされます。

オーディオブロックをスイッチオフするには、アプリケーションソフトウェアでこのビットを 0 に設定し、このビットが 0 に読み返されるまでポーリングする必要があります。これは、ブロックを完全に無効にするということです。このビットを 1 に設定する前に、それが 0 に設定されていることを確認します。そうしないと、有効化のコマンドが考慮されないことになります。

このビットを使って、SAI オーディオ･ブロックの状態を制御できます。オーディオ･フレーム転送の途中で無効化した場合、処理中の転送は完了し、このオーディオ･フレーム転送の転送後にセル全体が完全に無効化されます。

0：SAI オーディオブロックは無効です。

1：SAI オーディオブロックは有効です。

注： SAI ブロック（A または B）をマスタモードに設定した場合、SAIEN ビットをセットする前に SAI の入力にクロックが存在する必要があります。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **OUTDRIV**：出力駆動

このビットは、ソフトウェアによってセット／クリアされます。

0：SAIEN がセットされたときオーディオブロック出力が駆動されます。

1：このビットをセットした直後にオーディオブロック出力が駆動されます。

注： このビットは、オーディオブロックを有効化する前で、かつオーディオブロックを設定した後にセットする必要があります。

ビット 12 **MONO**：モノラル･モード

このビットは、ソフトウェアによってセット／クリアされます。このビットはスロット数が 2 のときにのみ意味を持ちます。モノラルモードを選択した場合、オーディオブロックがトランスミッタとして動作する場合にスロット 0 のデータがスロット 1 に複製されます。受信モードでは、スロット 1 は破棄され、スロット 0 から受け取ったデータのみが保存されます。詳細については、セクション ：モノラル／ステレオモードを参照してください。

0：ステレオモード

1：モノラルモード

ビット 11:10 **SYNCEN[1:0]**：同期有効化

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオサブブロックを無効化して、これらのビットを設定する必要があります。

00：オーディオサブブロックは非同期モードです。

01：オーディオサブブロックは他の内部オーディオサブブロックと同期します。この場合、オーディオサブブロックはスレーブモードに設定する必要があります。

10：予約済み

11：予約済みです。

注： オーディオサブブロックは、SPDIF モードが有効な場合、非同期として設定する必要があります。

ビット 9 **CKSTR**：クロック･ストローブ･エッジ

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。このビットは、SPDIF オーディオプロトコルでは意味を持ちません。

0：SCK 立ち上がりエッジでの SAI の変化によって信号が生成され、SCK 立ち下がりエッジでの SAI のサンプリングによって信号が受信されます。

1：SCK 立ち下がりエッジでの SAI の変化によって信号が生成され、SCK 立ち上がりエッジでの SAI のサンプリングによって信号が受信されます。

ビット 8 **LSBFIRST**：LSB ファースト

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、このビットを設定する必要があります。AC’97 ではデータは常に MSB 先頭で転送されるため、このビットは AC’97 オーディオプロトコルでは意味を持ちません。SPDIF ではデータは常に LSB 先頭で転送されるため、このビットは SPDIF オーディオプロトコルでは意味を持ちません。

0：データは MSB から先に転送されます。

1：データは LSB から先に転送されます。

ビット 7:5 **DS[2:0]：**データサイズ

これらのビットは、ソフトウェアによってセット／クリアされます。SPDIF プロトコルを選択した場合（PRTCFG[1:0] ビット）、フレームおよびデータサイズは固定されているため、これらのビットは無視されます。コンパンディングモードを選択した場合（COMP[1:0] ビット）、データサイズはアルゴリズムによって 8 ビットに固定されているため、DS[1:0] は無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

000：予約済みです。

001：予約済みです。

010：8 ビット

011：10 ビット

100：16 ビット

101：20 ビット

110：24 ビット

111：32 ビット

ビット 4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **PRTCFG[1:0]：**プロトコル設定

これらのビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを無効化して、これらのビットを設定する必要があります。

00：フリープロトコル。フリー･プロトコルを選択すると、オーディオ･ブロックの強力な設定機能を使用し、大部分の設定レジスタ･ビットとフレーム設定レジスタを設定して、特定のオーディオ･プロトコル（例えば、I^2S、LSB/MSB 詰め、TDM、PCM/DSP）を処理することができます。

01：SPDIF プロトコル

10：AC'97 プロトコル

11：予約済みです。

ビット 1:0 **MODE[1:0]：**SAIx オーディオブロックモード

これらのビットは、ソフトウェアによってセット／クリアされます。SAIx オーディオブロックを無効化した場合、これらのビットを設定する必要があります。

00：マスタトランスミッタ

01：マスタレシーバ

10：スレーブトランスミッタ

11：スレーブレシーバ

注： オーディオブロックを SPDIF モードに設定した場合、マスタ送信モードが強制されます（MODE[1:0] = 00)）。マスタ送信モードでは、オーディオブロックはただちに FS およびクロックの生成を開始します。

40.6.3 SAI 設定レジスタ 2（SAI_ACR2）

アドレス･オフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTEVAL	MUTE	TRIS	FFLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law（対数）は CCITT G.711 勧告の一部であり、使用される補数のタイプは CPL ビットによって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注： コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュート･フレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを有効化（SAIEN）する前に書き込む必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データ･ラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、セクション ：非アクティブ･スロットでの出力データ･ライン管理を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読出しと書込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割込みを無効化する必要があります。

ビット 2:0 **FTH[2:0]**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済みです。

110：予約済みです。

111：予約済みです。

40.6.4 SAI 設定レジスタ 2（SAI_BCR2）

アドレス･オフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP[1:0]		CPL	MUTECNT[5:0]						MUTEVAL	MUTE	TRIS	FFLUSH	FTH[2:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	w	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:14 **COMP[1:0]**：コンパンディングモード。

これらのビットは、ソフトウェアによってセット／クリアされます。μ-Law と A-Law（対数）は CCITT G.711 勧告の一部であり、使用される補数のタイプは *CPL ビット*によって決まります。

データ伸張かデータ圧縮かは、MODE[0] ビットの状態で決まります。

オーディオブロックがトランスミッタとして設定された場合、データ圧縮が適用されます。

オーディオブロックがレシーバとして設定された場合、自動的にデータ伸張が適用されます。

詳細については、セクション ：コンパンディングモードを参照してください。

00：コンパンディングアルゴリズムなし

01：予約済み。

10：μ-Law アルゴリズム

11：A-Law アルゴリズム

注： コンパンディングモードは、フリープロトコルモードが選択された場合にのみ適用されます。

ビット 13 **CPL**：補数ビット。

このビットは、ソフトウェアによってセット／クリアされます。

このビットにより、コンパンディングモードに使用される補数のタイプを指定します。

0：1 の補数表現。

1：2 の補数表現。

注： コンパンディングモードが μ-Law アルゴリズムまたは A-Law アルゴリズムの場合にのみ、このビットは有効です。

ビット 12:7 **MUTECNT[5:0]**：ミュートカウンタ。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは受信モードでのみ使用されます。

これらのビットに設定された値は、受信で検出された連続ミュートフレーム数と比較されます。ミュート･フレーム数がこの値と一致すると、MUTEDET フラグがセットされ、MUTEDETIE ビットがセットされている場合、割込みが生成されます。

詳細については、セクション ：ミュートモードを参照してください。

ビット 6 **MUTEVAL**：ミュート値。

このビットは、ソフトウェアによってセット／クリアされます。オーディオブロックを有効化（SAIEN）する前に書き込む必要があります。オーディオブロックがトランスミッタでとして動作し、スロット数が 2 以下で、かつ MUTE ビットがセットされている場合にのみ、このビットは意味を持ちます。

これより多くのスロットが指定された場合、ミュートモードでの送信時に送信されるビット値は、この MUTEVAL ビットの値とは無関係に 0 になります。

スロット数が 2 以下で MUTEVAL = 1 の場合、送信される各スロットのミュート値は、前のフレームで送信したものと同じになります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモード時にビット値 0 を送信します。

1：ミュートモード時に最後の値を送信します。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 5 **MUTE**：ミュート

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして動作する場合にのみ意味を持ちます。スロット数が 2 以下の場合、ミュート値は MUTEVAL 値に関連付けられますが、スロット数が 2 より大きい場合、ミュート値は 0 になります。

詳細については、セクション ：ミュートモードを参照してください。

0：ミュートモードではありません。

1：ミュートモードは有効です。

注： このビットは意味を持たないので、SPDIF オーディオブロックには使用すべきではありません。

ビット 4 **TRIS**：データ･ラインのトライステート管理。

このビットは、ソフトウェアによってセット／クリアされます。このビットは、オーディオブロックがトランスミッタとして設定された場合にのみ意味を持ちます。オーディオブロックが SPDIF モードに設定されている場合、このビットは使用されません。このビットは SAI を無効にして設定してください。

詳細については、セクション ：非アクティブ･スロットでの出力データ･ライン管理を参照してください。

0：スロットが非アクティブのときでも、SD 出力ラインは SAI によって駆動されます。

1：次のスロットが非アクティブの場合、最終アクティブスロットの最終データビットの終わりで SD 出力ラインを開放（HI-Z）します。

ビット 3 **FFLUSH**：FIFO を一掃します。

このビットはソフトウェアでセットされます。常に 0 が読み出されます。このビットは SAI を無効にして設定してください。

0：FIFO を一掃しません。

1：FIFO を一掃します。このビットを 1 に設定すると、FIFO の一掃がトリガされます。すべての内部 FIFO ポインタ（読出しと書込み）がクリアされます。この場合、FIFO に残っているデータが失われます（送受信データはそれ以上失われません）。FIFO を一掃する前に、SAI DMA ストリーム／割込みを無効化する必要があります。

ビット 2:0 **FTH[2:0]**：FIFO 閾値。

このビットは、ソフトウェアによってセット／クリアされます。

000：FIFO エンプティ

001：FIFO の ¼

010：FIFO の ½

011：FIFO の ¾

100：FIFO フル

101：予約済みです。

110：予約済みです。

111：予約済みです。

40.6.5 SAI フレーム設定レジスタ（SAI_AFRCR）

アドレスオフセット：0x00C

リセット値：0x0000 0007

注： このレジスタは、AC’97および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC’97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 ビット前でアサートされます。

ビット 17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC’97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16 **FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネル専用に、残りのスロットを右チャネル専用に、それぞれ割り当てることを意味します（例えば、このビットは、I^2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC’97 または SPDIF オーディオ･ブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、ビットクロック数、(SCK) + 1 (FSALL[6:0] + 1) で指定します。

これらのビットは意味を持たないので、AC’97 または SPDIF オーディオブロックの設定には使用されません。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。

オーディオ･フレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオ･ブロックの動作は予測できません。これは、データ･サイズが 8 ビットで、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。

マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。

これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。オーディオブロックを無効化して、これらのビットを設定する必要があります。

40.6.6 SAI フレーム設定レジスタ（SAI_BFRCR）

アドレスオフセット：0x02C

リセット値：0x0000 0007

注： このレジスタは、AC'97 および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF
													rw	rw	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	FSALL[6:0]							FRL[7:0]							
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **FSOFF**：フレーム同期オフセット。

このビットは、ソフトウェアによってセット／クリアされます。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はスロット 0 の先頭ビットで開始されます。

1：FS はスロット 0 の先頭ビットの 1 ビット前でアサートされます。

ビット 17 **FSPOL**：フレーム同期極性。

このビットは、ソフトウェアによってセット／クリアされます。このビットを使用して、FS 信号上のフレーム開始のレベルを設定します。AC'97 または SPDIF オーディオブロックの設定には、意味を持たないので、使用されません。

オーディオブロックを無効化して、このビットを設定する必要があります。

0：FS はアクティブロー（立ち下がりエッジ）になります

1：FS はアクティブハイ（立ち上がりエッジ）になります

ビット 16 **FSDEF**：フレーム同期の定義。

このビットは、ソフトウェアによってセット／クリアされます。

0：FS 信号がフレーム開始信号になります

1：FS 信号がフレーム開始信号 + チャネルサイド識別子になります

このビットがセットされている場合、SAI_xSLOTR レジスタで指定されるスロット数は偶数である必要があります。これは、このスロット数の半分を左チャネル専用に、残りのスロットを右チャネル専用に、それぞれ割り当てることを意味します（例えば、このビットは、I2S プロトコルまたは MSB/LSB 詰めプロトコルなどに対して設定する必要があります）。

AC'97 または SPDIF オーディオ･ブロックの設定には、意味を持たないので、使用されません。オーディオブロックを無効化して、このビットを設定する必要があります。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **FSALL[6:0]**：フレーム同期アクティブレベル長。
これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレームにおける FS 信号のアクティブレベルの長さを、ビットクロック数、（SCK）+ 1（FSALL[6:0] + 1）で指定します。
これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。
オーディオブロックを無効化して、これらのビットを設定する必要があります。

ビット 7:0 **FRL[7:0]**：フレーム長。
これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、オーディオフレーム長を SCK クロックサイクル数で定義します。フレーム内のビット数は FRL[7:0] + 1 に等しくなります。
オーディオ･フレームで転送する最小ビット数は 8 でなければならず、その他の場合、オーディオ･ブロックの動作は予測できません。これは、データ･サイズが 8 ビットで、SAI_xSLOTR レジスタの NBSLOT[4:0] でスロット 0 が 1 つだけ指定されている場合の例です（NBSLOT[3:0] = 0000 ）。
マスタモードで、マスタクロック（MCLK_x ピンで使用可能）が使用されている場合、フレーム長は、8 から 256 のうちの 2 の累乗の数に調整する必要があります。マスタクロックが使用されていない場合（NODIV = 1）、フレーム長を 8 から 256 のうちの任意の値に設定することをお勧めします。
これらのビットは意味を持たないので、AC'97 または SPDIF オーディオブロックの設定には使用されません。

40.6.7 SAI スロットレジスタ（SAI_ASLOTR）

アドレスオフセット：0x010

リセット値：0x0000 0000

注： このレジスタは、AC'97および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。
これらのビットは、ソフトウェアによってセット／クリアされます。
各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。
0：非アクティブスロット。
1：アクティブスロット。
オーディオブロックを無効化して、スロットを有効にする必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。
これらのビットは、ソフトウェアによってセット／クリアされます。
このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。
SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。
オーディオブロックを無効化して、スロット数を設定する必要があります。
これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。

SD ラインの駆動方法については、非アクティブ･スロットでの出力データ･ライン管理 を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。

01：16 ビット

10：32 ビット

11：予約済みです。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

40.6.8 SAI スロットレジスタ（SAI_BSLOTR）

アドレスオフセット：0x030

リセット値：0x0000 0000

注： このレジスタは、AC'97および SPDIF オーディオプロトコルでは意味を持ちません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLOTEN[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
				rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

ビット 31:16 **SLOTEN[15:0]**：スロット有効化。

これらのビットは、ソフトウェアによってセット／クリアされます。

各 SLOTEN ビットは 0 から 15 までのスロット位置に対応します（最大 16 スロット）。

0：非アクティブスロット。

1：アクティブスロット。

オーディオブロックを無効化して、スロットを有効にする必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:8 **NBSLOT[3:0]**：オーディオフレーム内スロット数。

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、オーディオフレーム内のスロット数 + 1 を表します（非アクティブスロット数を含む）。最大スロット数は 16 です。

SAI_xFRCR レジスタの FSDEF ビットがセットされている場合、スロット数は偶数である必要があります。

オーディオブロックを無効化して、スロット数を設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

ビット 7:6 **SLOTSZ[1:0]**：スロットサイズ

このビットは、ソフトウェアによってセット／クリアされます。

スロットサイズは、データサイズ以上である必要があります。この条件が満たされない場合、SAI の動作は不確定となります。

SD ラインの駆動方法については、非アクティブ･スロットでの出力データ･ライン管理を参照してください。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

00：スロットサイズをデータサイズ（SAI_xCR1 レジスタの DS[3:0] で指定）に一致させます。

01：16 ビット

10：32 ビット

11：予約済みです。

ビット 5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **FBOFF[4:0]**：先頭ビットオフセット

これらのビットは、ソフトウェアによってセット／クリアされます。

このビットフィールドに設定された値は、スロット内の転送データの先頭ビット位置を定義します。これはオフセット値となっています。送信モードでは、データフィールド範囲外のビットは強制的に 0 に設定されます。受信モードでは、余分な受信ビットは無視されます。

オーディオブロックを無効化して、これらのビットを設定する必要があります。

これらのビットは、AC'97 または SPDIF モードでは無視されます。

40.6.9 SAI 割込みマスクレジスタ（SAI_AIM）

アドレスオフセット：0x014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET IE	AFSDET IE	CNRDY IE	FREQ IE	WCKCFG IE	MUTEDET IE	OVRUDR IE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると､SAI_xSR レジスタの LFSDET ビットがセットされたときに割込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオ･ブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC'97、SPDIF モードのとき、またはオーディオ･ブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割込み有効化（AC'97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC'97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC'97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割込みが生成されます。

オーディオ･ブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されているので、受信モードでの寄生割込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NODIV = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割込みが生成されます。

40.6.10 SAI 割込みマスク･レジスタ（SAI_BIM）

アドレスオフセット：0x034

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDETIE**：レイトフレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの LFSDET ビットがセットされたときに割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオ･ブロックがマスタとして動作するときは意味を持ちません。

ビット 5 **AFSDETIE**：予測フレーム同期検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの AFSDET ビットがセットされている場合は割込みが生成されます。

このビットは、AC’97、SPDIF モードのとき、またはオーディオ･ブロックがマスタとして動作するときは意味を持ちません。

ビット 4 **CNRDYIE**：コーデックノットレディ割込み有効化（AC’97）

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

割込みを有効化すると、オーディオブロックは、AC’97 フレームのスロット 0（tag0）でこのラインに接続されているコーデックが動作可能状態（レディ）であるか否かを検出します。動作可能状態にない場合は、SAI_xSR レジスタの CNRDY フラグがセットされ、割込みが生成されます。

このビットは、PRTCFG[1:0] ビットを介して AC’97 モードが選択され、かつオーディオブロックがレシーバとして動作している場合のみ意味を持ちます。

ビット 3 **FREQIE**：FIFO リクエスト割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの FREQ ビットがセットされている場合は割込みが生成されます。

オーディオ･ブロックはリセット後にトランスミッタとして動作するようにデフォルト設定されているので、受信モードでの寄生割込みを回避するために、MODE ビットは FREQIE をセットする前に設定する必要があります。

ビット 2 **WCKCFGIE**：不正クロック設定割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットは、オーディオブロックがマスタとして設定されている場合（MODE[1] = 0 かつ NODIV = 0）にのみ考慮されます。

SAI_xSR レジスタの WCKCFG フラグがセットされている場合は割込みが生成されます。

注： このビットはフリープロトコルモードでのみ使用され、他のモードでは意味を持ちません。

ビット 1 **MUTEDETIE**：ミュート検出割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの MUTEDET ビットがセットされている場合は割込みが生成されます。

オーディオブロックが受信モードに設定されている場合にのみ、このビットは意味を持ちます。

ビット 0 **OVRUDRIE**：オーバーラン/アンダーラン割込み有効化。

このビットは、ソフトウェアによってセット／クリアされます。

0：割込みは無効です。

1：割込みは有効です。

このビットをセットすると、SAI_xSR レジスタの OVRUDR ビットがセットされている場合は割込みが生成されます。

40.6.11 SAI ステータスレジスタ（SAI_ASR）

アドレスオフセット：0x018

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL[2:0]**：FIFO レベル閾値。

このビットは読出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFO エンプティ（送信モードおよび受信モード）

001：FIFO ＜ ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFO ただしフルではない（送信モード）、¾ ≤ FIFO ただしフルではない（受信モード）

101：FIFO フル（送信モードおよび受信モード）

その他：予約済みです。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイト･フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC'97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは AC'97 と SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると、このフラグはクリアされます。

ビット 4 **CNRDY**：コーデック･ノット･レディ。

このビットは読出し専用です。

0：外部 AC'97 コーデックが動作可能状態です。

1：外部 AC'97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読出しまたは書込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 40.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NODIV = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

40.6.12 SAI ステータスレジスタ（SAI_BSR）

アドレスオフセット：0x038

リセット値：0x0000 0008

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]		
													r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
									r	r	r	r	r	r	r

ビット 31:19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **FLVL[2:0]**：FIFO レベル閾値。

このビットは読出し専用です。FIFO レベル閾値フラグはハードウェアにより管理され、その設定は SAI ブロックの設定（送信モードまたは受信モード）によって決まります。

000：FIFO エンプティ（送信モードおよび受信モード）

001：FIFO < ¼ ただしエンプティではない（送信モード）、FIFO < ¼ ただしエンプティではない（受信モード）

010：¼ < FIFO ≤ ½（送信モード）、¼ ≤ FIFO < ½（受信モード）

011：½ < FIFO ≤ ¾（送信モード）、½ ≤ FIFO < ¾（受信モード）

100：¾ < FIFO ただしフルではない（送信モード）、¾ ≤ FIFO ただしフルではない（受信モード）

101：FIFO フル（送信モードおよび受信モード）

その他：予約済みです。

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **LFSDET**：レイト･フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：フレーム同期信号が正しいタイミングで現れない。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このフラグは、AC'97 または SPDIF モードでは使用されません。

SAI_xIM レジスタの LFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CLFSDET ビットをセットすると、このフラグはクリアされます。

ビット 5 **AFSDET**：予測フレーム同期検出。

このビットは読出し専用です。

0：エラーなし。

1：予想より前にフレーム同期信号が検出されました。

オーディオブロックがスレーブモードに設定された場合のみ、このフラグをセットすることができます。

このビットは AC'97または SPDIF モードでは使用されません。

SAI_xIM レジスタの AFSDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CAFSDET ビットをセットすると､このフラグはクリアされます。

ビット 4 **CNRDY**：コーデック･ノット･レディ。

このビットは読出し専用です。

0：外部 AC'97 コーデックが動作可能状態です。

1：外部 AC'97 コーデックが動作可能状態ではありません。

SAI_xCR1 レジスタで AC'97 オーディオプロトコルが選択され、かつ受信モードに設定された場合にのみ、このビットが使用されます。

SAI_xIM レジスタの CNRDYIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CCNRDY ビットをセットすると、このフラグはクリアされます。

ビット 3 **FREQ**：FIFO リクエスト。

このビットは読出し専用です。

0：FIFO リクエストはありません。

1：SAI_xDR に対する読出しまたは書込みの FIFO リクエストがあります。

このリクエストはオーディオブロック設定によって決まります。

– ブロックが送信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の書込みリクエスト動作に関わります。

– ブロックが受信モードに設定されている場合、FIFO リクエストは SAI_xDR 内の読出しリクエスト動作に関わります。

SAI_xIM レジスタの FREQIE ビットがセットされている場合、このフラグによって割込みを生成することができます。

ビット 2 **WCKCFG**：不正クロック設定フラグ。

このビットは読出し専用です。

0：クロック設定は正常です。

1：クロック設定が セクション 40.4.6：フレーム同期で規定するフレーム長仕様に関する規則に準拠していません（SAI_xFRCR レジスタの FRL[7:0] ビットの設定）。

このビットは、オーディオブロックがマスタモードで動作する場合（MODE[1] = 0 かつ NODIV = 0）にのみ使用されます。

SAI_xIM レジスタの WCKCFGIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CWCKCFG ビットをセットすると、このフラグはクリアされます。

ビット 1 **MUTEDET**：ミュート検出。

このビットは読出し専用です。

0：SD 入力ラインでミュートは検出されません

1：指定数の連続オーディオフレームに対して SD 入力ラインでミュート値が検出されました（0 値）

連続するオーディオフレームに対してオーディオフレームの各スロットで連続する 0 値が受信された場合、このフラグがセットされます（SAI_xCR2 レジスタの MUTECNT ビットをセット）。

SAI_xIM レジスタの MUTEDETIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの CMUTEDET ビットをセットすると、このフラグはクリアされます。

ビット 0 **OVRUDR**：オーバーラン/アンダーラン。

このビットは読出し専用です。

0：オーバーラン/アンダーランエラーはありません。

1：オーバーラン/アンダーランエラーが検出されました。

オーバーランおよびアンダーラン状態は、オーディオブロックがそれぞれレシーバとトランスミッタとして設定されている場合にのみ発生します。

SAI_xIM レジスタの OVRUDRIE ビットがセットされている場合は割込みが生成されます。

ソフトウェアによって SAI_xCLRFR レジスタの COVRUDR ビットをセットすると、このフラグはクリアされます。

40.6.13 SAI クリアフラグレジスタ（SAI_ACLRFR）

アドレスオフセット：0x01C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。

このビットは書込み専用です。

このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。

このビットは、AC’97 と SPDIF モードでは使用されません。

このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。

このビットは書込み専用です。

このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。

このビットは AC’97 と SPDIF モードでは使用されません。

このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデック･ノット･レディ･フラグのクリア

このビットは書込み専用です。

このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。

SAI_xCR1 レジスタで AC’97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。

このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NODIV = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

40.6.14 SAI クリアフラグレジスタ（SAI_BCLRFR）

アドレスオフセット：0x03C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
									w	w	w		w	w	w

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **CLFSDET**：レイトフレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの LFSDET フラグがクリアされます。
このビットは、AC’97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 5 **CAFSDET**：予測フレーム同期検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの AFSDET フラグがクリアされます。
このビットは AC’97または SPDIF モードでは使用されません。
このビットを読み出すと、常に 0 値が返されます。

ビット 4 **CCNRDY**：コーデック･ノット･レディ･フラグのクリア
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの CNRDY フラグがクリアされます。
SAI_xCR1 レジスタで AC’97 オーディオプロトコルが選択された場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CWCKCFG**：不正クロック設定フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの WCKCFG フラグがクリアされます。
オーディオブロックがマスタとして設定され（MODE[1] = 0）、かつ SAI_xCR1 レジスタで NODIV = 0 の 場合にのみ、このビットが使用されます。
このビットを読み出すと、常に 0 値が返されます。

ビット 1 **CMUTEDET**：ミュート検出フラグクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの MUTEDET フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

ビット 0 **COVRUDR**：オーバーラン/アンダーランクリア。
このビットは書込み専用です。
このビットを 1 に設定すると SAI_xSR レジスタの OVRUDR フラグがクリアされます。
このビットを読み出すと、常に 0 値が返されます。

40.6.15 SAI データレジスタ（SAI_ADR）

アドレスオフセット：0x020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATA[31:0]**：データ
FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読出しが行われます。

40.6.16 SAI データレジスタ（SAI_BDR）

アドレスオフセット：0x040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATA[31:0]**：データ
FIFO がフルでない場合、このレジスタへ書き込むと、FIFO への書込みが行われます。
FIFO がエンプティでない場合、このレジスタを読み出すと、FIFO からの読出しが行われます。

40.6.17 SAI PDM 制御レジスタ（SAI_PDMCR）

アドレスオフセット：0x0044

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
				rw	rw	rw	rw			rw	rw				rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **CKEN4**：ビットストリームクロック 4 番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK4 クロック無効

1：SAI_CK4 クロック有効

注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

SAI_CK4 は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3：SAI の実装を参照してください。

ビット 10 **CKEN3**：ビットストリームクロック 3 番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK3 クロック無効

1：SAI_CK3 クロック有効

注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

SAI_CK3 は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3：SAI の実装を参照してください。

ビット 9 **CKEN2**：ビットストリームクロック 2 番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK2 クロック無効

1：SAI_CK2 クロック有効

注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

SAI_CK2 は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3：SAI の実装を参照してください。

ビット 8 **CKEN1**：ビットストリームクロック 1 番のクロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。

0：SAI_CK1 クロック無効

1：SAI_CK1 クロック有効

注： このビットを PDMEN = 1 のときに設定するのは推奨しません。

SAI_CK1 は、すべての SAI インスタンスで使用できるわけではありません。詳細については、セクション 40.3：SAI の実装を参照してください。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **MICNBR[1:0]**：マイクロフォンの本数

このビットは、ソフトウェアによってセット／クリアされます。

00：マイクロフォン 2 本の設定

01：マイクロフォン 4 本の設定

10：マイクロフォン 6 本の設定

11：マイクロフォン 8 本の設定

注： このフィールドを PDMEN = 1 のときに設定するのは推奨しません。*

すべての SAI インスタンスで全データ･ラインを使用できるわけではありません。詳細については、セクション 40.3：SAI の実装を参照してください。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **PDMEN**：PDM 有効化

このビットは、ソフトウェアによってセット／クリアされます。このビットを使って、PDM インタフェース･ブロックの状態を制御することができます。

PDM インタフェースを有効にする前に SAI がすでに TDM マスタ･モードで動作していることを確認してください。

0：PDM インタフェースが無効です。

1：PDM インタフェースが有効です。

40.6.18 SAI PDM ディレイレジスタ（SAI_PDMDLY）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30:28 **DLYM4R[2:0]**：**ペア No.4** の 2 番目のマイクロフォンのディレイライン

このビットは、ソフトウェアによってセット／クリアされます。

000：遅延なし

001：1 T_{SAI_CK} 周期のディレイ

010：2 T_{SAI_CK} 周期のディレイ

.....

111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D4 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:24 **DLYM4L[2:0]**：ペア No.4 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D4 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 23 予約済みであり、リセット値に保持する必要があります。

ビット 22:20 **DLYM3R[2:0]**：ペア No.3 の 2番目のマイクロフォンのディレイ･ライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D3 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18:16 **DLYM3L[2:0]**：ペア No.3 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D3 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **DLYM2R[2:0]**：ペア No.2 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注： このフィールドは、D2 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **DLYM2L[2:0]**：ペア No.2 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D2 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **DLYM1R[2:0]**：ペア No.1 の 2 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D1 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **DLYM1L[2:0]**：ペア No.1 の 1 番目のマイクロフォンのディレイライン
このビットは、ソフトウェアによってセット／クリアされます。
000：遅延なし
001：1 T_{SAI_CK} 周期のディレイ
010：2 T_{SAI_CK} 周期のディレイ
.....
111：7 T_{SAI_CK} 周期のディレイ

このフィールドは動作中に変更することができます。

注：　このフィールドは、D1 ラインが使用できる場合にのみ使用できます。使用の可否は セクション 40.3 : SAI の実装を参照してください。

40.6.19 SAI レジスタ･マップ

表 376. SAI レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	予約済みです。																																
0x0004 または 0x0024	SAI_xCR1	Res.	Res.	Res.	Res.	MCKEN	OSR	MCKDIV[5:0]						NODIV	Res.	DMAEN	SAIEN	Res.	Res.	OUTDRIV	MONO	SYNCEN[1:0]		CKSTR	LSBFIRST	DSI[2:0]			Res.	PRTCFG[1:0]		MODE[1:0]	
	リセット値					0	0	0	0	0	0	0	0	0		0	0			0	0	0	0	0	0	0	1	0		0	0	0	0
0x0008 または 0x0028	SAI_xCR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	COMP[1:0]		CPL	MUTECN[5:0]						MUTE VAL	MUTE	TRIS	FFLUS	FTH[2:0]		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x000C または 0x002C	SAI_xFRCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FSOFF	FSPOL	FSDEF	Res.	FSALL[6:0]							FRL[7:0]							
	リセット値														0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
0x0010 または 0x0030	SAI_xSLOTR	SLOTEN[15:0]																Res.	Res.	Res.	Res.	NBSLOT[3:0]				SLOTSZ[1:0]		Res.	FBOFF[4:0]				
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					0	0	0	0	0	0		0	0	0	0	0
0x0014 または 0x0034	SAI_xIM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDETIE	AFSDETIE	CNRDYIE	FREQIE	WCKCFGIE	MUTEDETIE	OVRUDRIE
	リセット値																										0	0	0	0	0	0	0
0x0018 または 0x0038	SAI_xSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLVL[2:0]			Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	LFSDET	AFSDET	CNRDY	FREQ	WCKCFG	MUTEDET	OVRUDR
	リセット値														0	0	0										0	0	0	0	1	0	0
0x001C または 0x003C	SAI_xCLRFR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CLFSDET	CAFSDET	CCNRDY	Res.	CWCKCFG	CMUTEDET	COVRUDR
	リセット値																										0	0	0		0	0	0
0x0020 または 0x0040	SAI_xDR	DATA[31:0]																															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	SAI_PDMCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CKEN4	CKEN3	CKEN2	CKEN1	Res.	Res.	MICNBR[1:0]		Res.	Res.	Res.	PDMEN
	リセット値																					0	0	0	0			0	0				0
0x0048	SAI_PDMDLY	Res.	DLYM4R [2:0]			Res.	DLYM4L [2:0]			Res.	DLYM3R [2:0]			Res.	DLYM3L [2:0]			Res.	DLYM2R [2:0]			Res.	DLYM2L [2:0]			Res.	DLYM1R [2:0]			Res.	DLYM1L [2:0]		
	リセット値		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0		0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

41 I²C（Inter-integrated circuit）インタフェース

41.1 概要

I²C（Inter-integrated circuit）バスインタフェースは、マイクロコントローラとシリアル I²C バス間の通信を処理します。マルチマスタ機能を備え、すべての I²C バス固有のシーケンシング、プロトコル、アービトレーション、およびタイミングを制御します。標準モード（Sm）、高速モード（Fm）、および高速モードプラス（Fm+）をサポートします。

また、SMBus（System Management Bus）および PMBus（Power Management Bus）と互換性があります。

DMA を使用して、CPU の負荷を軽減できます。

41.2 I²C の主な機能

- I²C バス仕様 rev03 との互換性：
  - スレーブおよびマスタモード
  - マルチマスタ機能
  - 標準モード（最大 100 kHz）
  - 高速モード（最大 400 kHz）
  - 高速モードプラス（最大 1 MHz）
  - 7 ビットおよび 10 ビットアドレッシングモード
  - 複数の 7 ビットスレーブアドレス（2 つのアドレス、1 つは設定可能なマスク付き）
  - すべての 7 ビットアドレス確認応答モード
  - 同報（General call）コール
  - プログラム可能なセットアップおよびホールド時間
  - 使いやすいイベント管理
  - クロックストレッチオプション
  - ソフトウェアリセット
- DMA 機能付きの 1 バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます（セクション 41.3：I2C の実装を参照）：

- SMBus 仕様 rev 3.0 との互換性：
  - ハードウェア PEC（Packet Error Checking）の生成と ACK 制御による確認
  - コマンドおよびデータ確認応答制御
  - アドレス解決プロトコル（ARP）サポート
  - ホストおよびデバイスのサポート
  - SMBus アラート
  - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.3 標準との互換性
- 独立したクロック：独立したクロックソースの選択により、I²C の通信速度は PCLK の再プログラミングから独立
- アドレス一致時に STOP モードからウェイクアップ

41.3 I²C の実装

このマニュアルでは、I²C ペリフェラルに実装されているすべての機能について説明しています。STM32G4 シリーズデバイスでは、I2C1、I2C2、I2C3、および I2C4 は、次の表に示すように、すべての機能を実装しています。

表 377. I²C の実装

I2C の機能(1)	I2C1	I2C2	I2C3	I2C4
7 ビット･アドレス･モード	X	X	X	X
10 ビットアドレスモード	X	X	X	X
標準モード（最大 100 kbit/s）	X	X	X	X
高速モード（最大 400 kbit/s）	X	X	X	X
20mA 出力駆動 I/O 搭載高速モードプラス（最大 1 Mbit/s）	X	X	X	X
独立クロック	X	X	X	X
STOP 1 モードからのウェイクアップ	X	X	X	X
SMBus/PMBus	X	X	X	X

1. X : サポートされています。

41.4 I²C の機能詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によって I²C バスに接続されます。標準（最大 100 kHz）、高速モード（最大 400 kHz）、または高速モードプラス（最大 1 MHz）の I²C バスで接続できます。

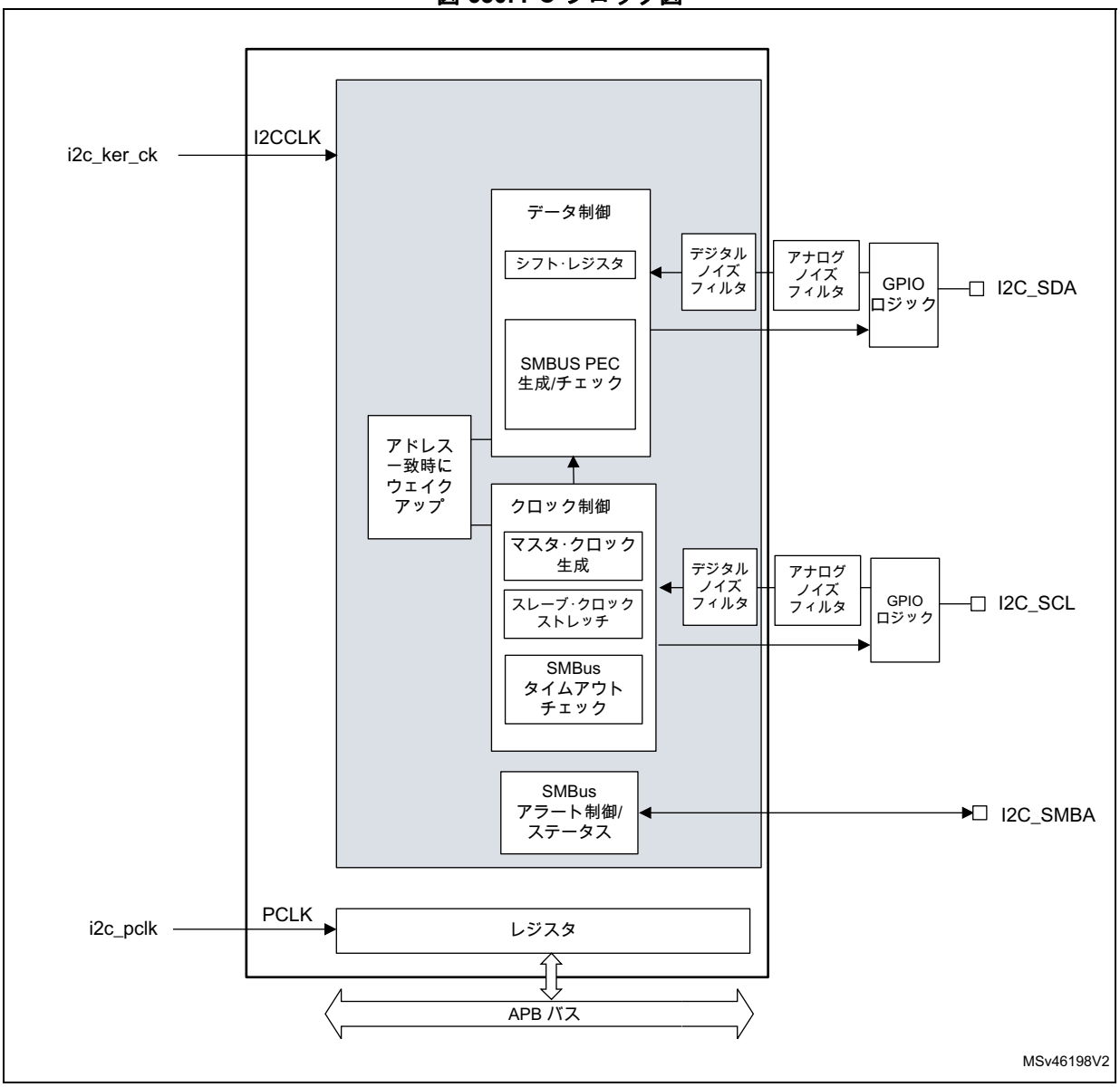
このインタフェースは、データピン（SDA）とクロックピン（SCL）によって SMBus に接続することもできます。

SMBus 機能がサポートされる場合、追加の SMBus アラートピン（SMBA）オプションも使用できます。

41.4.1 I²C ブロック図

I²C インタフェースのブロック図を図 630に示します。

図 630. I²C ブロック図



I²C は、独立したクロック·ソースによってクロック供給されるため PCLK 周波数から独立して動作できます。

高速モード·プラス動作のための 20 mA 出力電流駆動をサポートする I²C I/O では、駆動能力をシステム設定コントローラ（SYSCFG）の制御ビットで有効にします。セクション 41.3 : I2C の実装を参照してください。

41.4.2 I²C ピンおよび内部信号

表 378. I²C の入出力ピン

ピン名	信号タイプ	説明
I2C_SDA	双方向	I²C データ
I2C_SCL	双方向	I²C クロック
I2C_SMBA	双方向	SMBus アラート

表 379. I²C 内部入力／出力信号

内部信号名	信号タイプ	説明
i2c_ker_ck	入力	I²C カーネルクロック（本書では I2CCLK とも言います）
i2c_pclk	入力	I²C APB クロック
i2c_it	出力	I²C 割込み。割込みソースの全リストは、表 393 : I2C 割込みリクエスト を参照してください。
i2c_rx_dma	出力	I²C 受信データ DMA リクエスト（I2C_RX）
i2c_tx_dma	出力	I²C 送信データ DMA リクエスト（I2C_TX）

41.4.3 I²C クロックの要件

I²C カーネルは I2CCLKによってクロック供給されます。

I2CCLK の周期 t_{I2CCLK} は、次の条件を満たす必要があります。

$t_{I2CCLK} < (t_{LOW} - t_{filters}) / 4$ および $t_{I2CCLK} < t_{HIGH}$

ここで：

t_{LOW}：SCL ロー時間、および t_{HIGH}：SCL ハイ時間

$t_{filters}$：有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

アナログフィルタの遅延は、最大 260 ns です。デジタルフィルタの遅延は、DNF x t_{I2CCLK} です。

PCLK のクロック周期 t_{PCLK} は、次の条件を満たす必要があります。

$t_{PCLK} < 4/3\ t_{SCL}$

t_{SCL}：SCL 周期

注意： **I²C カーネルが PCLKによってクロック供給されるとき、このクロックは t_{I2CCLK} の条件を満たす必要があります。**

41.4.4 モード選択

このインタフェースは、次の 4 つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、START コンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失または STOP 生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

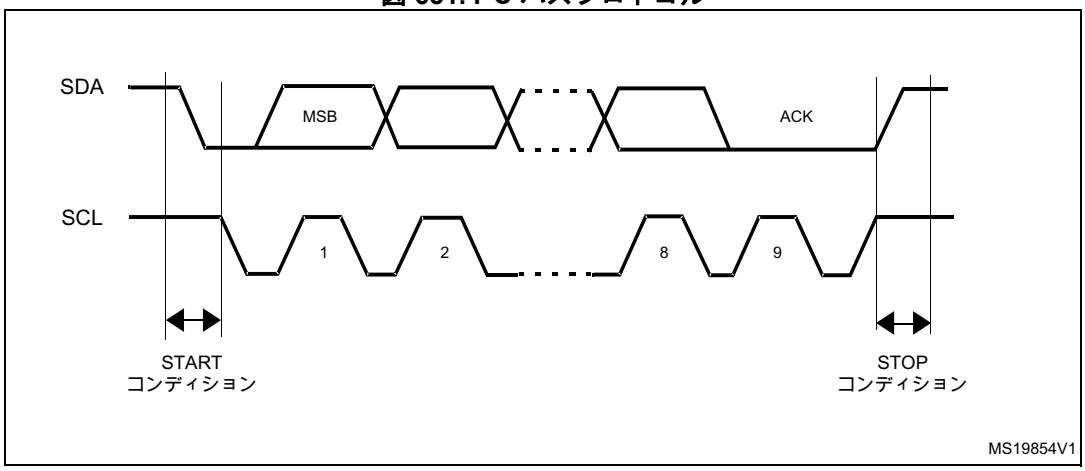
マスタモードでは、I²C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常に START コンディションで開始され、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

スレーブ·モードでは、このインタフェースは、自己アドレス（7 または 10 ビット）と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みの SMBus アドレスもソフトウェアによって有効にできます。

データとアドレスは、MSB ファーストの 8 ビットバイトとして転送されます。START コンディションの後に続く最初のバイト（7 ビットモードでは 1 バイト、10 ビットモードでは 2 バイト）にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8 クロック·サイクルのバイト転送の後には 9 番目のクロック·パルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります。次の図を参照してください。

図 631. I²C バスプロトコル



確認応答（Acknowledge）は、ソフトウェアによって有効または無効にできます。I²C インタフェースのアドレスは、ソフトウェアによって選択できます。

41.4.5 I²C の初期化

ペリフェラルの有効化と無効化

I²C ペリフェラルクロックは、クロックコントローラで設定し、有効にする必要があります。

そして、I2C_CR1 レジスタの PE ビットをセットすることによって、I²C を有効にできます。

I²C が無効なときに（PE = 0）、I²C はソフトウェア･リセットを実行します。詳細については、セクション 41.4.6：ソフトウェア･リセットを参照してください。

ノイズフィルタ

I2C_CR1 レジスタの PE ビットをセットすることによって I2C ペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDA および SCL 入力にアナログノイズフィルタがあります。このアナログフィルタは I²C 仕様に準拠しており、高速モードおよび高速モードプラスで最大 50 ns のパルス幅を持つスパイクを抑制します。ANFOFF ビットをセットすることによって、このアナログフィルタを無効にし、I2C_CR1 レジスタの DNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCL または SDA ラインのレベルは、DNF x I2CCLK 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な 1 ~ 15 I2CCLK 周期の長さを持つスパイクを抑制できます。

表 380. アナログ･フィルタとデジタル･フィルタの比較

-	アナログフィルタ	デジタルフィルタ
抑制されるスパイクのパルス幅	≥ 50 ns	長さを 1 ～ 15 I2C ペリフェラルクロックにプログラム可能
利点	STOP モードで使用可能	– プログラム可能な長さ：追加のフィルタリング機能対標準要件 – 安定した長さ
欠点	温度、電圧、プロセスのばらつき	デジタルフィルタが有効なときには、アドレス一致時の STOP モードからのウェイクアップは使用できない

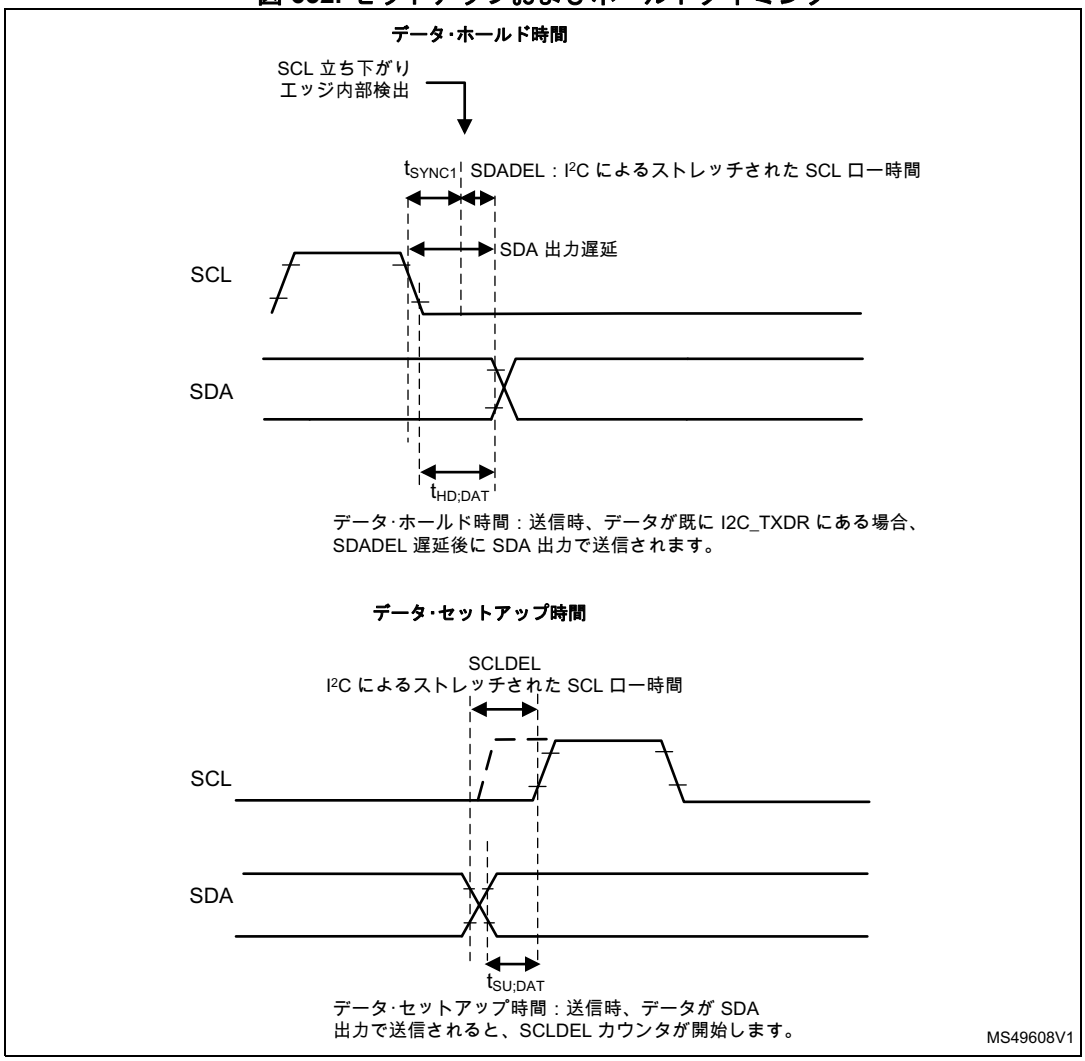
注意： **I²C が有効なときには、フィルタ構成の変更はできません。**

I²C のタイミング

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間が使用されるのを保証するためには、タイミングを設定する必要があります。これを行うには、I2C_TIMINGR レジスタの PRESC[3:0]、SCLDEL[3:0]、および SDADEL[3:0] ビットをプログラムします。

STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

図 632. セットアップおよびホールドタイミング



- SCL 立ち下がりエッジが内部で検出されると、SDA 出力を送信する前に遅延が挿入されます。この遅延は、t_{SDADEL} = SDADEL x t_{PRESC} + t_{I2CCLK} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  T_{SDADEL} はホールド時間 $t_{HD;DAT}$ に影響を与えます。

SDA 出力遅延の合計は、次のとおりです：

t_{SYNC1} + {[SDADEL x (PRESC+1) + 1] x t_{I2CCLK} }

t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- アナログフィルタが有効なときの入力遅延：$t_{AF(min)} < t_{AF} < t_{AF(max)}$
- デジタルフィルタが有効なときの入力遅延：t_{DNF} = DNF x t_{I2CCLK}
- SCL と I2CCLK クロックの同期による遅延（2 ～ 3 I2CCLK 周期）

SCL 立ち下がりエッジの未定義の領域をブリッジするためには、SDADEL を次のようにプログラムする必要があります：

{$t_{f\ (max)}$ +$t_{HD;DAT\ (min)}$ -$t_{AF(min)}$ - [(*DNF* +3) x t_{I2CCLK}]} / {(PRESC +1) x t_{I2CCLK} } ≤ SDADEL

SDADEL ≤ {$t_{HD;DAT\ (max)}$ -$t_{AF(max)}$ - [(*DNF+4) x* t_{I2CCLK}]} / {(PRESC +1) x t_{I2CCLK} }

注： **$t_{AF(min)}$ / $t_{AF(max)}$ は、アナログフィルタが有効なときのみ、等式に含まれます。t_{AF} の値については、デバイスのデータシートを参照してください。**

最大 $t_{HD;DAT}$ は、標準モード、高速モード、および高速モードプラスで 3.45 µs、0.9 µs、および 0.45 µs ですが、遷移時間による $t_{VD;DAT}$ の最大値より短い必要があります。この最大値を満たす必要があるのは、デバイスが SCL 信号の LOW 周期（t_{LOW}）をストレッチしない場合だけです。クロックが SCL をストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

SDA 立ち上がりエッジは、通常、最悪ケースであり、この場合、前の等式は次のようになります：

SDADEL ≤ {$t_{VD;DAT\ (max)}$ - $t_{r\ (max)}$ - *260 ns* - [(*DNF+4) x* t_{I2CCLK}]} / {(PRESC +1) x t_{I2CCLK} }.

注： **NOSTRETCH = 0 のときには、SCLDEL の値に従って、デバイスはセットアップ時間を保証するために SCL ローをストレッチするので、この条件に違反することがあります。**

t_f、t_r、$t_{HD;DAT}$、および $t_{VD;DAT}$ の標準値については、表 381：I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください。

- t_{SDADEL} 遅延の後や、データが I2C_TXDR レジスタにまだ書き込まれていないためスレーブがクロックをストレッチしなければならなかった場合の SDA 出力送信後、SCL ラインはセットアップ時間中、ローレベルで保持されます。このセットアップ時間は、t_{SCLDEL} = (SCLDEL+1) x t_{PRESC} であり t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLDEL} は、セットアップ時間 $t_{SU;DAT}$ に影響を与えます。

SDA 遷移（立ち上がりエッジは通常、最悪のケース）の未定義の領域をブリッジするためには、SCLDEL を次のようにプログラムする必要があります：

{[$t_{r\ (max)}$ + $t_{SU;DAT\ (min)}$] / [(PRESC+1)] x t_{I2CCLK}]} - 1 ≤ SCLDEL

t_r および $t_{SU;DAT}$ の標準値については、表 381：I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください。

使用される SDA および SCL 遷移時間の値は、アプリケーションの値です。標準から最大値を使用すると、SDADEL と SCLDEL の計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

注： **各クロックパルスで、SCL 立ち下がりエッジの検出後、I2C マスタまたはスレーブは、最低 [(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、送信および受信の両モードで SCL ローを**

ストレッチします。送信モードで、SDADEL カウンタ終了時にデータがまだ I2C_TXDR に書き込まれていない場合、I²C は次のデータが書き込まれるまで SCL ローをストレッチし続けます。その時、新しいデータ MSB が SDA 出力で送信され、SCLDEL カウンタが開始し、SCL ローのストレッチを継続して、データセットアップ時間を保証します。

スレーブ·モードで NOSTRETCH = 1 の場合、SCL はストレッチされません。したがって、十分なセットアップ時間を確実に確保するために、SDADEL をこのようにプログラムする必要があります。

表 381. I²C-SMBus 仕様のデータのセットアップおよびホールド時間

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス（Fm+）		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
$t_{HD;DAT}$	データホールド時間	0	-	0	-	0	-	0.3	-	µs
$t_{VD;DAT}$	データ有効時間	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	データセットアップ時間	250	-	100	-	50	-	250	-	ns
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	

また、マスタモードでは、I2C_TIMINGR レジスタの PRESC[3:0]、SCLH[7:0]、および SCLL[7:0] ビットをプログラムすることによって、SCL クロックのハイおよびローレベルを設定する必要があります。

- SCL 立ち下がりエッジが内部で検出されると、SCL 出力をリリースする前に遅延が挿入されます。この遅延は、t_{SCLL} = (SCLL+1) x t_{PRESC} であり、t_{PRESC} = (PRESC+1) x t_{I2CCLK} です。
  t_{SCLL} は、SCL ロー時間 t_{LOW} に影響を与えます。
- SCL 立ち上がりエッジが内部で検出されると、SCL 出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、t_{SCLH} =（SCLH+1）x t_{PRESC} であり、t_{PRESC} =（PRESC+1）x t_{I2CCLK} です。t_{SCLH} は SCL ハイ時間 t_{HIGH} に影響を与えます。

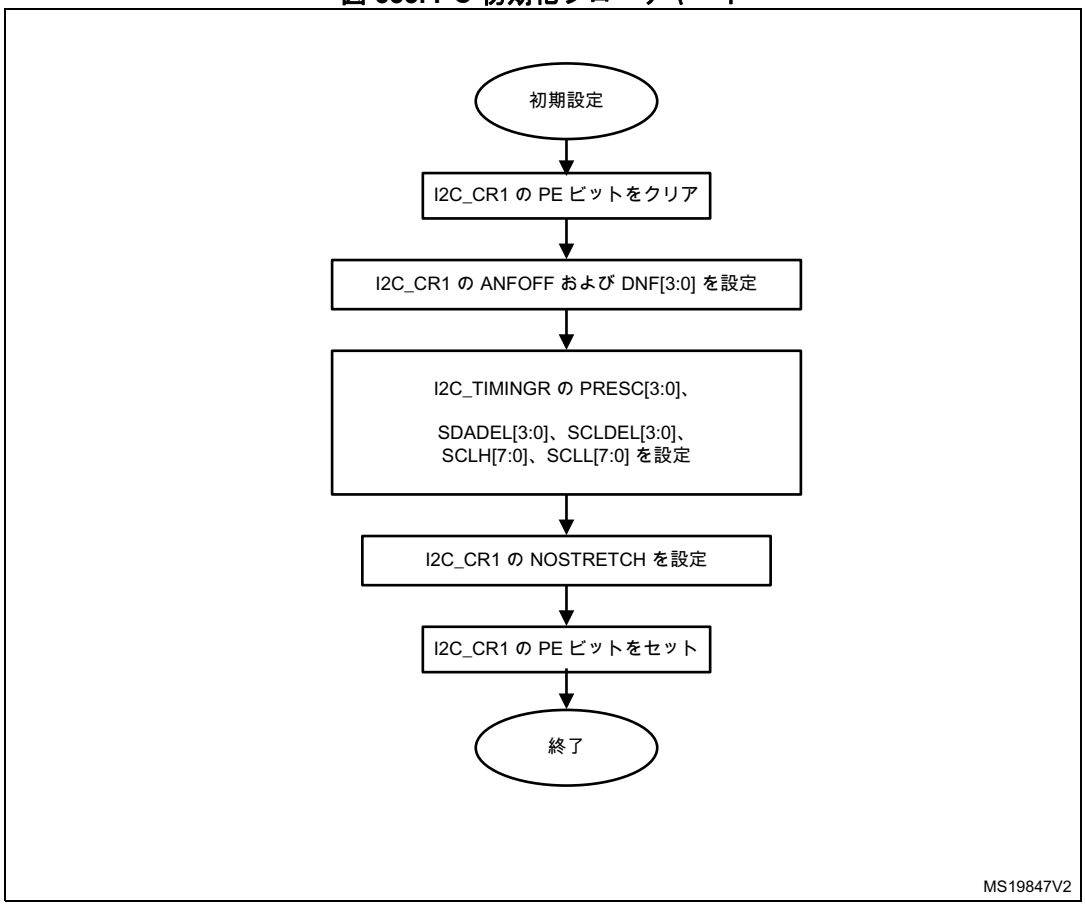
詳細については、I2C マスタ初期化を参照してください。

注意： **I²C が有効なときには、タイミング構成の設定はできません。**

ペリフェラルを有効にする前に、I2C スレーブ NOSTRETCH モードも設定する必要があります。詳細については、I2C スレーブ初期化を参照してください。

注意： **I²C が有効なときには、NOSTRETCH 構成の変更はできません。**

図 633. I2C 初期化フローチャート



41.4.6 ソフトウェア･リセット

ソフトウェアリセットを行うには、I2C_CR1 レジスタの PE ビットをクリアします。その場合、I2C のライン SCL および SDA がリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2C_CR2 レジスタ：START、STOP、NACK
2. I2C_ISR レジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus 機能がサポートされるときには、以下も影響を受けます：

1. I2C_CR2 レジスタ：PECBYTE
2. I2C_ISR レジスタ：PECERR、TIMEOUT、ALERT

ソフトウェア･リセットを実行するには、少なくとも 3 APB クロック･サイクルの間、PE をローに保持する必要があります。そのためには、次のソフトウェア･シーケンスを記述します。

1. PE = 0 を書き込む。
2. PE = 0 であることを確認する。
3. PE = 1 を書き込む。

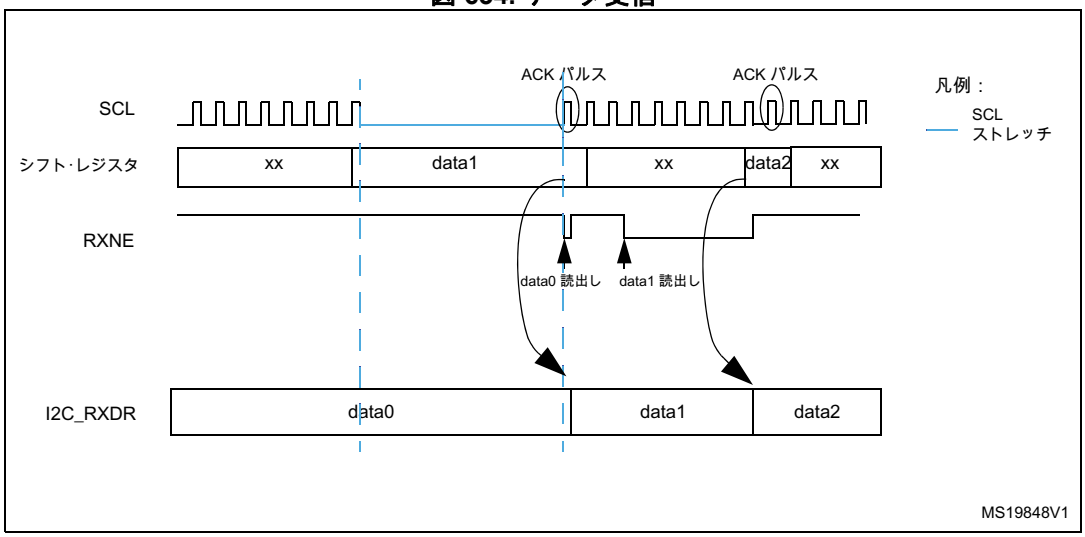
41.4.7 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA 入力はシフトレジスタに送られます。8 番目の SCL パルスの後（完全なデータ･バイトの受信後）、I2C_RXDR レジスタが空の場合（RXNE = 0）、このレジスタにシフト･レジスタがコピーされます。RXNE = 1 の場合、すなわち、前に受信されたデータ･バイトがまだ読み出されていなかった場合、I2C_RXDR が読み出されるまで SCL ラインはロー･レベルでストレッチされます。このストレッチは、8 番目と 9 番目の SCL パルスの間（確認応答パルスの前）に挿入されます。

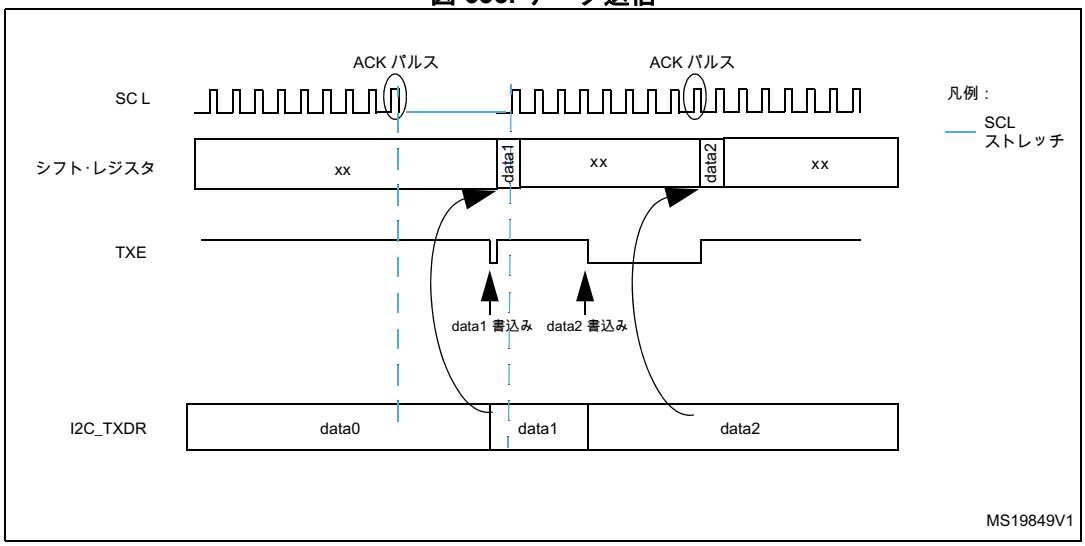
図 634. データ受信



送信

I2C_TXDR レジスタが空でない場合（TXE = 0）、その内容が 9 番目の SCL パルス（確認応答パルス）の後、シフト･レジスタにコピーされます。次に、シフト･レジスタの内容が SDA ラインにシフト･アウトされます。TXE = 1 の場合、すなわち、I2C_TXDR にデータがまだ書き込まれていない場合、SCL ラインは I2C_TXDR が書き込まれるまでロー･レベルでストレッチされます。このストレッチは、9 番目の SCL パルスの後で行われます。

図 635. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2C にはハードウェアにバイトカウンタが組み込まれています。

- マスタモードでの NACK、STOP、および ReSTART 生成
- スレーブレシーバモードでの ACK 制御
- SMBus 機能がサポートされているときの PEC 生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2C_CR2 レジスタの SBC（スレーブバイト制御）ビットをセットすることによって、ソフトウェアにより有効にできます。

転送されるバイト数は、I2C_CR2 レジスタの NBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数（NBYTES）が 255 より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTES でプログラムされたバイト数が転送されると、TCR フラグがセットされ、TCIE がセットされている場合は割込みが生成されます。SCL は、TCR フラグがセットされている間、ストレッチされます。TCR は、NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTES カウンタに最後のバイト数が再ロードされたときには、RELOAD ビットがクリアされる必要があります。

マスタモードで RELOAD=0 のときには、カウンタは 2 つのモードで使用できます：

- **自動終了モード**（I2C_CR2 レジスタの AUTOEND = “1”）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタは STOP コンディションを自動的に送信します。
- **ソフトウェア終了モード**（I2C_CR2 レジスタの AUTOEND = 0）。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェア･アクションが求められます。TC フラグがセットされ、TCIE ビットがセットされている場合は割込みが生成されます。SCL 信号は、TC フラグがセットされている間、ストレッチされます。TC フラグは、I2C_CR2 レジスタの START または STOP ビットがセットされたときに、ソフトウェアによってクリアされます。マスタが RESTART コンディションを送信したいときには、このモードを使用する必要があります。

注意： **AUTOEND ビットは、RELOAD ビットがセットされているときには効果がありません。**

表 382. I²C 設定

機能	SBC ビット	RELOAD ビット	AUTOEND ビット
マスタ Tx/Rx NBYTES + STOP	x	0	1
マスタ Tx/Rx + NBYTES + RESTART	x	0	0
スレーブ Tx/Rx すべての受信バイトに ACK	0	x	x
スレーブ Rx および ACK 制御	1	1	x

41.4.8 I²C スレーブモード

I²C スレーブ初期化

スレーブモードで動作するには、少なくとも 1 つのスレーブアドレスを有効にする必要があります。2 つのレジスタ I2C_OAR1 と I2C_OAR2 を使用して、スレーブ専用アドレス OA1 および OA2 をプログラムできます。

- OA1 は、I2C_OAR1 レジスタの OA1MODE ビットをセットすることによって、7 ビットモード（デフォルト）または 10 ビットアドレッシングモードに設定できます。

  OA1 を有効にするには、I2C_OAR1 レジスタの OA1EN ビットをセットします。
- 追加のスレーブ･アドレスが必要な場合は、2 番目のスレーブ･アドレス OA2 を設定できます。I2C_OAR2 レジスタの OA2MSK[2:0] ビットを設定することによって、最大 7 つの OA2 LSB をマスクできます。したがって、OA2MSK が 1 から 6 まで設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、または OA2[7] のみが受信アドレスと比較されます。OA2MSK が 0 に等しくなくなるとすぐに、OA2 のアドレスコンパレータは、確認応答されない I²C 予約済みアドレス（0000 XXX および 1111 XXX）を除外します。OA2MSK=7 の場合、受信されたすべてのアドレスが確認応答されます（予約済みアドレスを除く）。OA2 は常に 7 ビットアドレスです。

  これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2C_OAR1 または I2C_OAR2 レジスタが OA2MSK=0 でプログラムされた場合、確認応答できます。

  OA2 を有効にするには、I2C_OAR2 レジスタの OA2EN ビットをセットします。
- 同報アドレスは、I2C_CR1 レジスタの GCEN ビットをセットすることで有効になります。

I²C がその有効アドレスの 1 つによって選択されると、ADDR 割込みステータス･フラグがセットされ、ADDRIE ビットがセットされている場合は割込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用し、必要なときには、ソフトウェアアクションを実行するために、SCL 信号をローレベルでストレッチすることを意味します。マスタがクロックのストレッチをサポートしていない場合、I2C_CR1 レジスタで NOSTRETCH = 1 を設定して I²C を設定する必要があります。

ADDR 割込みの受信後、いくつかのアドレスが有効な場合は、I2C_ISR レジスタの ADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIR フラグも確認する必要があります。

スレーブクロックストレッチ（NOSTRETCH = 0）

デフォルトモードでは、I2C スレーブは次の状況で SCL クロックをストレッチします :

- ADDR フラグがセットされると : 受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCF ビットをセットすることによりソフトウェアによって ADDR フラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了したときに、新しいデータが I2C_TXDR レジスタに書き込まれていない場合、または ADDR フラグがクリアされたときに、最初のデータ･バイトが書き込まれていない場合（TXE = 1）。このストレッチは、データが I2C_TXDR レジスタに書き込まれたときにリリースされます。
- 受信時、I2C_RXDR レジスタがまだ読み出されておらず、新しいデータ受信が完了したとき。このストレッチは、I2C_RXDR が読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード（SBC=1 および RELOAD=1）で TCR = 1 のとき、すなわち、最後データバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによって TCR がクリアされたときにリリースされます。
- SCL 立ち下がりエッジの検出後、I2C は、[(SDADEL+SCLDEL+1) x (PRESC+1) + 1] x t_{I2CCLK} の間、SCL ローをストレッチします。

クロックストレッチなしのスレーブ（NOSTRETCH = 1）

I2C_CR1 レジスタの NOSTRETCH = 1 のとき、I2C スレーブは SCL 信号をストレッチしません。

- ADDR フラグがセットされている間、SCL クロックはストレッチされません。
- 送信時、転送に対応する最初の SCL パルスが発生する前に、I2C_TXDR レジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2C_ISR レジスタで OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。OVR フラグは、最初のデータ送信が開始し、STOPF ビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後でのみ、前に転送の STOPF フラグをクリアすることによって、送信される最初のデータについても、OVR ステータスが提供されることを確実にできます。
- 受信時、次のデータ･バイトの 9 番目の SCL パルス（ACK パルス）が発生する前に、I2C_RXDR レジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

スレーブ·バイト制御モード

スレーブ受信モードでバイト ACK 制御を可能にするには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブ·バイト制御モードを有効にする必要があります。これは、SMBus 標準に準拠するために必要な措置です。

スレーブ受信モードでバイト ACK 制御を可能にするには、再ロード·モードを選択する必要があります（RELOAD = 1）。各バイトを制御できるようにするには、ADDR 割込みサブルーチンで NBYTES を 0x1 に初期化し、バイトを受信するたびに 0x1 に再ロードする必要があります。バイトが受信されると、TCR ビットがセットされ、8 番目と 9 番目の SCL パルスの間で SCL 信号がローでストレッチされます。I2C_RXDR レジスタからデータを読み出すことができ、その後、I2C_CR2 レジスタの ACK ビットを設定することによって、確認応答するかどうかを決定できます。SCL ストレッチは、NBYTES をゼロ以外の値にプログラムすることによってリリースされ、確認応答または非確認応答が送信され、次のバイトを受信できます。

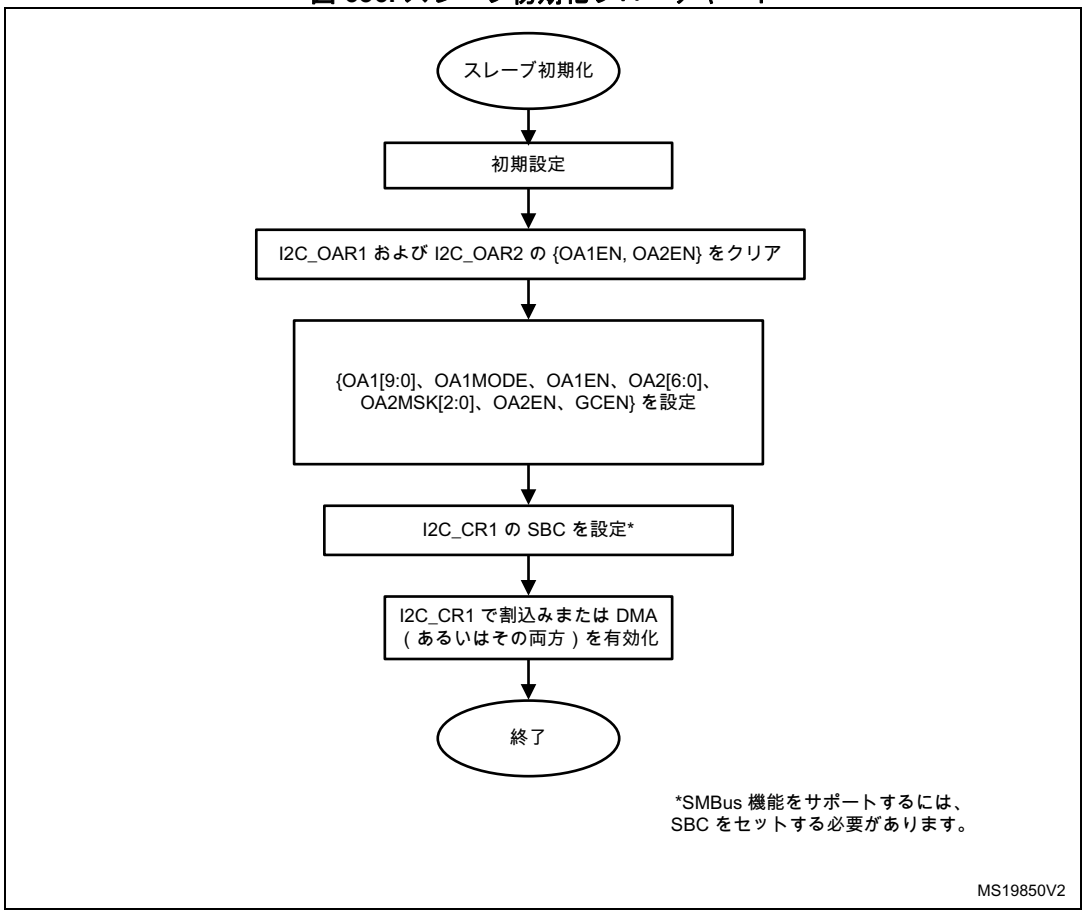
NBYTES に 0x1 より大きい値をロードでき、この場合、受信フローは NBYTES データ受信中、継続します。

注： **SBC ビットは、I²C が無効なとき、スレーブがアドレス指定されていないとき、または ADDR = 1 のときに設定する必要があります。**

RELOAD ビットの値は、ADDR = 1 のとき、または TCR = 1 のときに変更できます。

注意： **スレーブ·バイト制御モードは、NOSTRETCH モードと互換性がありません。NOSTRETCH = 1 のときに SBC をセットすることはできません。**

図 636. スレーブ初期化フローチャート



スレーブ･トランスミッタ

I2C_TXDR レジスタが空になると、送信割込みステータス（TXIS）が生成されます。I2C_CR1 レジスタの TXIE ビットがセットされている場合は、割込みが生成されます。

TXIS ビットは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACK が受信されると、I2C_ISR レジスタの NACKF ビットがセットされ、I2C_CR1 レジスタの NACKIE ビットがセットされていた場合は割込みが生成されます。マスタが STOP または RESTART コンディションを実行できるように、スレーブは SCL および SDA ラインを自動的にリリースします。TXIS ビットは、NACK 受信時にはセットされません。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR レジスタの STOPF フラグがセットされ、割込みが生成されます。ほとんどのアプリケーションでは、SBC は通常、0 にプログラムされます。この場合、スレーブ･アドレスが受信されたときに（ADDR = 1）、TXE = 0 であった場合、I2C_TXDR レジスタの内容を最初のデータ･バイトとして送信するか、新しいデータ･バイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタを一掃するかを選択できます。

スレーブ･バイト制御モード（SBC = 1）では、アドレス一致割込みサブルーチン（ADDR = 1）で NBYTES に送信バイト数をプログラムする必要があります。この場合、転送中の TXIS イベントの数は、NBYTES でプログラムされた値に相当します。

注意： **NOSTRETCH = 1 のとき、ADDR フラグがセットされている間は SCL クロックがストレッチされないので、最初のデータ･バイトをプログラムするために ADDR サブルーチンで I2C_TXDR レジスタの内容を一掃することはできません。最初に送信されるデータ･バイトは、I2C_TXDR レジスタで前もってプログラムされている必要があります。**

- このデータは、前の送信メッセージの最後の TXIS イベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2C_TXDR レジスタをFlashできます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPF ビットのクリアは、これらのアクションの後でのみ行う必要があります。

  最初のデータ送信が開始したときに STOPF がまだセットされていた場合、アンダーランエラーが生成されます（OVR フラグがセットされます）。

  TXIS イベントが必要な場合（送信割込みまたは送信 DMA リクエスト）、TXIS イベントを生成するには、TXE ビットに加えて TXIS ビットもセットする必要があります。

図 637. I²C スレーブ・トランスミッタの転送シーケンス・フローチャート（NOSTRETCH = 0）

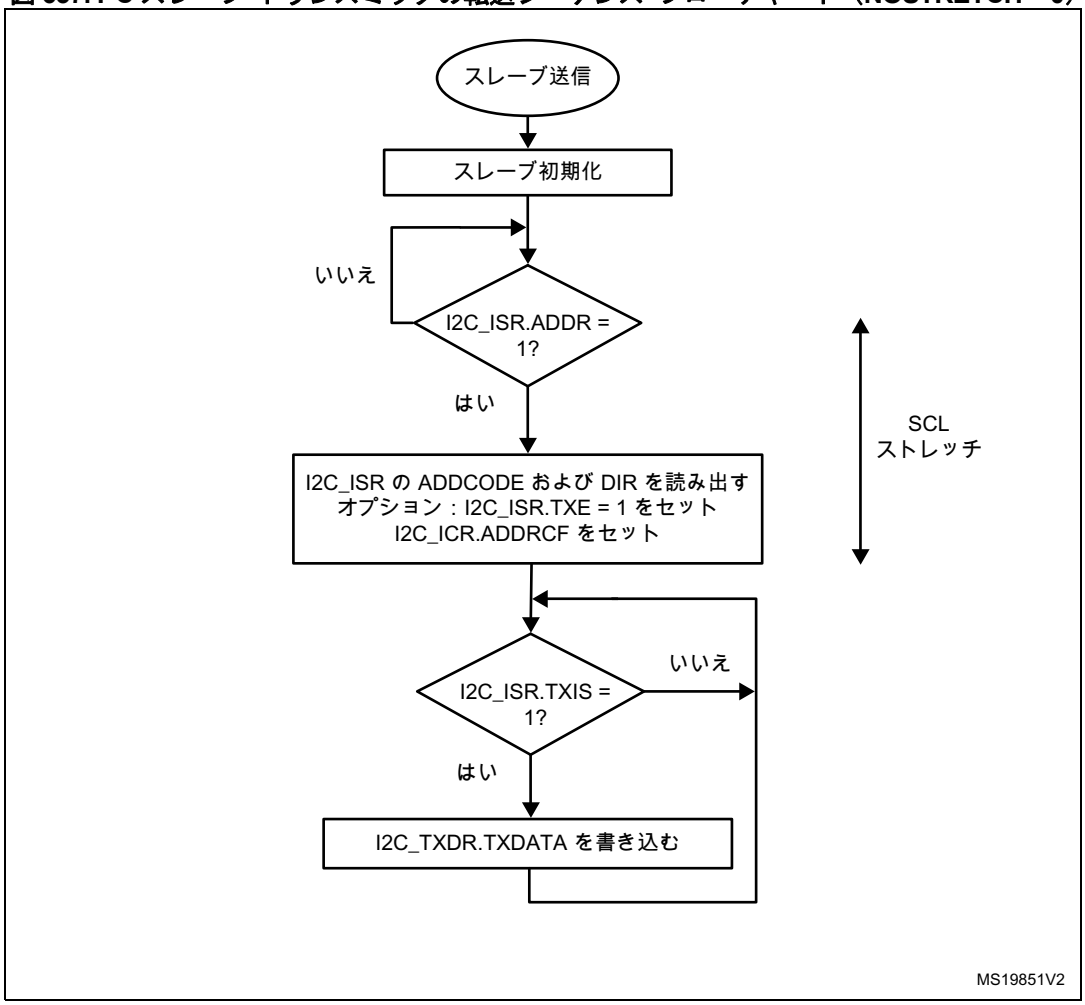


図 638. I2C スレーブ・トランスミッタの転送シーケンス・フローチャート（NOSTRETCH = 1）

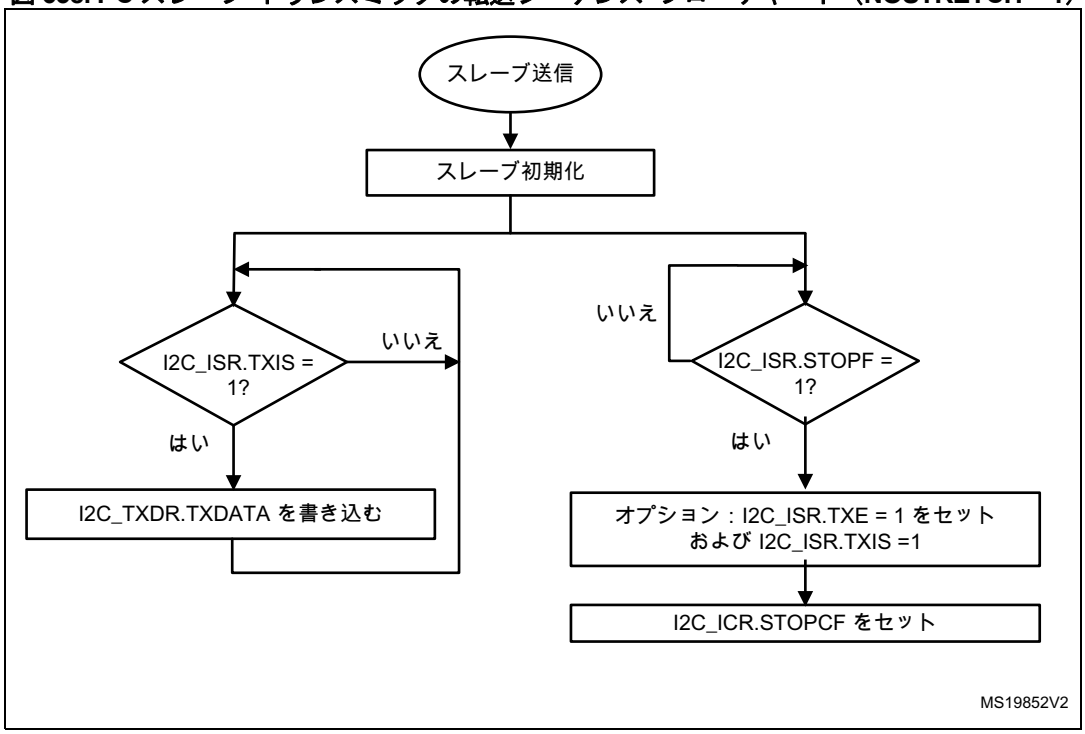
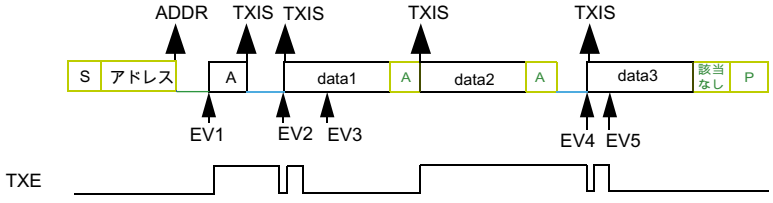


図 639. I²C スレーブ・トランスミッタの転送バス図

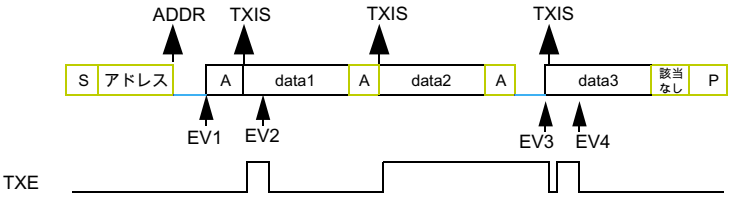
最初のデータを一掃する I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、TXE をセット、ADDRCF をセット
EV2：TXIS ISR：data1 を書き込む
EV3：TXIS ISR：data2 を書き込む
EV4：TXIS ISR：data3 を書き込む
EV5：TXIS ISR：data4 を書き込む（送信されない）

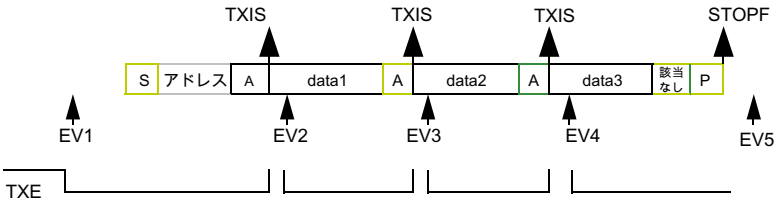
最初のデータをFlashしない I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=0 の例：



凡例：
送信
受信
SCL ストレッチ

EV1：ADDR ISR：ADDCODE および DIR をチェック、ADDRCF をセット
EV2：TXIS ISR：data2 を書き込む
EV3：TXIS ISR：data3 を書き込む
EV4：TXIS ISR：data4 を書き込む（送信されない）

I²C スレーブトランスミッタ 3 バイト、NOSTRETCH=1 の例：



凡例：
送信
受信
SCL ストレッチ

Ev1：data1 を書き込む
EV2：TXIS ISR：data2 を書き込む
EV3：TXIS ISR：data3 を書き込む
EV4：TXIS ISR：data4 を書き込む（送信されない）
EV5：STOPF ISR：（オプション：TXE および TXIS をセット）、STOPCF をセット

MS19853V2

スレーブレシーバ

I2C_RXDR がフルのときには、I2C_ISR の RXNE がセットされ、I2C_CR1 の RXIE がセットされている場合は割込みが生成されます。RXNE は、I2C_RXDR が読み出されたときにクリアされます。

STOP が受信され、I2C_CR1 レジスタの STOPIE ビットがセットされると、I2C_ISR の STOPF がセットされ、割込みが生成されます。

図 640. スレーブ･レシーバの転送シーケンス･フローチャート（NOSTRETCH = 0）

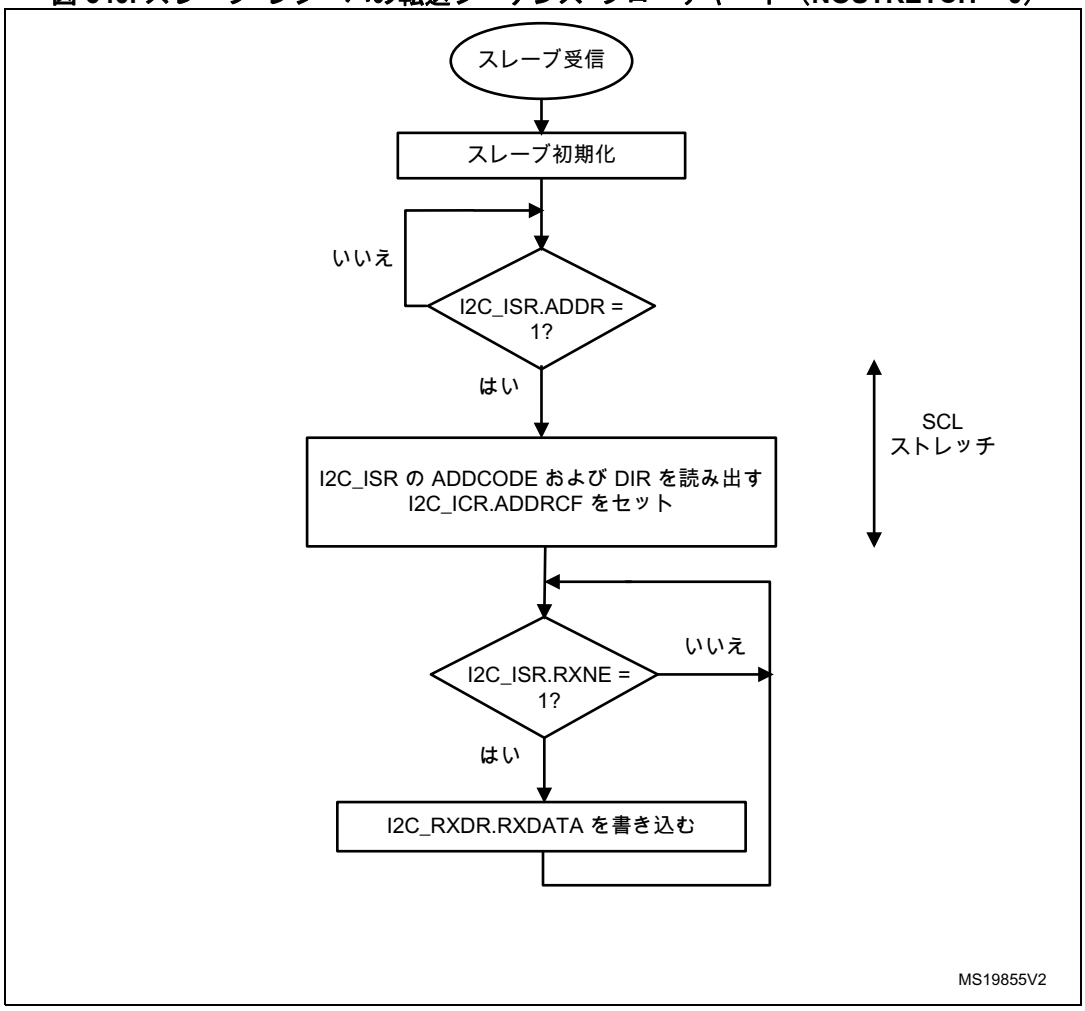


図 641. スレーブ･レシーバの転送シーケンス･フローチャート（NOSTRETCH = 1）

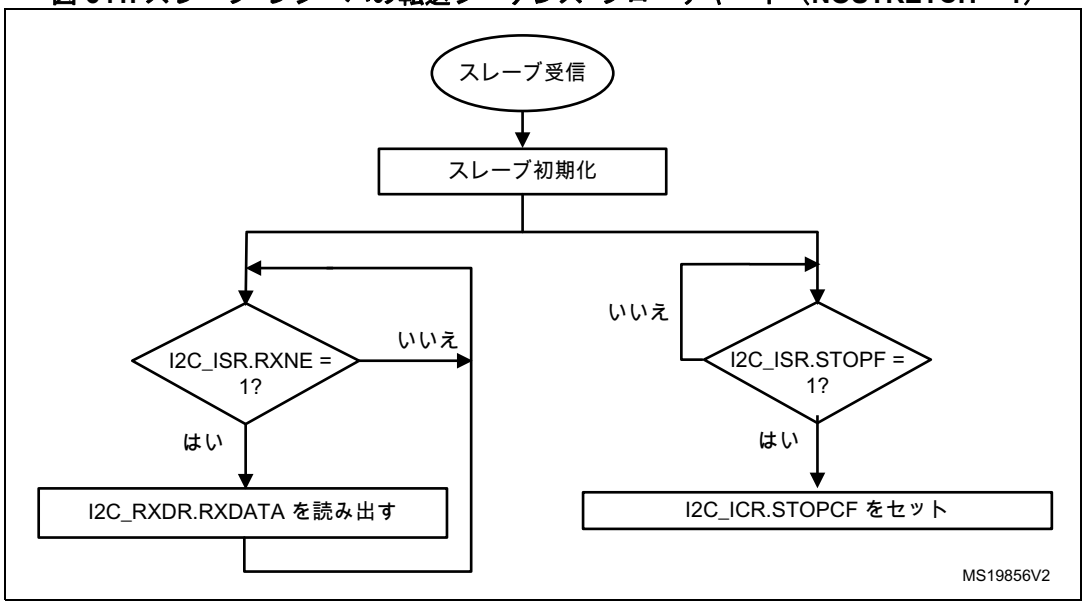
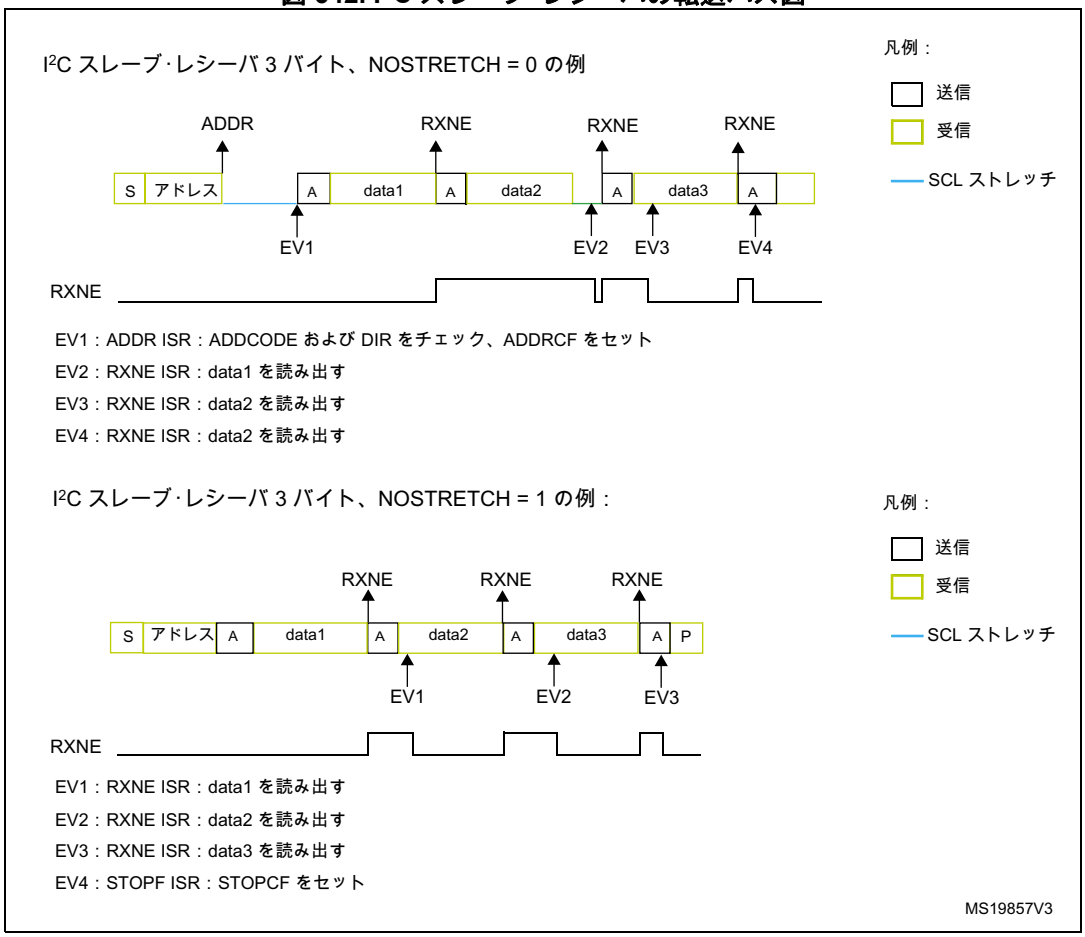


図 642. I²C スレーブ･レシーバの転送バス図



41.4.9 I²C マスタモード

I²C マスタ初期化

ペリフェラルを有効にする前に、I2C_TIMINGR レジスタの SCLH および SCLL ビットをセットすることによって、I2C マスタクロックを設定する必要があります。

STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルは SCLL カウンタを使用してカウントされ、SCL ローレベル内部検出から開始されます。
- クロックのハイレベルは SCLH カウンタを使用してカウントされ、SCL ハイレベル内部検出から開始されます。

I²C は、SCL 立ち下がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC1} の後に SCL ローレベルを検出します。SCLL カウンタが I2C_TIMINGR レジスタの SCLL[7:0] ビットでプログラムされた値に達すると、I²C は SCL をハイレベルにリリースします。

I²C は、SCL 立ち上がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC2} の後に SCL ハイレベルを検出します。SCLH カウンタが I2C_TIMINGR レジスタの SCLH[7:0] ビットでプログラムされた値に達すると、I2C は SCL をローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$t_{SCL} = t_{SYNC1} + t_{SYNC2} + \{[(SCLH+1) + (SCLL+1)] \times (PRESC+1) \times t_{I2CCLK}\}$

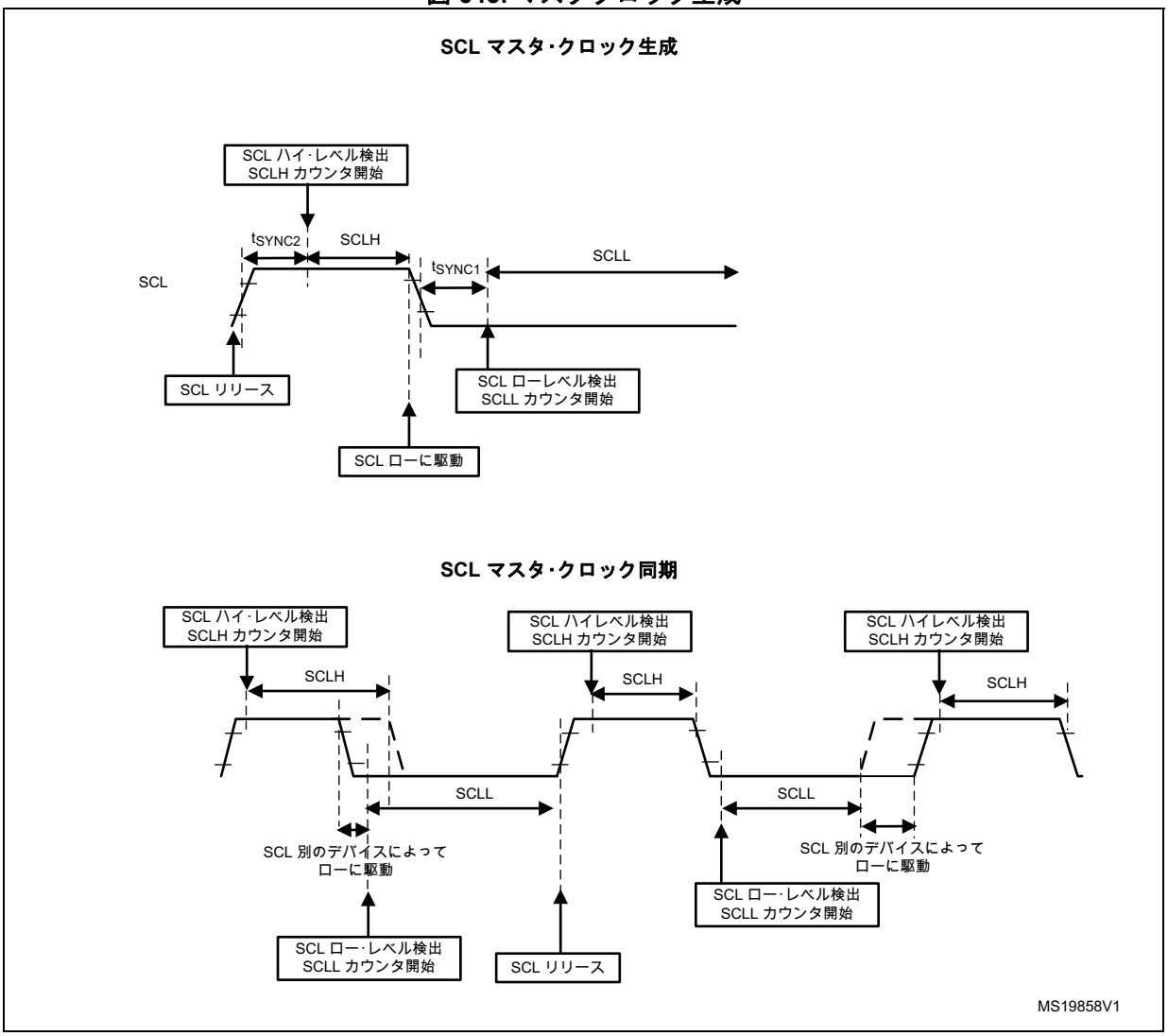
t_{SYNC1} の長さは、次のパラメータに依存します：

- SCL 立ち下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL 立ち上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF x t_{I2CCLK}
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

図 643. マスタクロック生成



注意： **I^2C または SMBus に準拠するには、マスタ·クロックが次のタイミングを満たす必要があります：**

表 383. I^2C-SMBus 仕様のクロック·タイミング

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス (Fm+)		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
f_{SCL}	SCL クロック周波数	-	100	-	400	-	1000	-	100	kHz
$t_{HD:STA}$	（反復）START コンディションのホールド時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
$t_{SU:STA}$	反復 START コンディションのセットアップ時間	4.7	-	0.6	-	0.26	-	4.7	-	µs
$t_{SU:STO}$	STOP コンディションのセットアップ時間	4.0	-	0.6	-	0.26	-	4.0	-	µs
t_{BUF}	STOP コンディションと START コンディションの間のバスフリー時間	4.7	-	1.3	-	0.5	-	4.7	-	µs

表 383. I²C-SMBus 仕様のクロック･タイミング （続き）

記号	パラメータ	標準モード（Sm）		高速モード（Fm）		高速モードプラス (Fm+)		SMBus		単位
		最小値	最大値	最小値	最大値	最小値	最大値	最小値	最大値	
t_{LOW}	SCL クロックのロー周期	4.7	-	1.3	-	0.5	-	4.7	-	µs
t_{HIGH}	SCL クロックの周期	4.0	-	0.6	-	0.26	-	4.0	50	µs
t_r	SDA および SCL 信号の立ち上がり時間	-	1000	-	300	-	120	-	1000	ns
t_f	SDA および SCL 信号の立ち下がり時間	-	300	-	300	-	120	-	300	ns

注： **SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングの生成にも使用されます。**

SCLH は、$t_{HD:STA}$ および $t_{SU:STO}$ タイミングの生成にも使用されます。

I2C_TIMINGR 設定と I2CCLK 周波数の例については、セクション 41.4.10 : I2C_TIMINGR レジスタの設定例を参照してください。

マスタ通信の初期化（アドレスフェーズ）

通信を初期化するためには､I2C_CR2 レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります。

- アドレッシングモード（7 ビットまたは 10 ビット）: ADD10
- 送信されるスレーブ･アドレス : SADD[9:0]
- 転送方向 : RD_WRN
- 10 ビットアドレスが読み出される場合 : HEAD10R ビット。HEAD10R を設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ（方向の変更の場合）かを示す必要があります。
- 転送されるバイト数 : NBYTES[7:0]バイト数が 255 バイト以上の場合は、NBYTES[7:0] に 0xFF を書き込む必要があります。

次に、I2C_CR2 レジスタの START ビットをセットする必要があります。START ビットがセットされているとき、上記のすべてのビットを変更することはできません。

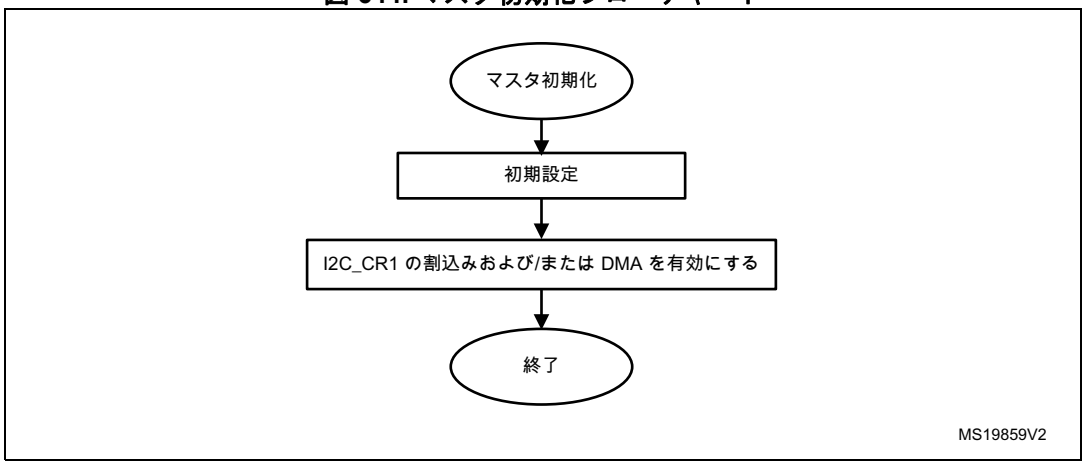
その場合、マスタは、バスがフリーである（BUSY = 0）ことを検出すると、t_{BUF} の遅延後に、自動的に START コンディションとスレーブアドレスを送信します。

アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

注： **START ビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。START ビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。**
10 ビットアドレッシングモードでは、スレーブアドレスの最初の 7 ビットがスレーブによって NACK されている場合、マスタは ACK が受信されるまで自動的にスレーブアドレスの送信を再開します。この場合、NACK をスレーブから受信するかどうか、ADDRCF をセットしてスレーブアドレスの送信を停止する必要があります。
START ビットがセットされているときに、I²C がスレーブとしてアドレス指定された場合（ADDR = 1）、I²C はスレーブ･モードに切り替わり、ADDRCF ビットがセットされたときにSTART ビットがクリアされます。

注： **反復スタート･コンディションにも同じ手順が適用されます。この場合、BUSY = 1 です。**

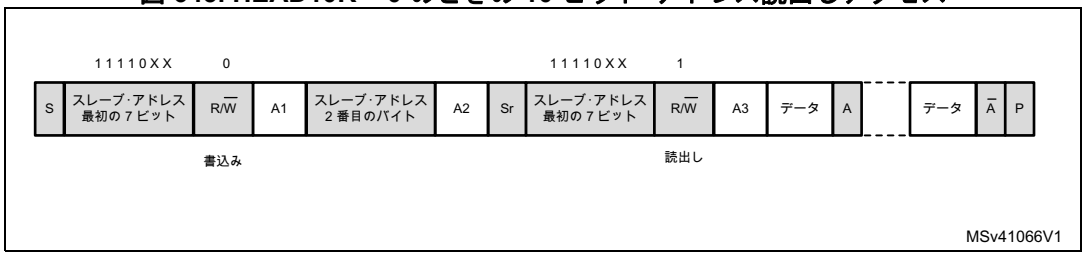
図 644. マスタ初期化フローチャート



10 ビット･アドレス･スレーブをアドレス指定するマスタ･レシーバの初期化

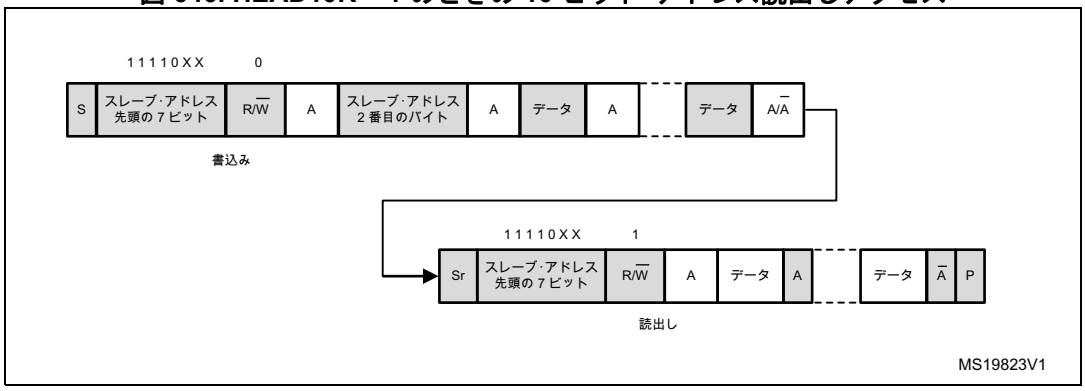
- スレーブアドレスが 10 ビット形式の場合、I2C_CR2 レジスタの HEAD10R ビットをクリアすることによって、完全な読出しシーケンスを送信することができます。この場合、マスタは、START ビットがセットされた後、次のような完全なシーケンスを自動的に送信します：（Re）START + スレーブ･アドレス 10 ビット･ヘッダ書込み + スレーブ･アドレスの 2 番目のバイト + REStart + スレーブ･アドレス 10 ビット･ヘッダ読出し

図 645. HEAD10R = 0 のときの 10 ビット･アドレス読出しアクセス



- マスタが 10 ビット･アドレス･スレーブをアドレス指定して、このスレーブ･アドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その後、HEAD10R = 1 で設定された 10 ビット･スレーブ･アドレスで反復開始が設定されます。この場合、マスタは次のシーケンスを送信します：ReStart + スレーブアドレス 10 ビットヘッダ読出し。

図 646. HEAD10R = 1 のときの 10 ビット･アドレス読出しアクセス



マスタ・トランスミッタ

書込み転送の場合、ACK が受信されたときの 9 番目の SCL パルス後、各バイトの送信後に TXIS フラグがセットされます。

I2C_CR1 レジスタの TXIE ビットがセットされている場合、TXIS イベント時に割込みが生成されます。このフラグは、I2C_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中の TXIS イベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

TXIS フラグは、NACK 受信時にはセットされません。

- RELOAD=0 で NBYTES データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、TC フラグがセットされ、ソフトウェアアクションを実行するために SCL ラインがローでストレッチされます：

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。
- NACK が受信された場合：TXIS フラグはセットされず、NACK 受信後、自動的に STOP コンディションが送信されます。I2C_ISR レジスタの NACKF フラグがセットされ、NACKIE ビットがセットされていた場合は割込みが生成されます。

図 647. N ≤ 255 バイトの場合の I²C マスタ・トランスミッタの転送シーケンス・フローチャート

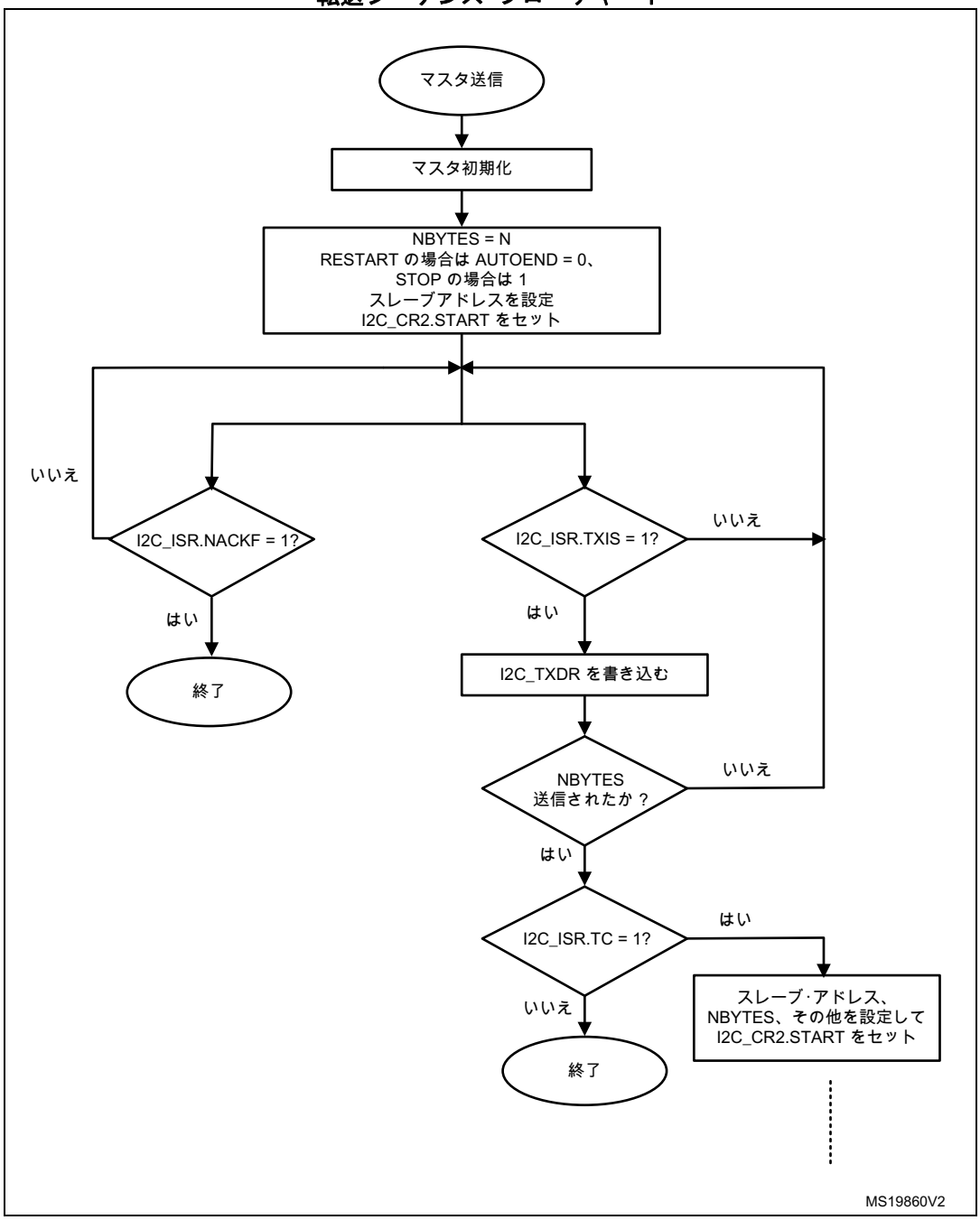


図 648. N > 255 バイトの場合の I2C マスタ・トランスミッタの転送シーケンス・フローチャート

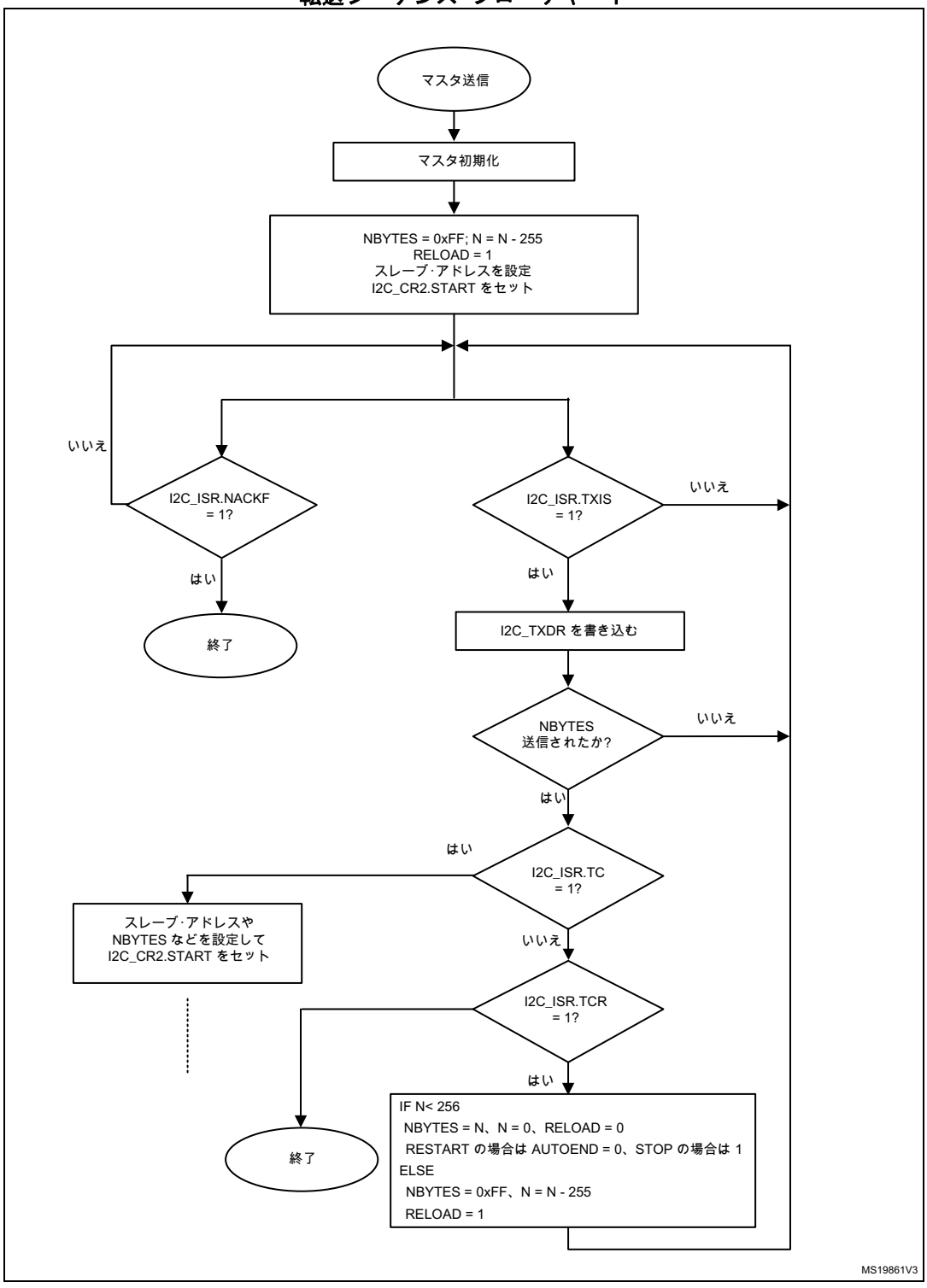
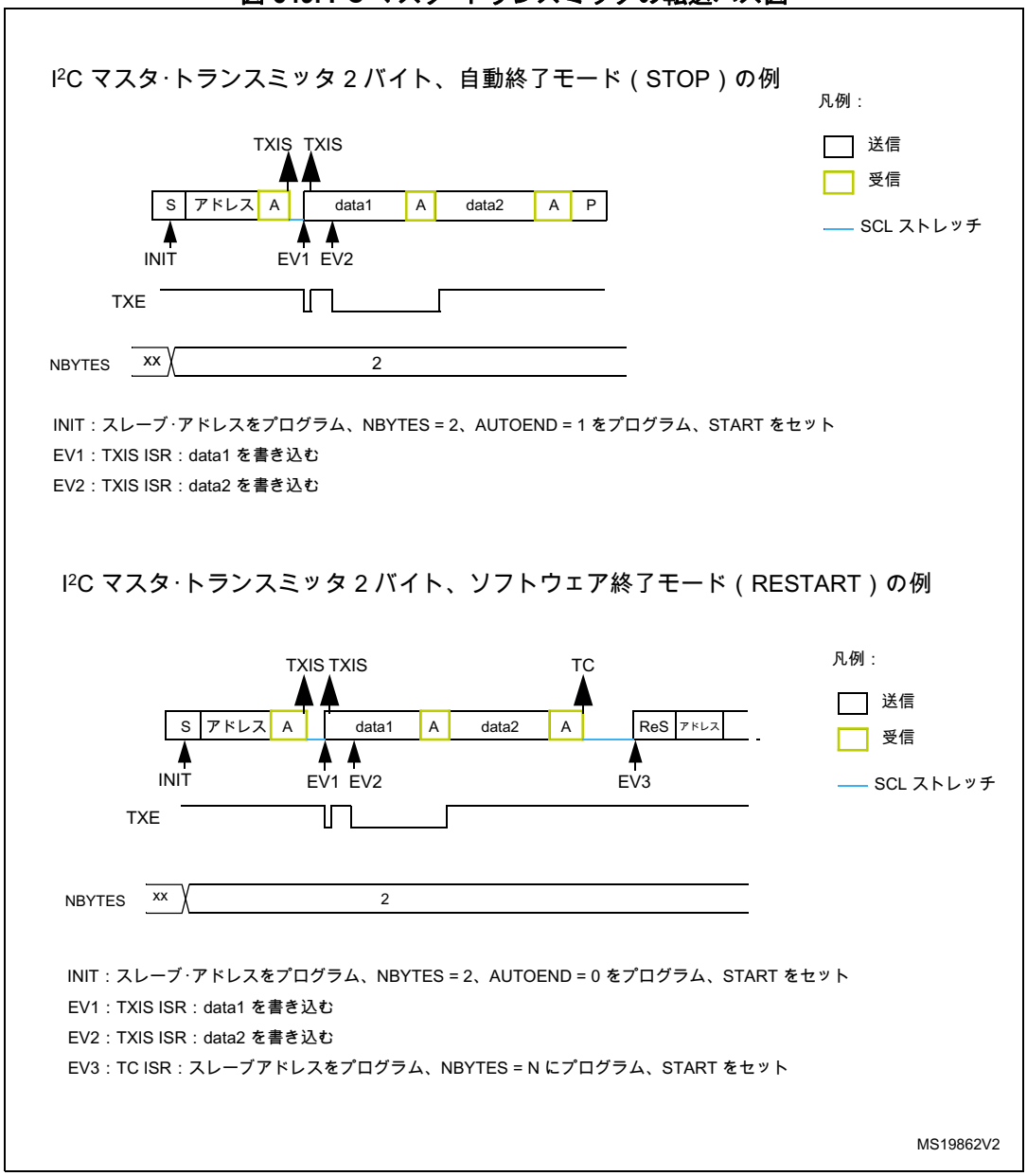


図 649. I²C マスタ・トランスミッタの転送バス図



マスタレシーバ

読出し転送の場合、各バイトの受信後や 8 番目の SCL パルス後に RXNE フラグがセットされます。I2C_CR1 レジスタの RXIE ビットがセットされている場合、RXNE イベント時に割込みが生成されます。このフラグは、I2C_RXDR が読み出されたときにクリアされます。

受信されるデータバイト数の合計が 255 より大きい場合、I2C_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

- RELOAD=0 で NBYTES[7:0] データが転送されたとき：
  - 自動終了モード（AUTOEND=1）では、最後の受信バイト後に NACK および STOP が自動的に送信されます。
  - ソフトウェア終了モード（AUTOEND=0）では、最後の受信バイト後に NACK が自動的に送信され、TC フラグがセットされ、ソフトウェアアクションを実行できるように、SCL ラインがローでストレッチされます。

    正しいスレーブアドレス設定と転送バイト数で I2C_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションとスレーブアドレスがバスに送信されます。

    I2C_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。

図 650. N ≤ 255 バイトの場合の I²C マスタ・レシーバの転送シーケンス・フローチャート

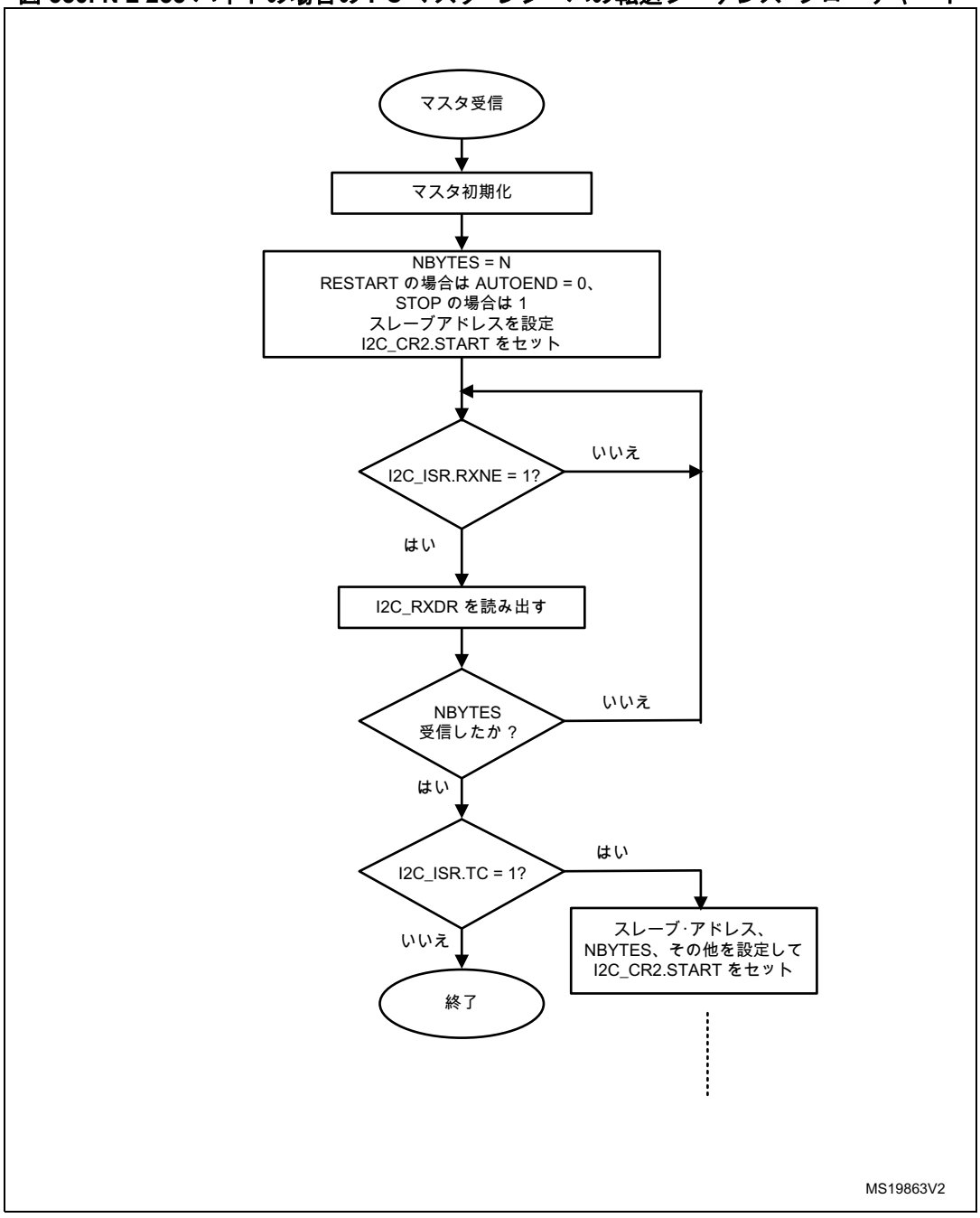


図 651. N > 255 バイトの場合の I²C マスタ・レシーバの転送シーケンス・フローチャート

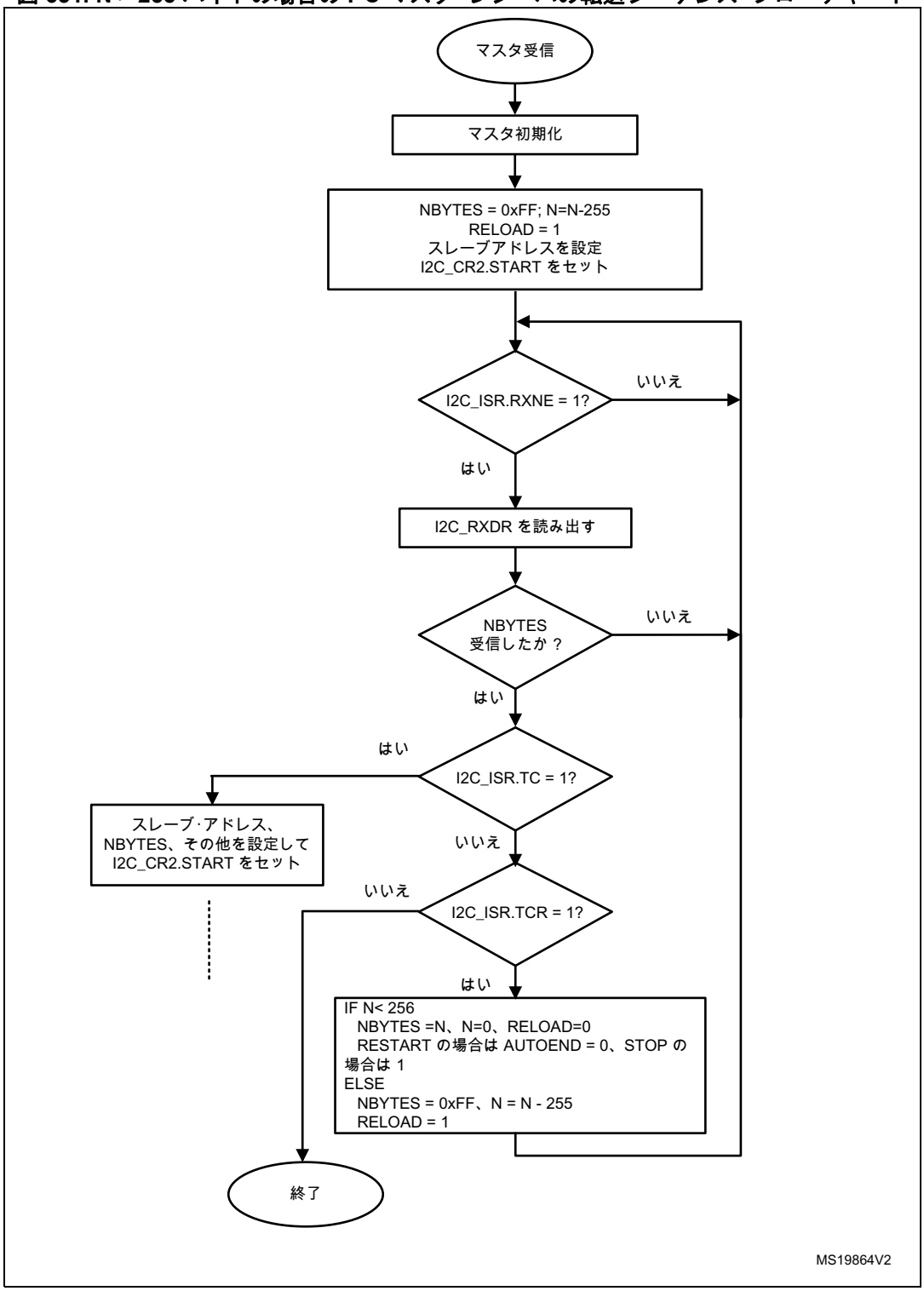
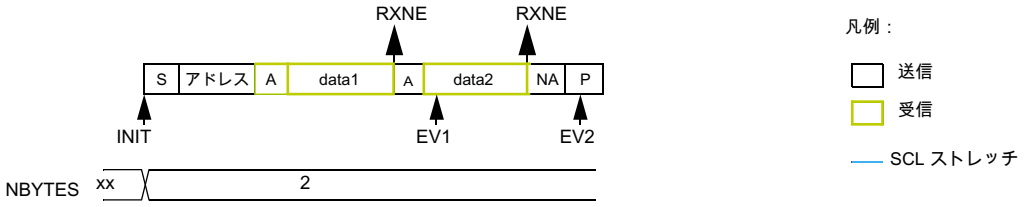


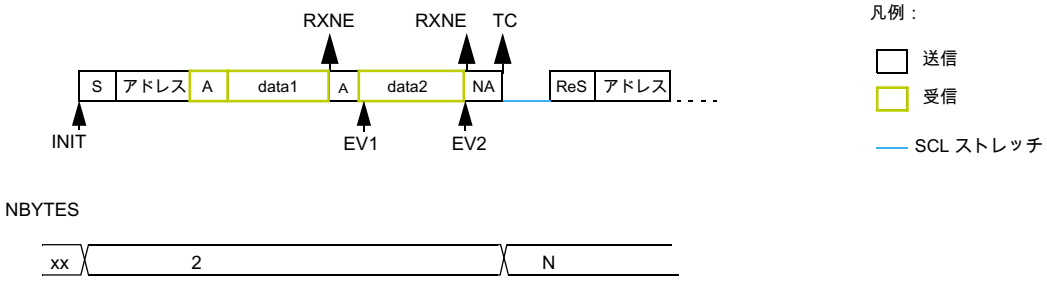
図 652. I²C マスタ·レシーバの転送バス図

I²C マスタ·レシーバ 2 バイト、自動終了モード（STOP）の例



INIT：スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=1 にプログラム、START をセット
EV1：RXNE ISR：data1 を読み出す
EV2：RXNE ISR：data2 を読み出す

I²C マスタ·レシーバ 2 バイト、ソフトウェア終了モード（RESTART）の例



INIT：スレーブ·アドレスをプログラム、NBYTES = 2、AUTOEND = 0 をプログラム、START をセット
EV1：RXNE ISR：data1 を読み出す
EV2：RXNE ISR：data2 を読み出す
EV3：TC ISR：スレーブアドレスをプログラム、NBYTES = N にプログラム、START をセット

MS19865V1

41.4.10 I2C_TIMINGR レジスタの設定例

下の表に、I²C 仕様に準拠したタイミングを得るための I2C_TIMINGR をプログラムする方法の例を示します。より正確な設定値を得るには、STM32CubeMX ツール（I2C 設定ウィンドウ）を使用する必要があります。

表 384. f_{I2CCLK} = 8 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	500 kHz
PRESC	1	1	0	0
SCLL	0xC7	0x13	0x9	0x6
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	7 x 125 ns = 875 ns
SCLH	0xC3	0xF	0x3	0x3
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125ns = 500 ns	4 x 125ns = 500 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～2000 ns(4)
SDADEL	0x2	0x2	0x1	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	1 x 125 ns = 125 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125ns = 500 ns	2 x 125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 655 ns

表 385. f_{I2CCLK} = 16 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	3	3	1	0
SCLL	0xC7	0x13	0x9	0x4
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	5 x 62.5 ns = 312.5 ns
SCLH	0xC3	0xF	0x3	0x2
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125ns = 500 ns	3 x 62.5 ns = 187.5 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～1000 ns(4)
SDADEL	0x2	0x2	0x2	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	2 x 125 ns = 250 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x2
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125ns = 500 ns	3 x 62.5 ns = 187.5 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 500 ns

表 386. f_{I2CCLK} = 48 MHz でのタイミング設定の例

パラメータ	標準モード（Sm）		高速モード（Fm）	高速モードプラス (Fm+)
	10 kHz	100 kHz	400 kHz	1000 kHz
PRESC	0xB	0xB	5	5
SCLL	0xC7	0x13	0x9	0x3
t_{SCLL}	200 x 250 ns = 50 µs	20 x 250 ns = 5.0 µs	10 x 125 ns = 1250 ns	4 x 125ns = 500 ns
SCLH	0xC3	0xF	0x3	0x1
t_{SCLH}	196 x 250 ns = 49 µs	16 x 250 ns = 4.0 µs	4 x 125ns = 500 ns	2 x 125 ns = 250 ns
t_{SCL}(1)	～100 µs(2)	～10 µs (2)	～2500 ns(3)	～875 ns(4)
SDADEL	0x2	0x2	0x3	0x0
t_{SDADEL}	2 x 250 ns = 500 ns	2 x 250 ns = 500 ns	3 x 125 ns = 375 ns	0 ns
SCLDEL	0x4	0x4	0x3	0x1
t_{SCLDEL}	5 x 250 ns = 1250 ns	5 x 250 ns = 1250 ns	4 x 125ns = 500 ns	2 x 125 ns = 250 ns

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 x t_{I2CCLK} = 83.3 ns です。例：t_{SYNC1} + t_{SYNC2} = 250 ns

41.4.11 SMBus 固有の機能

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 41.3：I2C の実装を参照してください。

概要

システム管理バス（SMBus）は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる 2 線インタフェースです。I²C の動作原理に基づきます。SMBus により、システムおよびパワー･マネージメント関連のタスク向けの制御バスを実現できます。

このペリフェラルは、SMBus 仕様（http://smbus.org）と互換性があります。

システム管理バス仕様では、3 種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムの CPU にメインインタフェースを提供する特殊なマスタです。ホストは、マスタ／スレーブとすることができ、SMBus ホスト通知プロトコルをサポートする必要があります。システム内では、ただ 1 つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

バスプロトコル

特定のデバイスについて、11 の可能なコマンドプロトコルがあります。デバイスは、11 のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、および Block Write-Block Read Process Call です。これらのプロトコルは、ユーザのソフトウェアによって実装してください。

これらのプロトコルの詳細については、SMBus 仕様（http://smbus.org）を参照してください。

アドレス解決プロトコル（ARP）

SMBus スレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子（UDID）を実装する必要があります。128 ビットの番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル（ARP）をサポートします。SMBus デバイスのデフォルトアドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。ARP コマンドは、ユーザのソフトウェアによって実装してください。

ARP サポートのために、スレーブモードでアービトレーションも行われます。

SMBus アドレス解決プロトコルの詳細については、SMBus 仕様（http://smbus.org）を参照してください。

受信コマンドおよびデータ確認応答制御

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、1839 ページのスレーブ･バイト制御モードを参照してください。

Host Notify プロトコル

このペリフェラルは、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって、Host Notify（ホスト通知）プロトコルをサポートします。この場合、ホストは SMBus ホスト･アドレス（0b0001 000）を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBus アラート

SMBus ALERT オプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストの SMBALERT# ピンを通じてホストに信号を送信します。ホストは、割込みを処理し、アラート応答アドレス（0b0001 100）を通じて全 SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブ･デバイスとして設定されている場合（SMBHEN = 0）、I2C_CR1 レジスタの ALERTEN ビットをセットすることによって、SMBA ピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき（SMBHEN=1）、SMBA ピンで立ち下がりエッジが検出され、ALERTEN=1 のとき、I2C_ISR レジスタの ALERT フラグがセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割込みが生成されます。ALERTEN=0 のときには、外部 SMBA ピンがローの場合でも、ALERT ラインはハイとみなされます。

SMBus ALERT ピンが不要な場合には、ALERTEN=0 の場合、SMBA ピンを標準 GPIO として使用できます。

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus 仕様にパケットエラーチェックメカニズムが導入されました。パケット･エラー･チェックは、各メッセージ転送の終わりにパケット･エラー･コード（PEC）を付加することによって実装されます。PEC は、すべてのメッセージ･バイト（アドレスと読出し／書込みビットを含む）に対して $C(x) = x_8 + x^2 + x + 1$ CRC-8 多項式を使用して計算されます。

ペリフェラルはハードウェア PEC 計算機を備えているので、受信バイトとハードウェアによって計算した PEC が一致しない場合は自動的に非確認応答を送信できます。

タイムアウト

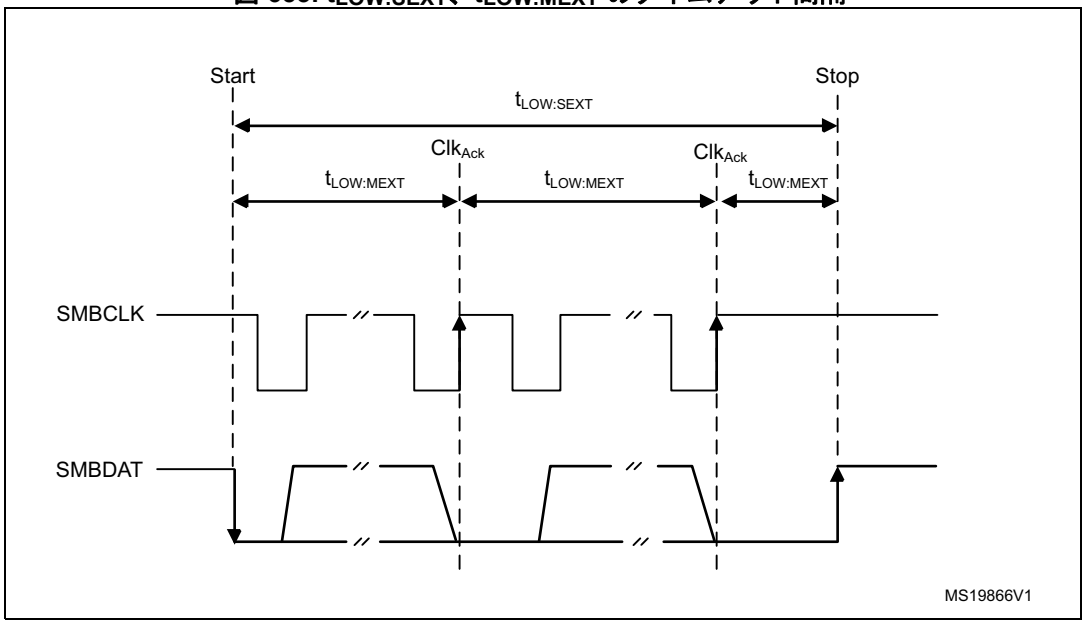
このペリフェラルは、SMBus 仕様で定義された 3 つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

表 387. SMBus タイムアウト仕様

記号	パラメータ	リミット		単位
		最小値	最大値	
$t_{TIMEOUT}$	クロックロータイムアウト検出	25	35	ms
$t_{LOW:SEXT}$(1)	累積クロックロー延長時間（スレーブデバイス）	-	25	ms
$t_{LOW:MEXT}$(2)	累積クロックロー延長時間（マスタデバイス）	-	10	ms

1. $t_{LOW:SEXT}$ は、特定のスレーブデバイスが初めの START から STOP までの 1 つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が $t_{LOW:SEXT}$ より大きくなることがあります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。
2. $t_{LOW:MEXT}$ は、マスタがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、または ACK-to-STOP から定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が $t_{LOW:MEXT}$ より大きくなることがあります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図 653. $t_{LOW:SEXT}$、$t_{LOW:MEXT}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が $t_{HIGH,MAX}$ より大きい t_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます（表 381 : I2C-SMBus 仕様のデータのセットアップおよびホールド時間を参照してください）。

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLK または SMBDAT ラインで状態遷移を検出しなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

41.4.12 SMBus 初期化

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 41.3 : I2C の実装を参照してください。

SMBus 通信を行うためには、I²C 初期化に加えて、他にも特定の初期化を行う必要があります。

受信コマンドおよびデータ確認応答制御（スレーブ･モード）

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブ･モードで ACK 制御を可能にするには、I2C_CR1 レジスタの SBC ビットをセットすることによって、スレーブ･バイト制御モードを有効にする必要があります。詳細については、1839 ページのスレーブ･バイト制御モードを参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定の SMBus アドレスを有効にしてください。詳細については、1862 ページのバスアイドル検出を参照してください。

- SMBus デバイスのデフォルト･アドレス（0b1100 001）は、I2C_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。
- SMBus ホスト･アドレス（0b0001 000）は、I2C_CR1 レジスタの SMBHEN ビットをセットすることによって有効になります。
- アラート応答アドレス（0b0001100）は、I2C_CR1 レジスタの ALERTEN ビットをセットすることによって有効になります。

パケット･エラー･チェック

PEC 計算を有効にするには、I2C_CR1 レジスタの PECEN ビットをセットします。その場合、PEC 転送はハードウェアバイトカウンタ（I2C_CR2 レジスタの NBYTES[7:0]）を使用して管理されます。PECEN ビットは、I²C を有効にする前に設定する必要があります。

PEC 転送はハードウェアバイトカウンタによって管理されるので、スレーブモードで SMBus とインタフェースするときには SBC ビットをセットする必要があります。PECBYTE ビットがセットされ、RELOAD ビットがクリアされているとき、NBYTES - 1 個のデータの転送後に PEC が転送されます。RELOAD をセットした場合、PECBYTE は効果がありません。

注意： **I²C が有効なときには、PECEN 設定の変更はできません。**

表 388. SMBus の PEC 設定

モード	SBC ビット	RELOAD ビット	AUTOEND ビット	PECBYTE ビット
マスタ Tx/Rx NBYTES + PEC+ STOP	x	0	1	1
マスタ Tx/Rx NBYTES + PEC + ReSTART	x	0	0	1
スレーブ Tx/Rx と PEC	1	0	x	1

タイムアウト検出

タイムアウト検出は、I2C_TIMEOUTR レジスタの TIMOUTEN および TEXTEN ビットをセットすることによって有効になります。SMBus 仕様で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- $t_{TIMEOUT}$ チェック

  $t_{TIMEOUT}$ チェックを有効にするためには、12 ビットの TIMEOUTA[11:0] ビットを $t_{TIMEOUT}$ パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCL ローレベルタイムアウトを検出するためには、TIDLE ビットを 0 に設定する必要があります。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN をセットすることによって有効になります。

  SCL が (TIMEOUTA+1) x 2048 x t_{I2CCLK} より長い時間、ローに設定された場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 389：さまざまな I2CCLK周波数での TIMEOUTA の設定例 （最大値 tTIMEOUT = 25 ms）を参照してください。

注意： **TIMEOUTEN ビットがセットされているときには、TIMEOUTA[11:0] ビットおよび TIDLE ビットの設定変更はできません。**

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ チェック

  ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12 ビットの TIMEOUTB タイマは、スレーブの場合は $t_{LOW:SEXT}$ をチェックするために、マスタの場合は $t_{LOW:MEXT}$ をチェックするために、設定する必要があります。標準では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

  その場合、タイマは、I2C_TIMEOUTR レジスタの TEXTEN ビットをセットすることによって有効になります。

  SMBus ペリフェラルが、（TIMEOUTB+1）x 2048 x t_{I2CCLK} より長い時間および 1862 ページのバスアイドル検出 セクションで述べられているタイムアウト間隔で、累積 SCL ストレッチを実行した場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

  表 390：さまざまな I2CCLK 周波数での TIMEOUTB の設定例を参照してください。

注意： **TEXTEN ビットがセットされているときには、TIMEOUTB 設定変更はできません。**

バス･アイドル検出

t_{IDLE} チェックを有効にするには、12 ビットの TIMEOUTA[11:0] フィールドを、t_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCL および SDA ハイレベルタイムアウトを検出するためには、TIDLE ビットを 1 に設定する必要があります。

その場合、タイマは、I2C_TIMEOUTR レジスタの TIMOUTEN ビットをセットすることによって有効になります。

SCL および SDA の両方のラインが（TIMEOUTA+1）x 4 x t_{I2CCLK} より長い間ハイのままであった場合、I2C_ISR レジスタの TIMEOUT フラグがセットされます。

表 391：さまざまな I2CCLK周波数での TIMEOUTA の設定例（最大値 tIDLE = 50 µs）を参照してください。

注意： **TIMEOUTEN がセットされているときに、TIMEOUTA および TIDLE 設定を変更することはできません。**

41.4.13 SMBus：I2C_TIMEOUTR レジスタの設定例

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 41.3：I2C の実装を参照してください。

- $t_{TIMEOUT}$ の最大時間を 25 ms に設定：

表 389. さまざまな I2CCLK周波数での TIMEOUTA の設定例（最大値 $t_{TIMEOUT}$ = 25 ms）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	$t_{TIMEOUT}$
8 MHz	0x61	0	1	98 x 2048 x 125 ns = 25 ms
16 MHz	0xC3	0	1	196 x 2048 x 62.5 ns = 25 ms
32 MHz	0x186	0	1	391 x 2048 x 31.25 ns = 25 ms
48 MHz	0x249	0	1	586 x 2048 x 20.08 ns = 25 ms

- $t_{LOW:SEXT}$ および $t_{LOW:MEXT}$ の最大時間を 8 ms に設定：

表 390. さまざまな I2CCLK 周波数での TIMEOUTB の設定例

f_{I2CCLK}	TIMEOUTB[11:0] ビット	TEXTEN ビット	$t_{LOW:EXT}$
8 MHz	0x1F	1	32 x 2048 x 125 ns = 8 ms
16 MHz	0x3F	1	64 x 2048 x 62.5 ns = 8 ms
48 MHz	0xBB	1	188 x 2048 x 20.08 ns = 8 ms

- t_{IDLE} の最大時間を 50 µs に設定

表 391. さまざまな I2CCLK周波数での TIMEOUTA の設定例（最大値 t_{IDLE} = 50 µs）

f_{I2CCLK}	TIMEOUTA[11:0] ビット	TIDLE ビット	TIMEOUTEN ビット	t_{TIDLE}
8 MHz	0x63	1	1	100 x 4 x 125 ns = 50 µs
16 MHz	0xC7	1	1	200 x 4 x 62.5 ns = 50 µs
48 MHz	0x257	1	1	600 x 4 x 20.08 ns = 50 µs

41.4.14 SMBus スレーブモード

このセクションは、SMBus 機能がサポートされている場合にのみ適用されます。セクション 41.3 : I2C の実装を参照してください。

I²C スレーブ転送管理（セクション 41.4.8 : I2C スレーブモードを参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェア·フローがあります。

SMBus スレーブ·トランスミッタ

IP を SMBus で使用する場合は、プログラムした数のデータ·バイトの送信が終了したときに PEC を送信できるように、SBC を 1 にプログラムする必要があります。PECBYTE ビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数には PEC 送信が含まれます。その場合、TXIS 割込みの合計数は NBYTES - 1 であり、NBYTES - 1 のデータ転送後にマスタが追加のバイトをリクエストすると、I2C_PECR レジスタの内容が自動的に送信されます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 654. N バイト + PEC の場合の SMBus スレーブ·トランスミッタの転送シーケンス·フローチャート

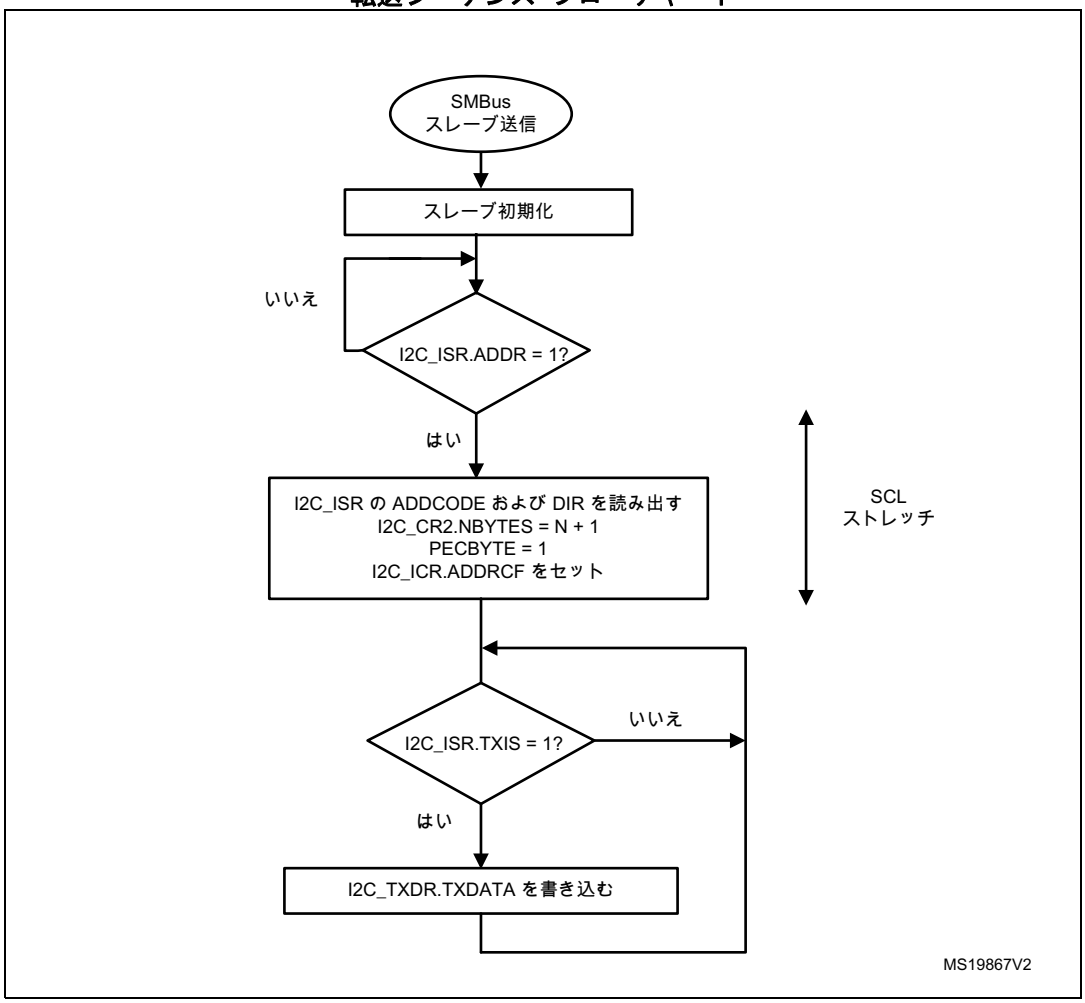
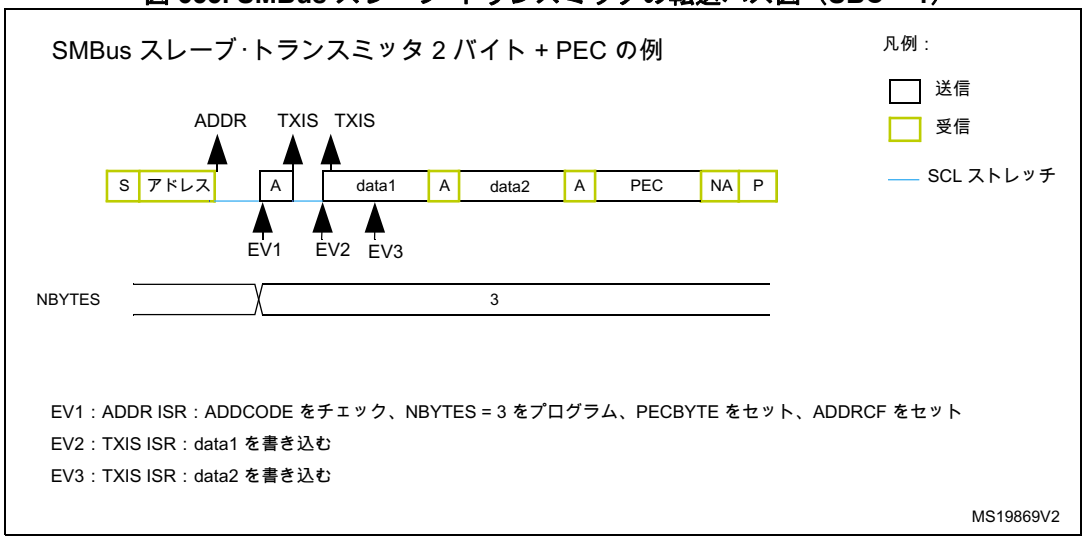


図 655. SMBus スレーブ·トランスミッタの転送バス図（SBC = 1）



SMBus スレーブレシーバ

I2C が SMBus モードで使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC チェックを可能にするため、1 にプログラムする必要があります。各バイトの ACK 制御を可能にするためには、再ロードモードを選択する必要があります（RELOAD=1）。詳細については、1839 ページのスレーブ·バイト制御モードを参照してください。

PEC バイトをチェックするためには、RELOAD ビットをクリアして、PECBYTE ビットをセットする必要があります。この場合、NBYTES - 1 のデータが受信された後、次の受信バイトが内部 I2C_PECR レジスタの内容と比較されます。ACK ビットの値にかかわらず、比較が一致しなかった場合は NACK が自動的に生成され、比較が一致した場合は ACK が自動的に生成されます。PEC バイトが受信されると、他のデータと同様に I2C_RXDR レジスタにコピーされ、RXNE フラグがセットされます。

PEC 不一致の場合、PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

ACK ソフトウェア制御が不要な場合は、PECBYTE=1 をプログラムし、同じ書込み操作で NBYTES を連続フローで受信するバイト数にプログラムします。NBYTES - 1 が受信された後、次の受信バイトが PEC であるかどうかがチェックされます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 656. N バイト + PEC の場合の SMBus スレーブ･レシーバの転送シーケンス･フローチャート

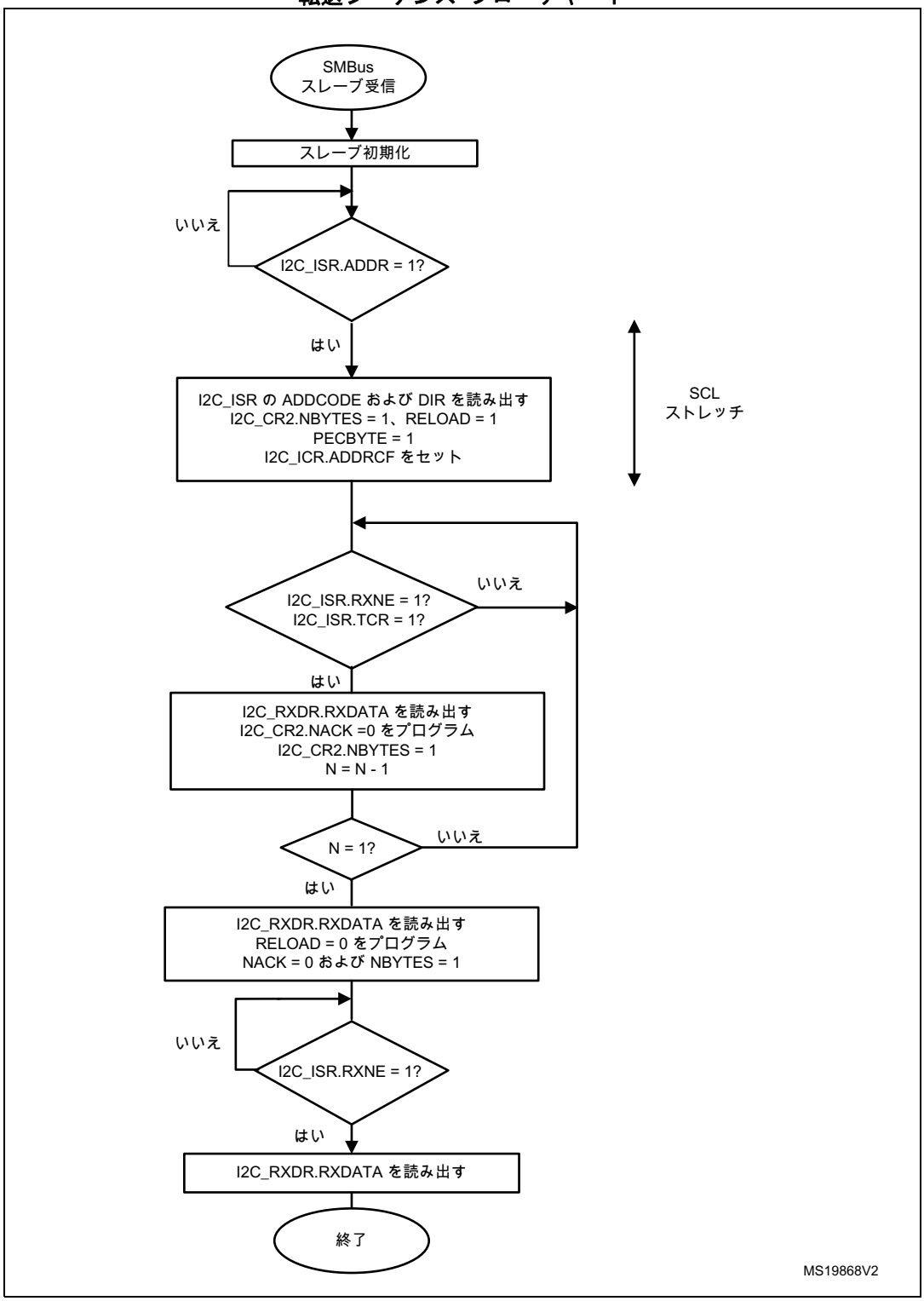
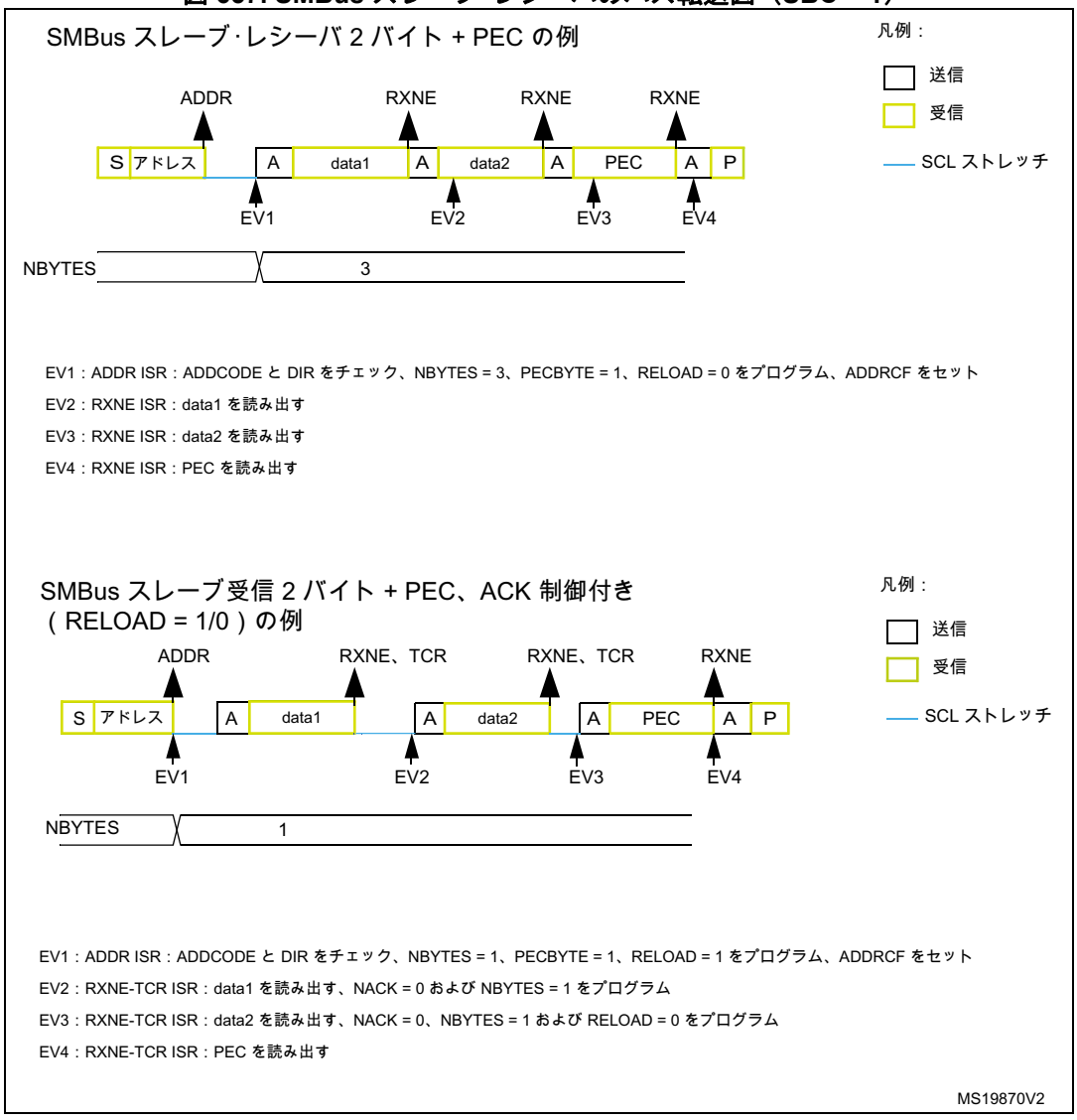


図 657. SMBus スレーブ·レシーバのバス転送図（SBC = 1）



このセクションは、SMBus 機能がサポートされている場合にのみ適用されます。セクション 41.3 : I2C の実装を参照してください。

I2C マスタ転送管理（セクション 41.4.9 : I2C マスタモードを参照）に加えて、SMBus をサポートするために、いくつか追加のソフトウェア·フローがあります。

SMBus マスタ·トランスミッタ

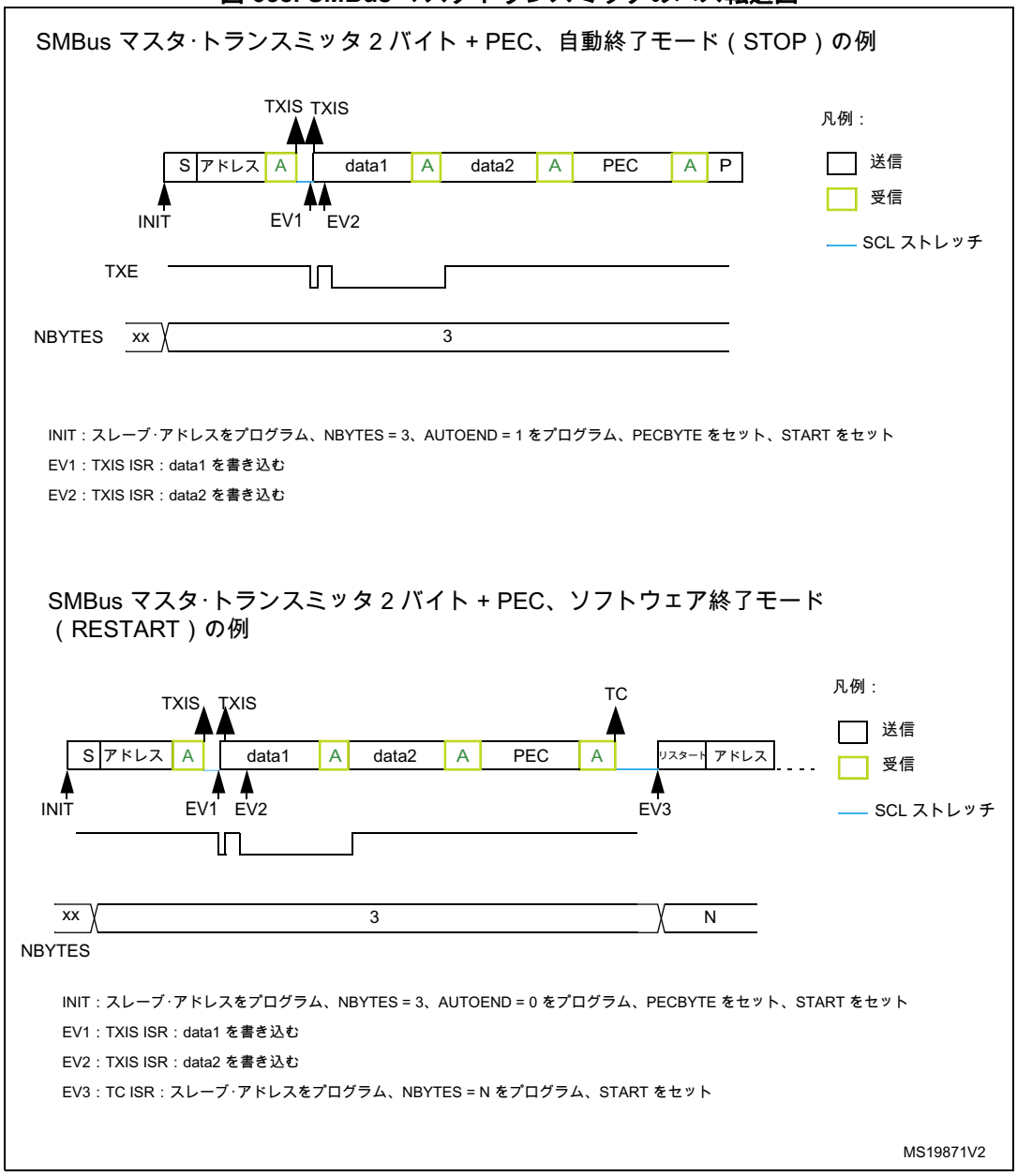
SMBus マスタから PEC を送信するには、START ビットをセットする前に、PECBYTE ビットをセットし、バイト数を NBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS 割込みの合計数は NBYTES - 1 になります。したがって、NBYTES = 0x1 のときに PECBYTE ビットがセットされた場合、I2C_PECR レジスタの内容が自動的に送信されます。

PEC 後に SMBus マスタから STOP コンディションを送信する場合は、自動終了モードを選択する必要があります（AUTOEND = 1）。この場合、PEC 送信に続いて、STOP コンディションが自動的に送信されます。

SMBus マスタが PEC 後に RESTART コンディションを送信したい場合は、ソフトウェアモードを選択してください（AUTOEND=0）。この場合、NBYTES - 1 が送信されると、I2C_PECR レジスタの内容が送信され、PEC の送信が完了すると TC フラグがセットされて、SCL ラインがローでストレッチされます。RESTART コンディションを TC 割込みサブルーチンでプログラムする必要があります。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 658. SMBus マスタトランスミッタのバス転送図



SMBus マスタ・レシーバ

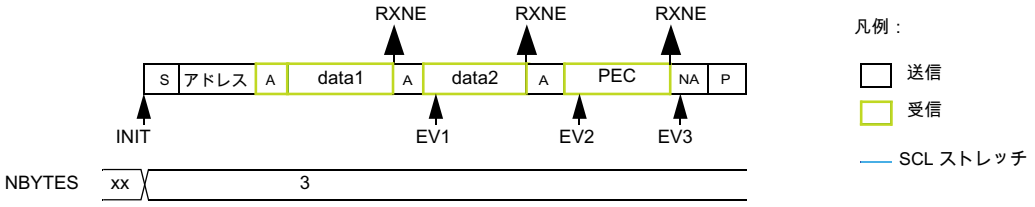
SMBus マスタで、転送終了後に PEC を受信してから STOP を受信するには、自動終了モードを選択します（AUTOEND = 1）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブ・アドレスをプログラムする必要があります。この場合、NBYTES - 1 のデータが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイトに対する NACK 応答が得られた後、STOP コンディションが送信されます。

SMBus マスタが転送終了時に PEC バイトを受信してから RESTART を受信したいときには、ソフトウェアモードを選択する必要があります（AUTOEND=0）。START ビットをセットする前に、PECBYTE ビットをセットする必要があり、スレーブアドレスをプログラムする必要があります。この場合、NBYTES - 1 のデータが受信された後、次の受信バイトが I2C_PECR レジスタの内容と自動的に照合されます。PEC バイト受信後に TC フラグがセットされ、SCL ラインがローでストレッチされます。RESTART コンディションは、TC 割込みサブルーチンでプログラムできます。

注意： **PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。**

図 659. SMBus マスタレシーバのバス転送図

SMBus マスタ·レシーバ 2 バイト + PEC、自動終了モード（STOP）の例



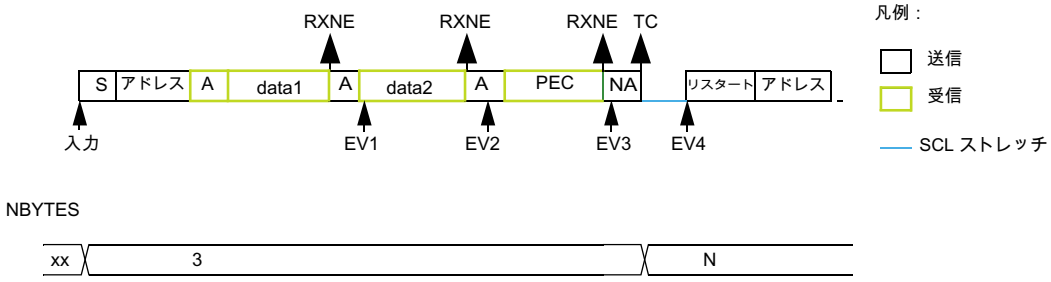
INIT：スレーブ·アドレスをプログラム、NBYTES = 3、AUTOEND = 1 をプログラム、PECBYTE をセット、START をセット

EV1：RXNE ISR：data1 を読み出す

EV2：RXNE ISR：data2 を読み出す

EV3：RXNE ISR：PEC を読み出す

SMBus マスタ·レシーバ 2 バイト + PEC、ソフトウェア終了モード（RESTART）の例



INIT：スレーブ·アドレスをプログラム、NBYTES = 3、AUTOEND = 0 をプログラム、PECBYTE をセット、START をセット

EV1：RXNE ISR：data1 を読み出す

EV2：RXNE ISR：data2 を読み出す

EV3：RXNE ISR：PEC を読み出す

EV4：TC ISR：スレーブアドレスをプログラム、NBYTES = N にプログラム、START をセット

MS19872V2

41.4.15 アドレス一致時に STOP モードからウェイクアップ

このセクションは、STOP モードからのウェイクアップ機能がサポートされるときにのみ適用されます。セクション 41.3：I2C の実装を参照してください。

I²C は、アドレス指定されたとき、MCU を STOP モードからウェイクアップできます（APB クロックはオフ）。すべてのアドレッシングモードがサポートされます。

STOP モードからのウェイクアップを有効にするには、I2C_CR1 レジスタの WUPEN ビットをセットします。STOP モードからのウェイクアップを可能にするには、HSI16 オシレータを I2CCLK のクロックソースとして選択する必要があります。

STOP モード中、HSI16 はオフです。START が検出されると、I2C インタフェースは HSI16 をオンに切り替えて、HSI16 がウェイクアップするまで SCL ローをストレッチします。

HSI16 は、アドレス受信に使用されます。

アドレス一致の場合、I²C は、MCU のウェイクアップ時間の間、SCLローをストレッチします。ストレッチは、ADDR フラグがソフトウェアによってクリアされたときにリリースされ、転送は通常通りに続行されます。

アドレスが一致しなかった場合、HSI16 は再びオフになり、MCU はウェイクアップしません。

注： **I²C クロックがシステムクロックの場合、または WUPEN = 0 の場合、START 受信後もHSI16はオンになりません。**

ADDR 割込みによってのみ、MCU をウェイクアップできます。したがって、I²C がマスタとして、または ADDR フラグのセット後にアドレス指定されたスレーブとして転送を行っているときには、STOP モードに入らないでください。これを管理するには、ADDR 割込みルーチンで SLEEPDEEP ビットをクリアして、STOPF フラグのセット後にのみ再びオンにセットします。

注意： **デジタルフィルタは、STOP モードからのウェイクアップ機能と互換性がありません。DNF ビットが 0 でない場合、WUPEN ビットをセットしても効果はありません。**

注意： **この機能は、I²C クロックソースが HSI16 オシレータのときのみ使用できます。**

注意： **STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロック･ストレッチを有効にする必要があります（NOSTRETCH = 0）。**

注意： **STOP モードからのウェイクアップが無効な場合（WUPEN = 0）、STOP モードに入る前に、I²C ペリフェラルを無効にする必要があります（PE=0）。**

41.4.16 エラー条件

以下のエラーは、通信エラーを引き起こす可能性のあるエラー条件です。

バスエラー（BERR）

バスエラーは、START または STOP コンディションが検出され、複数の 9 SCL クロックパルス後になかったときに検出されます。START または STOP コンディションは、SCL がハイとのときに SDA エッジが発生した場合に検出されます。

バスエラーフラグは、I²C がマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ（すなわち、スレーブモードのアドレスフェーズでないとき）、セットされます。

スレーブモードで START または RESTART の誤配置が検出された場合、I²C は、正しい START コンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2C_ISR レジスタの BERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

アービトレーション喪失（ARLO）

アービトレーション喪失は、SDA ラインでハイレベルが送信されたが、SCL 立ち上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDA および SCL ラインはリリースされ、START 制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCL および SDA ラインがリリースされます。

アービトレーション喪失が検出されると、I2C_ISR レジスタの ARLO フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

オーバーラン／アンダーランエラー（OVR）

オーバーラン･エラーまたはアンダーラン･エラーは、スレーブ･モードで NOSTRETCH = 1 に設定されていて、かつ、次の場合に検出されます：

- 受信時、新しいバイトが受信され、RXDR レジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答として NACK が自動的に送信されます。
- 送信時：
  - STOPF=1 のときには、最初のデータバイトが送信されなければなりません。TXE=0の場合、I2C_TXDR レジスタの内容が送信され、そうでない場合は0xFFが送信されます。
  - 新しいバイトが送信されるべきときに、I2C_TXDR レジスタがまだ書き込まれていなかった場合、0xFF が送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2C_ISR レジスタの OVR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

パケット･エラー･チェック･エラー（PECERR）

このセクションは、SMBus 機能がサポートされている場合にのみ適用されます。セクション 41.3：I2C の実装を参照してください。

PEC エラーは、受信した PEC バイトが I2C_PECR レジスタの内容と一致しなかったときに検出されます。正しくない PEC の受信後、NACK が自動的に送信されます。

PEC エラーが検出されると、I2C_ISR レジスタの PECERR フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

タイムアウトエラー（TIMEOUT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 41.3：I2C の実装を参照してください。

タイムアウトエラーは、次のような条件で発生します：

- TIDLE=0 であり、SCL が TIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これは SMBus タイムアウトの検出に使用されます。
- TIDLE=1 であり、SDA および SCL が TIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:MEXT}$ パラメータ）。
- スレーブ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合（SMBus $t_{LOW:SEXT}$ パラメータ）。

マスタモードでタイムアウト違反が検出されると、STOP コンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDA および SCL ラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2C_ISR レジスタの TIMEOUT フラグがセットされ、I2C_CR1 レジスタの ERRIE ビットがセットされていた場合は割込みが生成されます。

アラート（ALERT）

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。セクション 41.3 : I2C の実装を参照してください。

ALERT フラグは、I²C インタフェースがホストとして設定され（SMBHEN=1）、アラートピン検出が有効であり（ALERTEN=1）、SMBA ピンで立ち下がりエッジが検出されたときにセットされます。I2C_CR1 レジスタの ERRIE ビットがセットされている場合は、割込みが生成されます。

41.4.17 DMA リクエスト

DMA を使用した送信

送信で DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの TXDMAEN ビットをセットします。TXIS ビットがセットされるたびに、DMA ペリフェラル（セクション 12 : 397 ページのダイレクト･メモリ･アクセス（DMA）コントローラを参照）を使用して設定された SRAM 領域から I2C_TXDR レジスタにデータがロードされます。

データのみが DMA で転送されます。

- マスタモード : 初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます（送信されたスレーブアドレスを DMA で転送することはできません）。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。1850 ページのマスタ･トランスミッタを参照してください。
- スレーブモードでは :
  - NOSTRETCH = 0 では、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割込みサブルーチンで ADDR をクリアする前に DMA を初期化する必要があります。
  - NOSTRETCH = 1 では、アドレス一致イベントの前に DMA を初期化する必要があります。
- SMBus をサポートする場合 : PEC 転送は NBYTES カウンタによって管理されます。1865 ページの SMBus スレーブ･トランスミッタおよび 1868 ページの SMBus マスタ･トランスミッタを参照してください。

注 : DMA が送信に使用される場合、TXIE ビットが有効である必要はありません。

DMA を使用した受信

受信で DMA（Direct Memory Access）を有効にするには、I2C_CR1 レジスタの RXDMAEN ビットをセットします。RXNE ビットがセットされているときには、I2C_RXDR レジスタから、DMA ペリフェラル（セクション 12 : 397 ページのダイレクト･メモリ･アクセス（DMA）コントローラを参照）を使用して設定された SRAM 領域にデータがロードされます。データのみ（PEC を含む）が DMA で転送されます。

- マスタ･モード、初期化、スレーブ･アドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。

- NOSTRETCH = 0 のスレーブ･モードでは、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割込みサブルーチンで ADDR をクリアする前に DMA を初期化する必要があります。
- SMBus がサポートされる場合（セクション 41.3 : I2C の実装 を参照）: PEC 転送は NBYTES カウンタによって管理されます。1866 ページのSMBus スレーブレシーバおよび1870 ページのSMBus マスタ･レシーバを参照してください。

注 : DMA が受信に使用される場合、RXIE ビットが有効である必要はありません。

41.4.18 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールの DBG_I2Cx_ 設定ビットに応じて、SMBus タイムアウトは、通常の動作を続行するか、あるいは停止します。

41.5 I²C 低電力モード

表 392. 低電力モードが I²C に与える影響

モード	説明
SLEEP	影響はありません。I²C 割込みによって、デバイスは SLEEP モードから復帰します。
STOP(1)	I²C レジスタの内容は保たれます。WUPEN=1 かつ I²C が内部オシレータ（HSI16）によってクロック供給されている場合 : アドレス認識が機能します。I²C アドレス一致条件によって、デバイスは STOP モードから復帰します。WUPEN=0 の場合 : STOP モードに入る前に I²C を無効にする必要があります。
STANDBY	I²C ペリフェラルはパワーダウンされ、STANDBY モード終了後に再初期化する必要があります。

1. 各I²Cペリフェラルでサポートされる STOP モードについては、セクション 41.3 : I2C の実装を参照してください。特定の STOP モードからのウェイクアップがサポートされていない場合、その STOP モードに入る前にI²Cペリフェラルを無効にする必要があります。

41.6 I²C 割込み

次の表に、I²C 割込みリクエストの一覧を示します。

表 393. I²C 割込みリクエスト

項目（割込みの略称）		割込みイベント	イベントフラグ	有効制御ビット	割込みのクリア方法	SLEEP モードの終了	STOP モードの終了	STANDBY モードの終了
I2C	I2C_EV	受信バッファ･ノット･エンプティ	RXNE	RXIE	I2C_RXDR レジスタを読み出す	可能	不可	不可
		送信バッファ割込みステータス	TXIS	TXIE	I2C_TXDR レジスタに書き込む			
		STOP 検出割込みフラグ	STOPF	STOPIE	STOPCF = 1 を書き込む			
		転送完了再ロード	TCR	TCIE	I2C_CR2 の NBYTES[7:0] に 0 以外 を書き込む			
		転送完了	TC		START = 1 または STOP = 1 を書き込む			
		アドレス一致	ADDR	ADDRIE	ADDRCF = 1 を書き込む		可能(1)	
		NACK 受信	NACKF	NACKIE	NACKCF = 1 を書き込む		不可	
	I2C_ER	バス･エラー	BERR	ERRIE	BERRCF = 1 を書き込む	可能	不可	不可
		アービトレーション喪失	ARLO		ARLOCF = 1 を書き込む			
		オーバーラン / アンダーラン	OVR		OVRCF = 1 を書き込む			
		PEC エラー	PECERR		PECERRCF = 1 を書き込む			
		タイムアウト／t_{LOW} エラー	TIMEOUT		TIMEOUTCF=1 を書き込む			
		SMBus アラート	ALERT		ALERTCF = 1 を書き込む			

1. ADDR 一致イベントは、I²C インスタンスが STOP モードからのウェイクアップ機能をサポートしている場合にのみデバイスを STOP モードからウェイクアップできます。セクション 41.3 : I2C の実装を参照してください。

41.7 I²C レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル·レジスタは、ワード（32 ビット）単位でアクセスされます。

41.7.1 I²C 制御レジスタ 1（I2C_CR1）

アドレス·オフセット：0x00

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLKです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERT EN	SMBD EN	SMBH EN	GCEN	WUPEN	NOSTR ETCH	SBC
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMAE N	TXDMAE N	Res.	ANF OFF	DNF[3:0]				ERRIE	TCIE	STOP IE	NACK IE	ADDR IE	RXIE	TXIE	PE
rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **PECEN**：PEC 有効

0：PEC 計算は無効です。

1：PEC 計算は有効です。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 22 **ALERTEN**：SMBus アラート有効

0：SMBus アラートピン（SMBA）はホストモード（SMBHEN=1）でサポートされません。デバイスモード（SMBHEN=0）では、SMBA ピンが解放され、アラート応答アドレスヘッダが無効になります（0001100x の後には NACK が続きます）。

1：SMBus アラートピンはホストモード（SMBHEN=1）でサポートされます。デバイスモード（SMBHEN=0）では、SMBA ピンがローに駆動され、アラート応答アドレスヘッダが有効になります（0001100x の後には ACK が続きます）。

注： ALERTEN=0 のときには、SMBA ピンを標準 GPIO として使用できます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 21 **SMBDEN**：SMBus デバイス·デフォルト·アドレス有効

0：デバイス·デフォルト·アドレスを無効化します。アドレス 0b1100001x は NACK されます。

1：デバイス·デフォルト·アドレスを有効化します。アドレス 0b1100001x は ACK されます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 20 **SMBHEN**：SMBus ホスト·アドレス有効

0：ホスト·アドレスを無効化します。アドレス 0b0001000x は NACK されます。

1：ホスト·アドレスを有効化します。アドレス 0b0001000x は ACK されます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 19 **GCEN**：同報有効

0：同報は無効です。アドレス 0b00000000 は NACK されます。
1：同報は有効です。アドレス 0b00000000 は ACK されます。

ビット 18 **WUPEN**：STOP モードからのウェイクアップ有効

0：STOP モードからのウェイクアップ無効。
1：STOP モードからのウェイクアップ有効。

注： STOP モードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

注： WUPEN は、DNF = 0000 のときのみセットできます。

ビット 17 **NOSTRETCH**：クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。マスタモードではクリアされたままでなければなりません。

0：クロックストレッチ有効
1：クロックストレッチ無効

注： このビットは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 16 **SBC**：スレーブ･バイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

0：スレーブバイト制御無効
1：スレーブバイト制御有効

ビット 15 **RXDMAEN**：DMA 受信リクエスト有効

0：DMA モードは受信に無効
1：DMA モードは受信に有効

ビット 14 **TXDMAEN**：DMA 送信リクエスト有効

0：DMA モードは送信に無効
1：DMA モードは送信に有効

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **ANFOFF**：アナログ･ノイズ･フィルタ OFF

0：アナログノイズフィルタ有効
1：アナログノイズフィルタ無効

注： このビットは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 11:8 **DNF[3:0]**：デジタルノイズフィルタ

これらのビットは、SDA および SCL 入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大 DNF[3:0] * t_{I2CCLK} の長さのスパイクを除去します。

0000：デジタルフィルタ無効
0001：デジタルフィルタは有効であり、最大 1 t_{I2CCLK} の除去能力を持ちます。
.....
1111：デジタルフィルタは有効であり、最大 15 t_{I2CCLK} の除去能力を持ちます。

注： アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。このフィルタは、I²C が無効（PE = 0）のときのみプログラムできます。

ビット 7 **ERRIE**：エラー割込み有効

0：エラー検出割込み無効
1：エラー検出割込み有効

注： 次のようなエラーが発生すると、割込みが生成されます：
アービトレーション喪失（ARLO）
バスエラー検出（BERR）
オーバーラン／アンダーラン（OVR）
タイムアウト検出（TIMEOUT）
PEC エラー検出（PECERR）
アラートピンイベント検出（ALERT）

ビット 6 **TCIE**：転送完了割込み有効

0：転送完了割込み無効
1：転送完了割込み有効

注： 次のようなイベントが発生すると、割込みが生成されます：
転送完了（TC）
転送完了再ロード（TCR）

ビット 5 **STOPIE**：STOP 検出割込み有効

0：STOP 検出（STOPF）割込み無効
1：STOP 検出（STOPF）割込み有効

ビット 4 **NACKIE**：否定応答受信割込み有効

0：否定応答（NACKF）受信割込み無効
1：否定応答（NACKF）受信割込み有効

ビット 3 **ADDRIE**：アドレス一致割込み有効（スレーブのみ）

0：アドレス一致（ADDR）割込み無効
1：アドレス一致（ADDR）割込み有効

ビット 2 **RXIE**：RX 割込み有効

0：受信（RXNE）割込み無効
1：受信（RXNE）割込み有効

ビット 1 **TXIE**：TX 割込み有効

0：送信（TXIS）割込み無効
1：送信（TXIS）割込み有効

ビット 0 **PE**：ペリフェラル有効

0：ペリフェラルは無効です。
1：ペリフェラルは有効です。

注： PE = 0 のとき、I2C の SCL および SDA ラインはリリースされます。内部ステート･マシンおよびステータス･ビットはリセット値に戻されます。クリアされたときには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たれる必要があります。

41.7.2 I²C 制御レジスタ 2（I2C_CR2）

アドレス·オフセット：0x04

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLKです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	PEC BYTE	AUTOEND	RELOAD	NBYTES[7:0]							
					rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
rs	rs	rs	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **PECBYTE**：パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PEC が転送されたとき、STOP コンディションあるいはアドレス一致を受信したとき、または PE = 0 になったとき、ハードウェアによってクリアされます。

0：PEC 転送なし。

1：PEC 送信／受信がリクエストされます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、RELOAD がセットされているときには効果がありません。

このビットは、SBC=0 のとき、スレーブモードでは効果がありません。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 25 **AUTOEND**：自動終了モード（マスタモード）

このビットは、ソフトウェアによってセット／クリアされます。

0：ソフトウェア終了モード：NBYTES データが転送されると TC フラグがセットされ、SCL ローをストレッチします。

1：自動終了モード：NBYTES データが転送されると、STOP コンディションが自動的に送信されます。

注： このビットは、スレーブモードまたは RELOAD ビットがセットされているときには効果がありません。

ビット 24 **RELOAD**：NBYTES 再ロード·モード

このビットは、ソフトウェアによってセット／クリアされます。

0：転送は、NBYTES データの転送後、完了します（STOP または RESTART が続きます）。

1：転送は、NBYTES データの転送後に完了しません（NBYTES が再ロードされます）。NBYTES データが転送されると TCR フラグがセットされ、SCL ローをストレッチします。

ビット 23:16 **NBYTES[7:0]**：バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC=0 のスレーブモードでは効果がありません。

注： START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 15 **NACK**：NACK 生成（スレーブ･モード）

このビットはソフトウェアによってセットされ、NACK が送信されたとき、STOP コンディションあるいはアドレス一致を受信したとき、または PE = 0 になったとき、ハードウェアによってクリアされます。

0：現在の受信バイト後に ACK が送信されます。

1：現在の受信バイト後に NACK が送信されます。

注： このビットに 0 を書き込んでも、効果はありません。

このビットは、スレーブモードでのみ使用されます：マスタレシーバモードでは、NACK ビットの値にかかわらず、STOP または RESTART コンディション前の最後のバイト後に NACK が自動的に生成されます。

スレーブレシーバ NOSTRETCH モードでオーバーランが発生すると、NACK ビットの値にかかわらず、NACK が自動的に生成されます。

ハードウェア PEC チェックが有効なとき（PECBYTE=1）、PEC 確認応答値は NACK 値に依存しません。

ビット 14 **STOP**：STOP 生成（マスタ･モード）

このビットはソフトウェアによってセットされ、STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

マスタモード：

0：STOP 生成なし。

1：現在のバイト転送後の STOP 生成。

注： このビットに 0 を書き込んでも、効果はありません。

ビット 13 **START**：START 生成

このビットはソフトウェアによってセットされ、START とアドレスシーケンスが送信された後、アービトレーション喪失によって、タイムアウトエラー検出によって、または PE=0 のときに、ハードウェアによってクリアされます。I2C_ICR レジスタの ADDRCF ビットに "1" を書き込むことによって、ソフトウェアによってクリアすることもできます。

0：START 生成なし。

1：RESTART/START生成：

I2C がすでにマスタモードであり、AUTOEND=0 の場合、このビットをセットすると、RELOAD=0 のとき、NBYTES 転送の終了後に REPEATED START コンディションが生成されます。

そうでない場合、バスがフリーになると、このビットをセットすることで START コンディションが生成されます。

注： このビットに 0 を書き込んでも、効果はありません。

バスが BUSY の場合、または I²C がスレーブモードの場合でも、START ビットをセットできます。

このビットは、RELOAD がセットされているときには効果がありません。

ビット 12 **HEAD10R**：読出し方向のみの 10 ビットアドレスヘッダ（マスタレシーバモード）

0：マスタは完全な 10 ビットスレーブアドレス読出しシーケンスを送信します：START + 2 バイトの書込み方向の 10 ビットアドレス + RESTART + 読出し方向の 10 ビットアドレスの最初の 7 ビット。

1：マスタは 10 ビットアドレスの最初の 7 ビットのみを送信し、その後に読出し方向を送信します。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 11 **ADD10**：10 ビットアドレッシングモード（マスタモード）

0：マスタは 7 ビットアドレッシングモードで動作します。

1：マスタは 10 ビットアドレッシングモードで動作します。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 10 **RD_WRN**：転送方向（マスタモード）

0：マスタは書込み転送をリクエストします。

1：マスタは読出し転送をリクエストします。

注： START ビットがセットされているときに、このビットを変更することはできません。

ビット 9:0 **SADD[9:0]**：スレーブアドレス（マスタモード）

7 ビットアドレッシングモード（ADD10=0）：

SADD[7:1] には、送信される 7 ビットのスレーブアドレスを書き込みます。SADD[9]、SADD[8]、および SADD[0] ビットは無視されます。

10 ビットアドレッシングモード（ADD10=1）：

SADD[9:0] には、送信される 10 ビットのスレーブアドレスを書き込みます。

注： START ビットがセットされているときに、これらのビットを変更することはできません。

41.7.3 I²C Own Address 1 レジスタ（I2C_OAR1）

アドレス･オフセット：0x08

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLKです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA1EN	Res.	Res.	Res.	Res.	OA1 MODE	OA1[9:0]									
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA1EN**：Own Address 1 有効

0：Own Address 1 は無効です。受信されたスレーブアドレス OA1 は NACK されます。

1：Own Address 1 は有効です。受信されたスレーブアドレス OA1 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **OA1MODE**：Own Address 1 10 ビットモード

0：Own Address 1 は 7 ビットアドレスです。

1：Own Address 1 は 10 ビットアドレスです。

注： このビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 9:0 **OA1[9:0]**：インタフェース専用スレーブアドレス

7 ビットアドレスモード：OA1[7:1] には 7 ビットの専用スレーブアドレスが含まれます。OA1[9]、OA1[8]、および OA1[0] ビットは無視されます。

10 ビットアドレスモード：OA1[9:0] には 10 ビットの専用スレーブアドレスが含まれます。

注： これらのビットは、OA1EN=0 のときのみ書き込むことができます。

41.7.4 I²C Own Address 2 レジスタ（I2C_OAR2）

アドレス･オフセット：0x0C

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLKです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2 [7:1]							Res.
rw					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **OA2EN**：Own Address 2 有効

0：Own Address 2 は無効です。受信されたスレーブアドレス OA2 は NACK されます。

1：Own Address 2 は有効です。受信されたスレーブアドレス OA2 は ACK されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 **OA2MSK[2:0]**：Own Address 2 マスク

000：マスクなし。

001：OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。

010：OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。

011：OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。

100：OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。

101：OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。

110：OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。

111：OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7 ビット受信アドレスが確認応答されます。

注： これらのビットは、OA2EN=0 のときのみ書き込むことができます。

OA2MSK が 0 でなくなると、予約済み I²C アドレス（0b0000xxx および 0b1111xxx）は、比較が一致した場合でも確認応答されません。

ビット 7:1 **OA2[7:1]**：インタフェースアドレス

7 ビットアドレスモード：7 ビットアドレス

注： これらのビットは、OA2EN=0 のときのみ書き込むことができます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

41.7.5 I²C タイミングレジスタ（I2C_TIMINGR）

アドレス·オフセット：0x10

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL[3:0]				SDADEL[3:0]			
rw	rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCLH[7:0]								SCLL[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:28 **PRESC[3:0]**：タイミングプリスケーラ

このフィールドは、データのセットアップおよびホールドカウンタ（1831 ページのI2C のタイミングを参照）と SCL ハイおよびローレベルカウンタ（1846 ページのI2C マスタ初期化を参照）に使用されるクロック周期 t_{PRESC} を生成して、I2CCLK をプリスケールするために使用されます。

t_{PRESC} = (PRESC+1) x t_{I2CCLK}

ビット 27:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **SCLDEL[3:0]**：データセットアップ時間

このフィールドは、SDA エッジと SCL 立ち上がりエッジの間に遅延 t_{SCLDEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SCLDEL} の間ローにストレッチされます。

t_{SCLDEL} = (SCLDEL+1) x t_{PRESC}

注： t_{SCLDEL} は、$t_{SU:DAT}$ タイミングを生成するために使用されます。

ビット 19:16 **SDADEL[3:0]**：データホールド時間

このフィールドは、SCL 立ち下がりエッジと SDA エッジの間に遅延 t_{SDADEL} を生成するために使用されます。NOSTRETCH = 0 でのマスタモードおよびスレーブモードでは、SCL ラインは、t_{SDADEL} の間ローにストレッチされます。

t_{SDADEL}= SDADEL x t_{PRESC}

注： SDADEL は、$t_{HD:DAT}$ タイミングを生成するために使用されます。

ビット 15:8 **SCLH[7:0]**：SCL ハイ周期（マスタモード）

このフィールドは、マスタモードで SCL ハイ周期を生成するために使用されます。

t_{SCLH} = (SCLH+1) x t_{PRESC}

注： SCLH は、$t_{SU:STO}$ および $t_{HD:STA}$ タイミングを生成するためにも使用されます。

ビット 7:0 **SCLL[7:0]**：SCL ロー周期（マスタモード）

このフィールドは、マスタモードで SCL ロー周期を生成するために使用されます。

t_{SCLL} = (SCLL+1) x t_{PRESC}

注： SCLL は、t_{BUF} および $t_{SU:STA}$ タイミングを生成するためにも使用されます。

注： このレジスタは、I²C が無効（PE=0）のときに設定する必要があります。

注： STM32CubeMX ツールは、I²C 設定ウィンドウの I2C_TIMINGR コンテンツを計算し、提供します。

41.7.6 I²C タイムアウトレジスタ（I2C_TIMEOUTR）

アドレス·オフセット：0x14

リセット値：0x0000 0000

アクセス：このレジスタへの書込みアクセスが進行中のときに書込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書込みアクセスが完了するまで、2 番目の書込みアクセスにウェイト状態が挿入されます。2 番目の書込みアクセスの遅延は、最大 2 x PCLK1 + 6 x I2CCLKです。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]											
rw				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **TEXTEN**：拡張クロック·タイムアウト有効

0：拡張クロックタイムアウト検出は無効です。

1：拡張クロックタイムアウト検出は有効です。$t_{LOW:EXT}$ を超える累積 SCL ストレッチが I²C インタフェースによって行われると、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 30:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:16 **TIMEOUTB[11:0]**：バスタイムアウト B

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます：

マスタモードでは、マスタ累積クロックロー拡張時間（$t_{LOW:MEXT}$）が検出されます。

スレーブモードでは、スレーブ累積クロックロー拡張時間（$t_{LOW:SEXT}$）が検出されます。

$t_{LOW:EXT}$ = (TIMEOUTB+1) x 2048 x t_{I2CCLK}

注： これらのビットは、TEXTEN=0 のときのみ書き込むことができます。

ビット 15 **TIMOUTEN**：クロックタイムアウト有効

0：SCL タイムアウト検出は無効です。

1：SCL タイムアウト検出は有効です。SCL が $t_{TIMEOUT}$（TIDLE=0）を超えてローであるか、t_{IDLE}（TIDLE=1）を超えてハイであった場合、タイムアウトエラーが検出されます（TIMEOUT=1）。

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **TIDLE**：アイドル·クロック·タイムアウト検出

0：TIMEOUTA は、SCL ロータイムアウトの検出に使用されます。

1：TIMEOUTA は、SCL と SDA の両方のハイタイムアウト（バスアイドル条件）の検出に使用されます。

注： このビットは、TIMOUTEN=0 のときのみ書き込むことができます。

ビット 11:0 **TIMEOUTA[11:0]**：バスタイムアウト A

このフィールドは、以下を設定するために使用されます：

TIDLE=0 のときの SCL ロータイムアウト条件 $t_{TIMEOUT}$

$t_{TIMEOUT}$ = (TIMEOUTA+1) x 2048 x t_{I2CCLK}

TIDLE=1 のときのバスアイドル条件（SCL と SDA の両方のハイ）

t_{IDLE} = (TIMEOUTA+1) x 4 x t_{I2CCLK}

注： これらのビットは、TIMOUTEN=0 のときのみ書き込むことができます。

注： SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 41.3：I2C の実装を参照してください。

41.7.7 I²C 割込みおよびステータスレジスタ（I2C_ISR）

アドレス·オフセット：0x18

リセット値：0x0000 0001

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.	ALERT	TIME OUT	PEC ERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
r		r	r	r	r	r	r	r	r	r	r	r	r	rs	rs

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:17 **ADDCODE[6:0]**：アドレス一致コード（スレーブモード）

これらのビットは、アドレス一致イベントが発生したときに（ADDR = 1）、受信したアドレスで更新されます。

10 ビットアドレスの場合、ADDCODE は 10 ビットのヘッダとその後のアドレスの 2 つの MSB を示します。

ビット 16 **DIR**：転送方向（スレーブ·モード）

このフラグは、アドレス一致イベントが発生したときに（ADDR = 1）更新されます。

0：書込み転送、スレーブはレシーバモードになります。

1：読出し転送、スレーブはトランスミッタモードになります。

ビット 15 **BUSY**：バス·ビジー

このフラグは、バスで通信が進行中であることを示します。START コンディションが検出されたときに、ハードウェアによってセットされます。STOP コンディションが検出されたとき、または PE = 0 になったときにハードウェアによってクリアされます。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERT**：SMBus アラート

このフラグは、SMBHEN=1（SMBus ホスト設定）、ALERTEN=1、および SMBALERT イベント（立ち下がりエッジ）が SMBA ピンで検出されたときに、ハードウェアによってセットされます。ALERTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 12 **TIMEOUT**：タイムアウトまたは t_{LOW} 検出フラグ

このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注：　このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 11 **PECERR**：受信時の PEC エラー

このフラグは、受信した PEC が PEC レジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくない PEC の受信後、NACK が自動的に送信されます。PECCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 10 **OVR**：オーバーラン／アンダーラン（スレーブ･モード）

このフラグは、NOSTRETCH = 1 のスレーブ･モードで、オーバーラン／アンダーラン･エラーが発生したときに、ハードウェアによってセットされます。OVRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 9 **ARLO**：アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 8 **BERR**：バス･エラー

このフラグは、ペリフェラルが転送に関与しているので、START または STOP コンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 7 **TCR**：転送完了再ロード

このフラグは、RELOAD=1 および NBYTES データが転送されたときに、ハードウェアによってセットされます。NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

このフラグは、マスタ･モード、または SBC ビットがセットされているときのスレーブ･モードでのみ使用されます。

ビット 6 **TC**：転送完了（マスタ･モード）

このフラグは、RELOAD=0、AUTOEND=0、および NBYTES データが転送されたときに、ハードウェアによってセットされます。START ビットまたは STOP ビットがセットされたときに、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 5 **STOPF**：STOP 検出フラグ

このフラグは、バス上で STOP コンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます：

– マスタとして。ただし、STOP コンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 4 **NACKF**：否定応答受信フラグ

このフラグは、バイト送信後に NACK を受信したときに、ハードウェアによってセットされます。NACKCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 3 **ADDR**：アドレス一致（スレーブ･モード）フラグ

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの 1 つに一致したときに、ハードウェアによってセットされます。ADDRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 2 **RXNE**：受信データ･レジスタ･ノット･エンプティ（レシーバ）

このビットは、受信データが I2C_RXDR レジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2C_RXDR が読み出されたときにクリアされます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 1 **TXIS**：送信割込みステータス（トランスミッタ）

このビットは、I2C_TXDR レジスタが空であり、送信データを I2C_TXDR レジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

NOSTRETCH = 1 の場合にのみ、このビットにソフトウェアで 1 を書き込んで、TXIS イベント（TXIE = 1 の場合は割込み、TXDMAEN = 1 の場合は DMA リクエスト）を生成できます。

注： このビットは、PE = 0 になったとき、ハードウェアによってクリアされます。

ビット 0 **TXE**：送信データ･レジスタ･エンプティ（トランスミッタ）

このビットは、I2C_TXDR レジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2C_TXDR レジスタに書き込まれたときにクリアされます。

このビットは、ソフトウェアによって 1 を書き込んで、送信データレジスタ I2C_TXDR をFlashできます。

注： このビットは、PE = 0 になったとき、ハードウェアによってセットされます。

41.7.8 I²C 割込みクリア･レジスタ（I2C_ICR）

アドレス･オフセット：0x1C

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
		w	w	w	w	w	w			w	w	w			

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **ALERTCF**：アラートフラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの ALERT フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 12 **TIMOUTCF**：タイムアウト検出フラグクリア

このビットに 1 を書き込むと、I2C_ISR レジスタの TIMEOUT フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 11 **PECCF**：PEC エラー･フラグ･クリア

このビットに 1 を書き込むと、I2C_ISR レジスタの PECERR フラグがクリアされます。

注： SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 41.3：I2C の実装を参照してください。

ビット 10 **OVRCF**：オーバーラン／アンダーラン･フラグ･クリア

このビットに 1 を書き込むと、I2C_ISR レジスタの OVR フラグがクリアされます。

ビット 9 **ARLOCF**：アービトレーション喪失フラグ･クリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ARLO フラグがクリアされます。

ビット 8 **BERRCF**：バス･エラー･フラグ･クリア
このビットに 1 を書き込むと、I2C_ISR レジスタの BERRF フラグがクリアされます。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **STOPCF**：STOP 検出フラグ･クリア
このビットに 1 を書き込むと、I2C_ISR レジスタの STOPF フラグがクリアされます。

ビット 4 **NACKCF**：非確認応答フラグ･クリア
このビットに 1 を書き込むと、I2C_ISR レジスタの NACKF フラグがクリアされます。

ビット 3 **ADDRCF**：アドレス一致フラグ･クリア
このビットに 1 を書き込むと、I2C_ISR レジスタの ADDR フラグがクリアされます。このビットに 1 を書き込むと、I2C_CR2 レジスタの START ビットもクリアされます。

ビット 2:0 予約済みであり、リセット値に保持する必要があります。

41.7.9 I²C PEC レジスタ（I2C_PECR）

アドレス･オフセット：0x20

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **PEC[7:0]**：パケット･エラー･チェック･レジスタ
PECEN = 1 のとき、このフィールドは内部 PEC を含みます。
PEC は、PE = 0 になったとき、ハードウェアによってクリアされます。

注： **SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 41.3：I2C の実装を参照してください。**

41.7.10 I²C 受信データレジスタ（I2C_RXDR）

アドレス･オフセット：0x24

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXDATA[7:0]：**8 ビットの受信データ

I²C バスから受信したデータ･バイト。

41.7.11 I²C 送信データレジスタ（I2C_TXDR）

アドレス･オフセット：0x28

リセット値：0x0000 0000

アクセス：ウェイト状態なし

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXDATA[7:0]：**8 ビットの送信データ

I²C バスに送信されるデータ･バイト。

注： これらのビットは、TXE = 1 のときのみ書き込むことができます。

41.7.12 I²C レジスタ･マップ

次の表に、I2C のレジスタ･マップとリセット値を示します。

表 394. I²C レジスタマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0	I2C_CR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PECEN	ALERTEN	SMBDEN	SMBHEN	GCEN	WUPEN	NOSTRETCH	SBC	RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
	リセット値									0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0
0x4	I2C_CR2	Res.	Res.	Res.	Res.	Res.	PECBYTE	AUTOEND	RELOAD	NBYTES[7:0]								NACK	STOP	START	HEAD10R	ADD10	RD_WRN	SADD[9:0]									
	リセット値						0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x8	I2C_OAR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA1EN	Res.	Res.	Res.	Res.	OA1MODE	OA1[9:0]									
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	0
0xC	I2C_OAR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	OA2EN	Res.	Res.	Res.	Res.	OA2MSK [2:0]			OA2[7:1]							Res.
	リセット値																	0					0	0	0	0	0	0	0	0	0	0	
0x10	I2C_ TIMINGR	PRESC[3:0]				Res.	Res.	Res.	Res.	SCLDEL [3:0]				SDADEL [3:0]				SCLH[7:0]								SCLL[7:0]							
	リセット値	0	0	0	0					0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	I2C_ TIMEOUTR	TEXTEN	Res.	Res.	Res.	TIMEOUTB[11:0]												TIMOUTEN	Res.	Res.	TIDLE	TIMEOUTA[11:0]											
	リセット値	0				0	0	0	0	0	0	0	0	0	0	0	0	0			0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	I2C_ISR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ADDCODE[6:0]							DIR	BUSY	Res.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE
	リセット値									0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x1C	I2C_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ALERTCF	TIMOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.	Res.	STOPCF	NACKCF	ADDRCF	Res.	Res.	Res.
	リセット値																			0	0	0	0	0	0			0	0	0			
0x20	I2C_PECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PEC[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x24	I2C_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 394. I²C レジスタマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x28	I2C_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

42 独立型ウォッチドッグ（IWDG）

42.1 概要

デバイス は、内蔵ウォッチドッグペリフェラルを搭載しており、使用上、高い安全レベル、タイミングの正確さ、および柔軟性を兼ね備えています。独立型ウォッチドッグペリフェラルは、ソフトウェア障害による誤動作を検出および解決し、カウンタが所定のタイムアウト値に達すると、システムリセットをトリガします。

独立型ウォッチドッグ（IWDG）は、独自の低速クロック（LSI）によってクロック供給されるので、メインクロックに障害があってもアクティブなままです。

IWDG は、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、タイミング精度が低いという制約があります。ウィンドウ型ウォッチドッグの詳細については、1902ページのセクション 43 を参照してください。

42.2 IWDG の主な機能

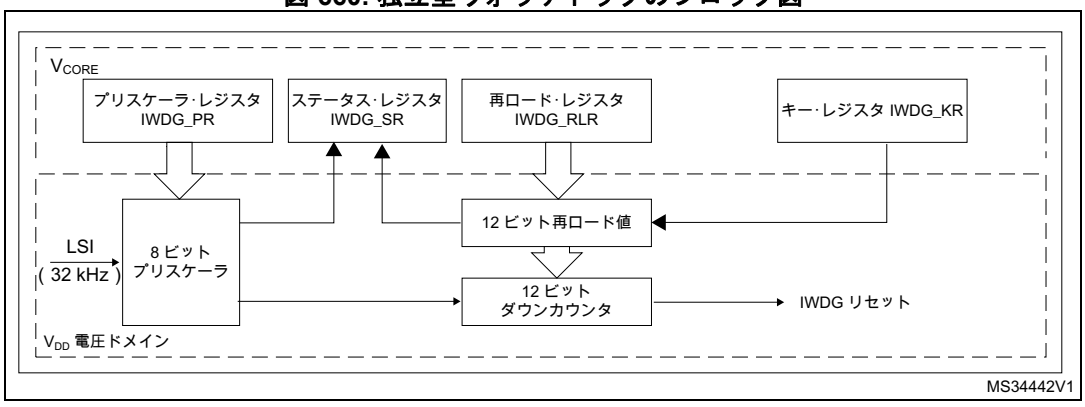
- フリーランニングダウンカウンタ
- 独立した RC オシレータからのクロック供給（STANDBY および STOP モードで動作可能）
- 条件付きリセット
  - ダウンカウンタの値が 0x000 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）。

42.3 IWDG の機能説明

42.3.1 IWDG ブロック図

図 660 に、独立型ウォッチドッグ･モジュールの機能ブロックを示します。

図 660. 独立型ウォッチドッグのブロック図



1. レジスタインタフェースは電圧ドメインにあります。ウォッチドッグ機能は、STANDBY モードでも機能する V_{DD} 電圧ドメインにあります。

IWDG キーレジスタ（IWDG_KR）に値 0x0000 CCCC が書き込まれることによって独立型ウォッチドッグが開始すると、カウンタはリセット値 0xFFF からカウントダウンを開始します。カウント値の終わり（0x000）に達すると、リセット信号が生成されます（IWDG_reset）。

IWDG キーレジスタ（IWDG_KR）にキー値 0x0000 AAAA が書き込まれると、IWDG_RLR の値がカウンタに再ロードされ、ウォッチドッグのリセットが防止されます。

一度起動すると、IWDG は停止できません。

42.3.2 ウィンドウオプション

IWDG は、IWDG ウィンドウレジスタ（IWDG_WINR）に適切なウィンドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタがIWDG ウィンドウレジスタ（IWDG_WINR）に格納された値より大きい間に再ロード操作が行われると、リセットが生成されます。

IWDG ウィンドウレジスタ（IWDG_WINR）のデフォルト値は 0x0000 0FFF です。この値が更新されない場合は、ウィンドウオプションは無効にされます。

ウィンドウ値が変わるとすぐに再ロード操作が行われ、ダウンカウンタをIWDG 再ロードレジスタ（IWDG_RLR）値にリセットし、次の再ロードを生成するためのサイクル数計算を容易にします。

ウィンドウオプションが有効な場合の IWDG の設定

1. IWDG キーレジスタ（IWDG_KR）に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. IWDG キーレジスタ（IWDG_KR）に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG プリスケーラレジスタ（IWDG_PR）を 0 から 7 までプログラムすることによって、IWDG のプリスケーラに書込みを行います。
4. IWDG 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. IWDG ウィンドウレジスタ（IWDG_WINR）に書き込みます。これにより、IWDG 再ロードレジスタ（IWDG_RLR）のカウンタ値が自動的にリフレッシュされます。

注： **ウィンドウ値を書き込むことで、IWDG ステータスレジスタ（IWDG_SR）が “0x0000 0000”にセットされた時点でカウンタ値を RLR でリフレッシュすることができます。**

ウィンドウオプションが無効な場合の IWDG の設定

ウィンドウオプションが使用されていない場合、IWDG は以下のように設定することができます。

1. IWDG キーレジスタ（IWDG_KR）に 0x0000 CCCC を書き込むことによって、IWDG を有効にします。
2. IWDG キーレジスタ（IWDG_KR）に 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG プリスケーラレジスタ（IWDG_PR）を 0 から 7 までプログラムすることによって、プリスケーラに書込みを行います。
4. IWDG 再ロードレジスタ（IWDG_RLR）に書き込みます。
5. レジスタが更新されるのを待ちます（IWDG_SR = 0x0000 0000）。
6. カウンタ値を IWDG_RLR（IWDG_KR = 0x0000 AAAA）でリフレッシュします。

42.3.3 ハードウェアウォッチドッグ

デバイスのオプションビットを使って「ハードウェアウォッチドッグ」機能が有効化されると、ウォッチドッグは電源投入時に自動的に有効になり、カウンタがカウントの終わりに達する前にソフトウェアによってIWDG キーレジスタ（IWDG_KR）へ書き込まれない限り、またはダウンカウンタがウィンドウ外で再ロードされた場合は、リセットを生成します。

42.3.4 低電力モード

IWDG_STOP および IWDG_STBY のオプション設定に応じて、IWDG では STOP モード中および STANDBY モード中にそれぞれカウントを継続または停止できます。STOP または STANDBY モード中に IWDG を実行したままにすると、このモードからデバイスをウェイクアップできます。詳細については、ユーザ·オプション·バイトおよび読出し保護オプション·バイトを参照してください。

42.3.5 レジスタのアクセス保護

IWDG プリスケーラレジスタ（IWDG_PR）、IWDG 再ロードレジスタ（IWDG_RLR）、およびIWDG ウィンドウレジスタ（IWDG_WINR）への書込みアクセスは保護されます。これらを変更するには、まず、IWDG キーレジスタ（IWDG_KR）にコード 0x0000 5555 を書き込む必要があります。このレジスタに別の値で書込みアクセスすると、シーケンスがブレークされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作（0x0000 AAAA の書込み）の場合に相当します。

ステータスレジスタは、プリスケーラの更新、あるいはダウンカウンタ再ロード値の更新やウィンドウ値の更新が行われていることを示すために使用されます。

42.3.6 デバッグモード

デバイスがデバッグモードになると（コアは停止状態）、IWDG カウンタは、DBGMCU フリーズレジスタの対応するビットの設定に応じて、通常どおりに動作を続けるか、または停止します。

42.4 IWDG レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル·レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

42.4.1 IWDG キーレジスタ（IWDG_KR）

アドレスオフセット：0x00

リセット値：0x0000 0000（STANDBY モードによりリセットされる）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **KEY[15:0]**：キー値（書込み専用、読出しでは 0x0000）

これらのビットには、ソフトウェアによって一定間隔でキー値 0xAAAA が書き込まれなければなりません。そうしないと、カウンタが 0 に達した時点でウォッチドッグがリセットを生成します。

キー値 0x5555 を書き込むことによって、IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへのアクセスが可能になります（セクション 42.3.5：レジスタのアクセス保護を参照）。

キー値 0xCCCC を書き込むと、ウォッチドッグが開始します（ハードウェアウォッチドッグオプションが選択されている場合を除く）。

42.4.2 IWDG プリスケーラレジスタ（IWDG_PR）

アドレスオフセット：0x04

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PR[2:0]**：プリスケーラ分周回路

これらのビットは、書込みアクセス保護されています（セクション 42.3.5 : レジスタのアクセス保護を参照）。カウンタクロックを供給するプリスケーラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケーラ分周回路を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の PVU ビットがリセットされる必要があります。

000：4 分周
001：8 分周
010：16 分周
011：32 分周
100：64 分周
101：128 分周
110：256 分周
111：256 分周

注： このレジスタを読み出すと、V_{DD} 電圧ドメインからプリスケーラ値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の PVU ビットがリセットされているときのみとなります。

42.4.3 IWDG 再ロードレジスタ（IWDG_RLR）

アドレスオフセット：0x08

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	RL[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **RL[11:0]**：ウォッチドッグカウンタ再ロード値

これらのビットは、書込みアクセス保護されています（ レジスタのアクセス保護を参照）。IWDG キーレジスタ（IWDG_KR）に値 0xAAAA が書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とクロックプリスケーラによって決まります。タイムアウトに関する詳細はデータシートを参照してください。

再ロード値を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の RVU ビットがリセットされる必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の RVU ビットがリセットされているときのみとなります。

42.4.4 IWDG ステータスレジスタ（IWDG_SR）

アドレスオフセット：0x0C

リセット値：0x0000 0000（STANDBY モードによりリセットされません）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WVU	RVU	PVU
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **WVU**：ウォッチドッグ･カウンタ･ウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 サイクルかかります）。

ウィンドウ値は、WVU ビットがリセットされているときのみ更新できます。

ビット 1 **RVU**：ウォッチドッグ･カウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 サイクルかかります）。

再ロード値は、RVU ビットがリセットされているときのみ更新できます。

ビット 0 **PVU**：ウォッチドッグ･プリスケーラ値の更新

このビットは、プリスケーラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。V_{DD} 電圧ドメインでプリスケーラの更新操作が完了したときに、ハードウェアによってリセットされます（最大 5 サイクルかかります）。

プリスケーラ値は、PVU ビットがリセットされているときのみ更新できます。

注： **複数の再ロード値、プリスケーラ値、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前に RVU ビットがリセットされるまで待つか、プリスケーラ値を変更する前に PVU ビットがリセットされるまで待つか、またはウィンドウ値を変更する前に WVU ビットがリセットされるまで待つ必要があります。ただし、プリスケーラ値、再ロード値、またはウィンドウ値を更新した後は、RVU、PVU 、または WVU がリセットされるのを待たずに、コード実行を続けることができます（低電力モードに入った場合を除く）。**

42.4.5 IWDG ウィンドウレジスタ（IWDG_WINR）

アドレスオフセット：0x10

リセット値：0x0000 0FFF（STANDBY モードによりリセットされる）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	WIN[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WIN[11:0]**：ウォッチドッグカウンタウィンドウ値

これらのビットは書込みアクセス保護されており（セクション 42.3.5 : を参照）、ダウンカウンタと比較されるウィンドウ値の上限を格納しています。

リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x0 よりも大きい間にダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、IWDG ステータスレジスタ（IWDG_SR）の WVU ビットがリセットされる必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG ステータスレジスタ（IWDG_SR）の WVU ビットがリセットされているときのみとなります。

42.4.6 IWDG レジスタマップ

次の表に、IWDG レジスタマップとリセット値を示します。

表 395. IWDG レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**IWDG_KR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	KEY[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**IWDG_PR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PR[2:0]		
	リセット値																														0	0	0
0x08	**IWDG_RLR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RL[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1
0x0C	**IWDG_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WVU	RVU	PVU
	リセット値																														0	0	0
0x10	**IWDG_WINR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WIN[11:0]											
	リセット値																					1	1	1	1	1	1	1	1	1	1	1	1

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

43 システム･ウィンドウ型ウォッチドッグ（WWDG）

43.1 概要

システムウィンドウ型ウォッチドッグ（WWDG）は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。ウォッチドッグ回路は、T6 ビットがクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後にマイクロコントローラ･リセットを生成します。マイクロコントローラ･リセットは、ダウンカウンタがウィンドウ･レジスタ値に達する前に 7 ビットのダウンカウンタの値（制御レジスタ内）がリフレッシュされた場合にも生成されます。これは、限られた時間内にカウンタをリフレッシュする必要があることを意味します。

WWDG クロックは、APB クロックから分周され、また設定可能な時間枠（time-window）があるので、これをプログラムしてアプリケーション動作の異常な進み・遅れを検出できます。

WWDG は、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

43.2 WWDG の主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
  - ダウンカウンタの値が 0x40 より小さくなったときにリセット（ウォッチドッグが有効な場合）。
  - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット（ウォッチドッグが有効な場合）（図 662 を参照）。
- 早期ウェイクアップ割込み（EWI）: ダウンカウンタが 0x40 になったときにトリガ（有効であり、ウォッチドッグがアクティブな場合）

43.3 WWDG の機能説明

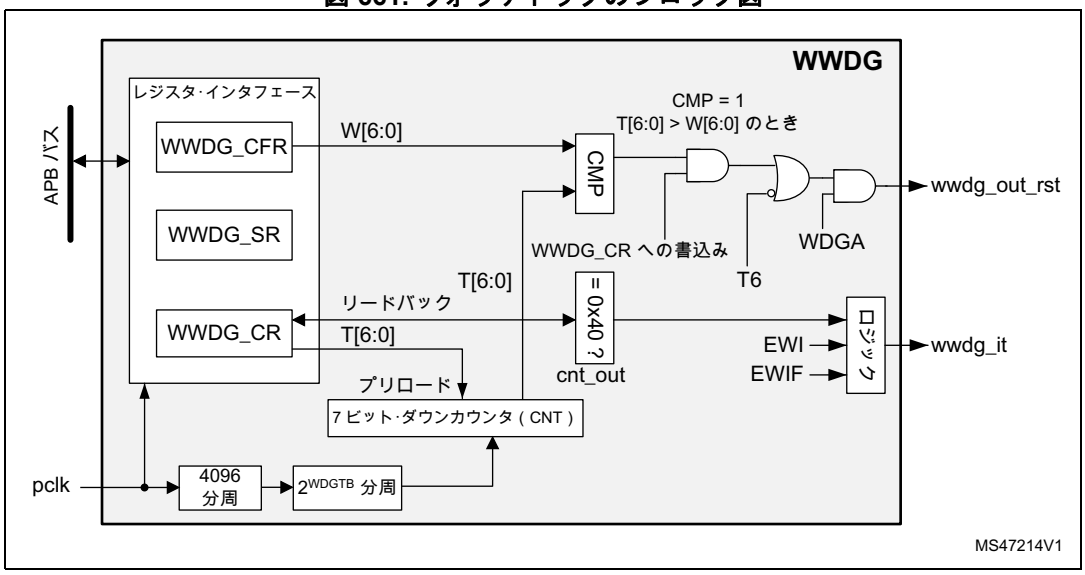
ウォッチドッグが有効な（WWDG_CR レジスタの WDGA ビットがセットされている）場合、7 ビットのダウンカウンタ（T[6:0] ビット）が 0x40 に達して 0x3F にデクリメントされた（T6 がクリアされた）時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

アプリケーションプログラムは､通常動作時には定期的に WWDG_CR レジスタへの書込みを行って、MCU リセットを防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値より小さいとき、かつ 0x3F より高いときに限られます。WWDG_CR レジスタに格納される値は、0xFF から 0xC0 の間でなければなりません。

WWDG のブロック図については、図 661 を参照してください。

43.3.1 WWDG ブロック図

図 661. ウォッチドッグのブロック図



43.3.2 ウォッチドッグの有効化

ユーザオプション WWDG_SW で “ソフトウェアウィンドウウォッチドッグ”を選択すると、リセット後、ウォッチドッグは常に無効です。これを有効にするには、WWDG_CR レジスタの WDGA ビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

ユーザオプション WWDG_SW で “ハードウェアウィンドウウォッチドッグ”を選択すると、リセット後、ウォッチドッグは常に有効であり、無効にできません。

43.3.3 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6 ビットをセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。このタイミングは、WWDG_CR レジスタへの書込み時のプリスケーラの状態が不明なので最小値から最大値の間で変化します（図 662 参照）。WWDG 設定レジスタ（WWDG_CFR）はウィンドウの上限値を含みます。リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x3F よりも大きい間にダウンカウンタを再ロードする必要があります。図 662 に、ウィンドウ型ウォッチドッグのプロセスを示します。

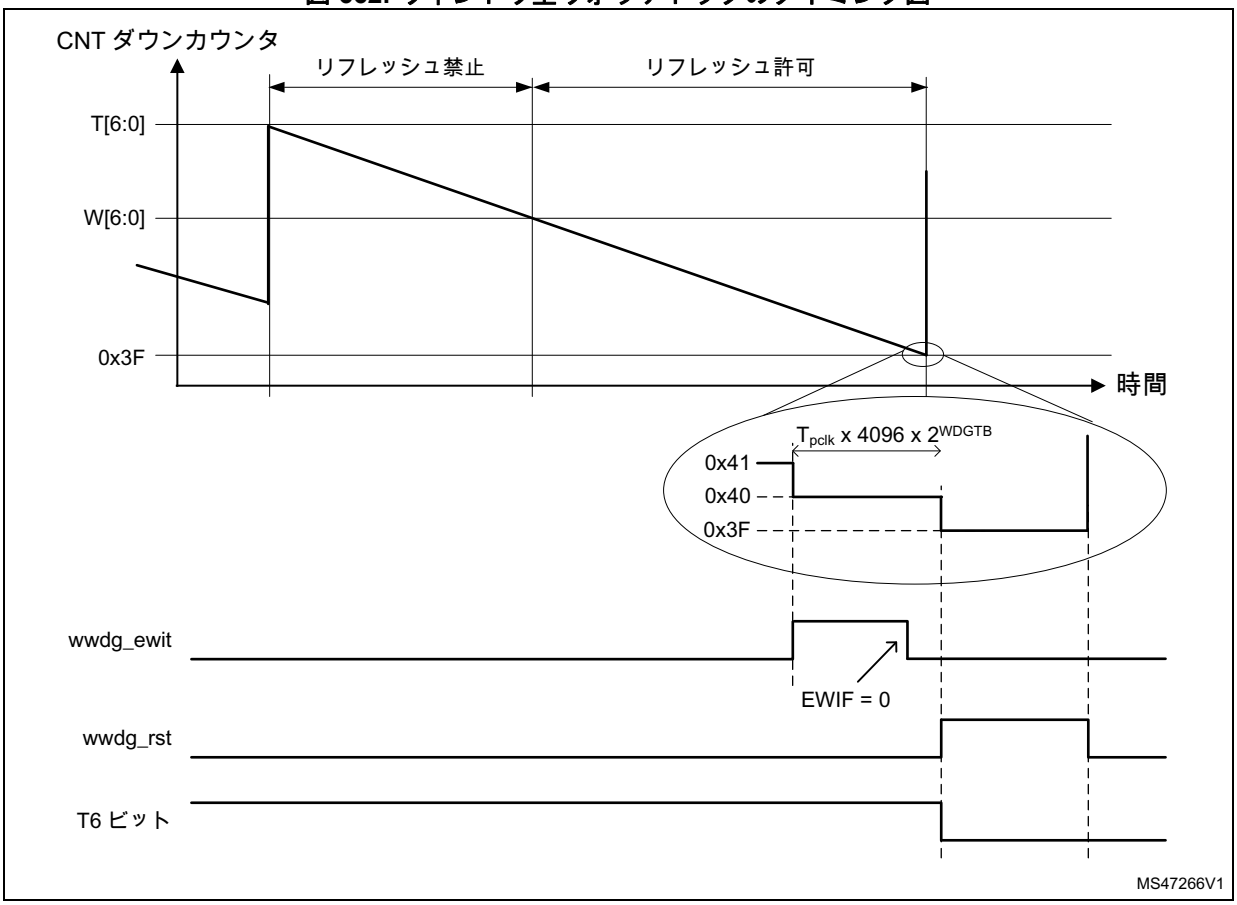
注： **T6 ビットを使用して、ソフトウェアリセットを生成することができます（WDGA ビットはセット、T6 ビットはクリアされる）。**

43.3.4 ウォッチドッグタイムアウトをプログラムする方法

図 662 の式を使用して、WWDG のタイムアウトを計算します。

警告： WWDG_CR レジスタに書き込むときには、ただちにリセットされるのを防ぐために、常に T6 ビットに 1 を書き込んでください。

図 662. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGTB[2:0]} \times (T[5:0] + 1) \quad (ms)$$

ここで、

t_{WWDG}：WWDG タイムアウト

t_{PCLK}：APB クロック周期の測定値（ms）

4096：内部分周器に対応する値

たとえば、APB 周波数が 48 MHz である場合、WDGTB[2:0] は 3 に設定され、T[5:0] は 63 に設定されます。

$$t_{WWDG} = (1/48000) \times 4096 \times 2^{3} \times (63 + 1) = 43.69ms$$

t_{WWDG} の最小値と最大値については、データシートを参照してください。

43.3.5 デバッグ･モード

デバイスがデバッグモードになると（プロセッサは停止状態）、WWDG カウンタは、DBG モジュールの設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、セクション 47.16.2: タイマ、RTC、ウォッチドッグ、および I2C のデバッグサポートを参照してください。

43.4 WWDG 割込み

実際にリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割込み（EWI）が使用できます。EWI 割込みは、WWDG_CFR レジスタの EWI ビットをセットすることによって有効になります。ダウンカウンタ値が 0x40 に到達すると、EWI 割込みが生成され、対応する割込みサービス･ルーチン（ISR）を使用して、デバイスをリセットする前に特定の処理（通信やデータ･ロギングなど）をトリガできます。

アプリケーションによっては、WWDG リセットを生成せずに、EWI 割込みを使用してソフトウェアのシステム･チェックやシステム復旧／グレースフル･デグラデーション（graceful degradation）を管理できます。この場合、対応する割込みサービス･ルーチン（ISR）で WWDG カウンタを再ロードして WWDG リセットを回避したうえで、必要な操作をトリガする必要があります。

EWI 割込みは、WWDG_SR レジスタの EWIF ビットに“0”を書き込むことによってクリアされます。

注： **優先順位の高いタスクで発生するシステム･ロックなどが原因で EWI 割込みを ISR で処理できない場合、最終的には WWDG リセットが生成されます。**

43.5 WWDG レジスタ

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル･レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

43.5.1 WWDG 制御レジスタ（WWDG_CR）

アドレスオフセット：0x000

リセット値：0x0000 007F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
								rs	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **WDGA:** 有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。WDGA = 1 のとき、ウォッチドッグはリセットを生成できます。

0：ウォッチドッグは無効です。

1：ウォッチドッグは有効です。

ビット 6:0 **T[6:0] :** 7 ビットカウンタ（MSB から LSB まで）

これらのビットは、ウォッチドッグカウンタの値を含みます。

（4096 x $2^{WDGTB[2:0]}$）PCLK サイクルごとにデクリメントされます。0x40 に達して 0x3F にデクリメントされると（T6 がクリアされると）、リセットが生成されます。

43.5.2 WWDG 設定レジスタ（WWDG_CFR）

アドレスオフセット：0x004

リセット値：0x0000 007F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	WDGTB[2:0]			Res.	EWI	Res.	Res.	W[6:0]						
		rw	rw	rw		rs			rw	rw	rw	rw	rw	rw	rw

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13:11 **WDGTB[2:0] :** タイマベース

プリスケーラのタイムベースは、次のように変更できます。

000：CK カウンタ･クロック（PCLK/4096）1 分周

001：CK カウンタ･クロック（PCLK/4096）2 分周

010：CK カウンタ･クロック（PCLK/4096）4 分周

011：CK カウンタ･クロック（PCLK/4096）8 分周

100：CK カウンタ･クロック（PCLK/4096）16 分周

101：CK カウンタ･クロック（PCLK/4096）32 分周

110：CK カウンタ･クロック（PCLK/4096）64 分周

111：CK カウンタ･クロック（PCLK/4096）128 分周

ビット 10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **EWI**：早期ウェイクアップ割込み

このビットがセットされているときには、カウンタの値が 0x40 に達したときに割込みが発生します。この割込みは、リセット後にハードウェアによってのみクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **W[6:0]**：7 ビットウィンドウ値

これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

43.5.3 WWDG ステータスレジスタ（WWDG_SR）

アドレスオフセット：0x008

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
															rc_w0

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **EWIF**：早期ウェイクアップ割込みフラグ

このビットは、カウンタの値が 0x40 に達したときにハードウェアによってセットされます。0 を書き込んでソフトウェアでクリアする必要があります。1 を書き込んでも、ビットの値は変化しません。このビットは、割込みが有効でない場合にもセットされます。

43.5.4 WWDG レジスタマップ

次の表に、WWDG レジスタ･マップとリセット値を示します。

表 396. WWDG レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	**WWDG_CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGA	T[6:0]						
	リセット値																									0	1	1	1	1	1	1	1
0x004	**WWDG_CFR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDGTB [2:0]			Res.	EWI	Res.	Res.	W[6:0]						
	リセット値																			0	0	0		0			1	1	1	1	1	1	1
0x008	**WWDG_SR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EWIF
	リセット値																																0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

44 FD コントローラ･エリア･ネットワーク（FDCAN）

44.1 概要

コントローラ･エリア･ネットワーク（CAN）のサブシステム（図 663 を参照）は、1 つの CAN モジュール、共有メッセージ RAM、設定ブロックで構成されます。これらの各部分のベース･アドレスについてはメモリ･マップを参照してください。

モジュール（FDCAN）は、ISO 11898-1: 2015（CAN プロトコル仕様バージョン 2.0 パート A、B）および CAN FD プロトコル仕様バージョン 1.0 に準拠しています。

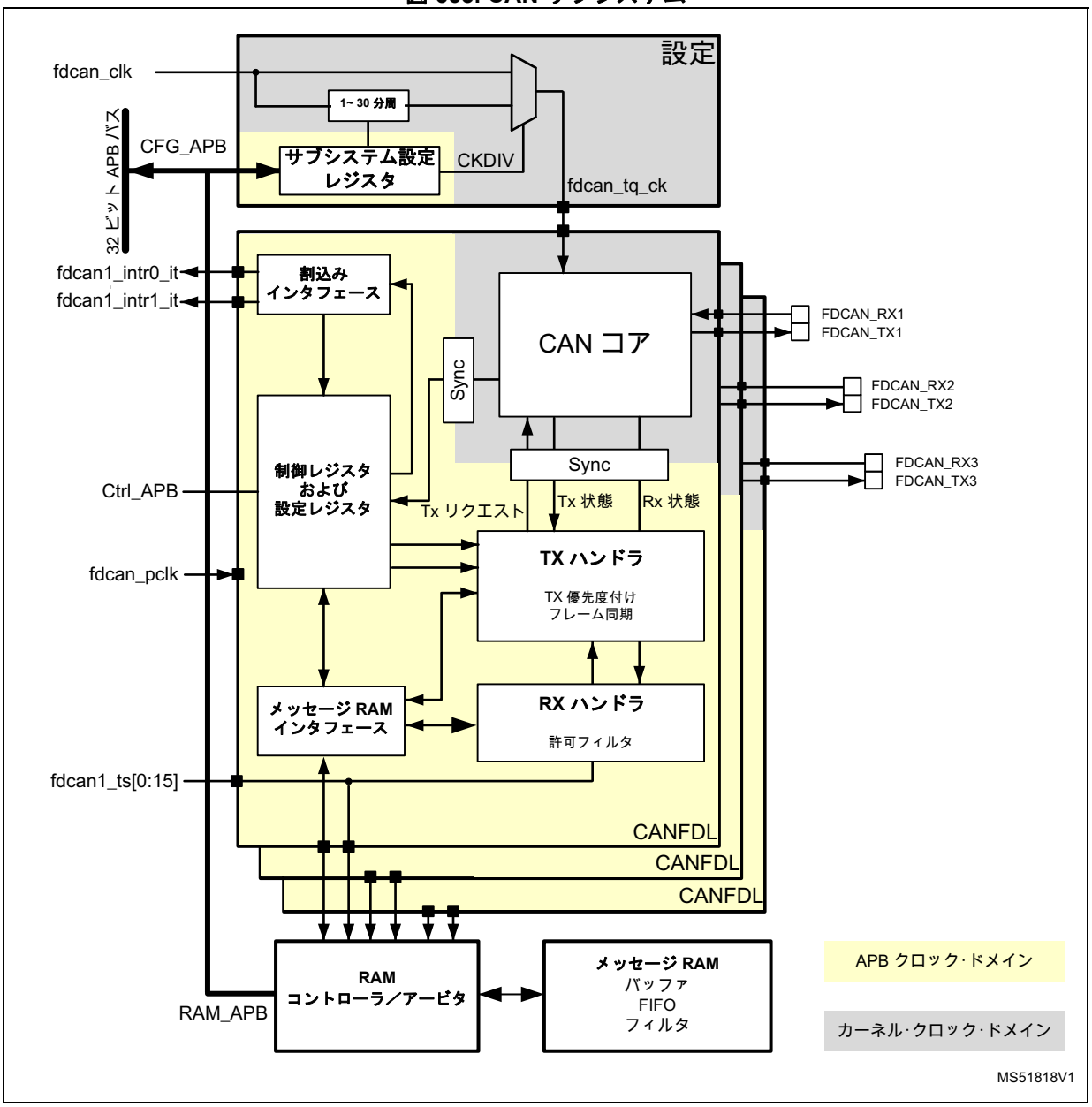
FDCAN インスタンスあたり 0.8 KB のメッセージ RAM には、フィルタ、受信 FIFO、送信イベント FIFO、および送信 FIFO が実装されます。

CAN サブシステムの I/O 信号とピンの詳細を、それぞれ表 397 および図 663 に示します。

表 397. CAN サブシステム I/O 信号

信号名	タイプ	説明
fdcan_ck	デジタル入力	CAN サブシステムカーネルクロック入力
fdcan_pclk		CAN サブシステム APB インタフェースクロック入力
fdcan_intr0_it	デジタル出力	FDCAN interrupt0
fdcan_intr1_it		FDCAN interrupt1
fdcan_ts[0:15]	-	外部タイムスタンプベクトル
FDCAN_RX	デジタル入力	FDCAN 受信ピン
FDCAN_TX	デジタル出力	FDCAN 送信ピン
APB インタフェース	デジタル入出力	設定、制御、および RAM アクセス用の複数の psel を持つシングル APP

図 663. CAN サブシステム

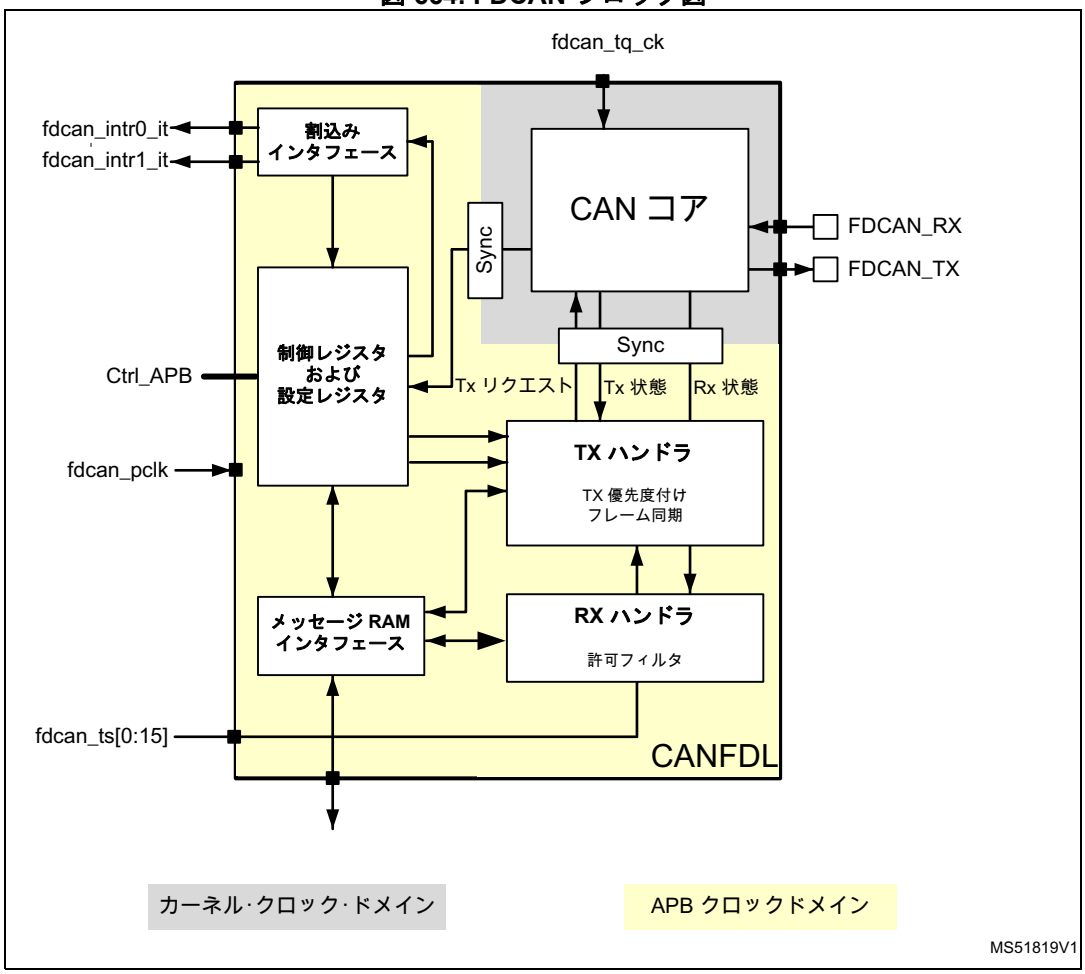


44.2 FDCAN の主な機能

- CAN プロトコルバージョン 2.0 パート A、Bおよび ISO 11898-1: 2015, -4
- 最大 64 データバイトをサポートする CAN FD
- CAN エラーロギング
- AUTOSAR および J1939 サポート
- 改良された許可フィルタ
- それぞれ 3 つのペイロードを持つ 2 つの受信 FIFO（ペイロード当たり最大 64 バイト）
- 高優先メッセージの受信で個別の信号を生成
- 3 つのペイロードの設定可能な送信 FIFO/キュー（ペイロード当たり最大 64 バイト）
- 送信イベント FIFO
- プログラム可能なループ･バック･テスト･モード
- マスク可能なモジュール割込み
- 2 つのクロックドメイン：APB バスインタフェースおよび CAN コアカーネルクロック
- パワーダウンサポート

44.3 FDCAN の機能詳細

図 664. FDCAN ブロック図



2 つの割込みライン

FDCAN ペリフェラルには、fdcan_intr0_it および fdcan_intr1_it の 2 つの割込みラインがあります。

FDCAN_ILE レジスタで EINT0 および EINT1 ビットをプログラムすることで、割込みラインを個別に有効化または無効化できます。

CAN コア

CAN コアには、プロトコル･コントローラおよび受信／送信シフト･レジスタがあります。コアはすべての ISO 11898-1: 2015 プロトコル機能を処理し、11 ビットおよび 29 ビットの識別子両方をサポートします。

Sync

同期ブロックは、APB クロックドメインと CAN カーネルクロックドメインとの信号を同期します。

Tx ハンドラ

メッセージ RAM から CAN コアへのメッセージ転送を制御します。送信に対して、最大 3 個の Tx バッファを使用できます。Tx バッファは、Tx FIFO または Tx キューとして使用できます。Tx イベント FIFO には、対応するメッセージ ID とともに Tx タイムスタンプを格納します。送信キャンセルもサポートしています。

Rx ハンドラ

CAN コアから受信したメッセージを外部メッセージ RAM に転送する処理を制御します。Rx ハンドラは、許可フィルタを通過したすべてのメッセージのストレージ用に、2 つの受信 FIFO をサポートします。Rx タイムスタンプが、各メッセージとともに格納されます。11 ビットの ID には最大 28 個のフィルタ、29 ビットの ID には最大 8 個のフィルタを定義できます。

APB インタフェース

設定レジスタ、コントローラの設定、および RAM アクセス用の APB バスに FDCAN を接続します。

メッセージ RAM インタフェース

FDCAN には RAM コントローラ／アービタ経由で外部の 1 KB メッセージ RAM が接続されています。

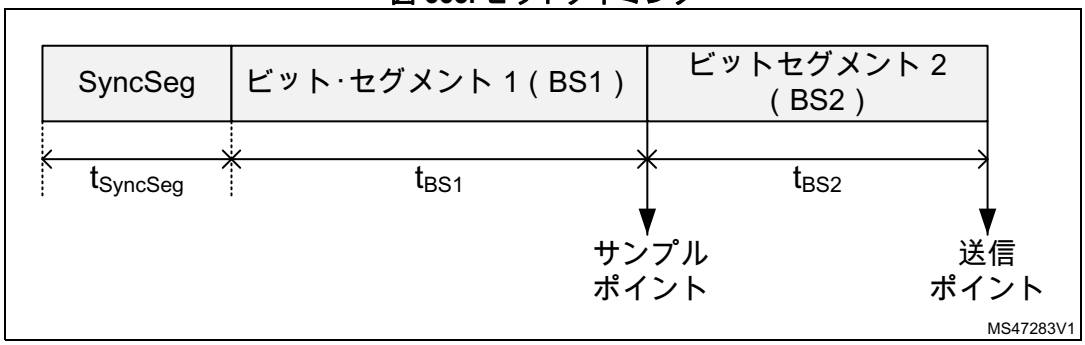
44.3.1 ビット･タイミング

ビットタイミング回路は、シリアルバスラインを監視し、スタートビットエッジでの同期と後続エッジでの再同期によって、サンプリングとサンプリングポイントの調整を行います。

図 665 に示すように、この動作は、ビット時間を次のように 3 つのセグメントに分割するとわかりやすく説明できます。

- 同期セグメント（SYNC_SEG）: この 1 タイムクォンタム（1 x tq）に固定された長さの時間セグメント内にビット変化が生じることが想定されます。
- ビットセグメント 1（BS1）: サンプルポイントの位置を定義します。これには、CAN 規格の PROP_SEG と PHASE_SEG1 が含まれます。セグメントの時間は 1～16 タイムクォンタムの間でプログラミング可能ですが、ネットワークのさまざまなノード間の周波数の差に起因する正の位相ドリフトを補正するために、自動的に延長されることがあります。
- ビットセグメント 2（BS2）: 送信ポイントの位置を定義します。これは、CAN 規格の PHASE_SEG2 に相当します。セグメントの時間は 1～8 タイムクォンタムの間でプログラミング可能ですが、負の位相ドリフトを補正するために、自動的に短縮されることもあります。

図 665. ビットタイミング



ボーレートはビット時間の逆数（ボーレート = 1／ビット時間）で、3 つのコンポーネントの和です。図 665 は、ビット時間 = $t_{SyncSeg}$ + t_{BS1} + t_{BS2} を示します。ここで、

- 公称ビット時間の場合
  - tq = (FDCAN_NBTP.NBRP[8:0] + 1) * $t_{fdcan_tq_clk}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_NBTP.NTSEG1[7:0] + 1)
  - t_{BS2} = tq * (FDCAN_NBTP.NTSEG2[6:0] + 1)
- データビット時間の場合
  - tq = (FDCAN_DBTP.DBRP[4:0] + 1) * $t_{fdcan_tq_clk}$
  - $t_{SyncSeg}$ = 1 tq
  - t_{BS1} = tq * (FDCAN_DBTP.DTSEG1[4:0] + 1)
  - t_{BS2} = tq * (FDCAN_DBTP.DTSEG2[3:0] + 1)

（再）同期ジャンプ幅（SJW）は、ビット･セグメントの延長量または短縮量の上限を規定し、1 ~ 4 タイム･クォンタムの範囲でプログラミング可能です。

有効なエッジは、コントローラ自体はリセッシブビットを送信しないという条件で、あるビット時間において、ドミナントからリセッシブへのバスレベルの最初の遷移と定義されます。

有効なエッジが SYNC_SEG ではなく BS1 で検出された場合、サンプルポイントが遅くなるように、BS1 は最大で SJW 延長されます。

逆に、有効なエッジが SYNC_SEG ではなく BS2 で検出された場合、送信ポイントが早くなるように、BS2 は最大で SJW 短縮されます。

プログラミング･エラーに対する安全対策としてのビット･タイミング･レジスタは、デバイスが STANDBY モードの場合にのみ設定可能です。レジスタ FDCAN_DBTP および FDCAN_NBTP（それぞれデータ･ビット時間および公称ビット時間専用）は、CCCR.CCE と CCCR.INIT がセットされている場合にのみアクセス可能です。

注 : **CAN ビット･タイミングと再同期メカニズムの詳細については、ISO 11898-1 規格を参照してください。**

44.3.2 動作モード

設定

IP バージョン、ハードウェア、および入力クロック分周回路を設定します。クロック分周回路を 0 にセットすると、主な入力クロックはそのまま使用されます。

ソフトウェアの初期化

ソフトウェアの初期化は、FDCAN_CCCR レジスタで INIT ビットをセットするか、ソフトウェアやハードウェアリセット、Bus_Off への移行で開始されます。FDCAN_CCCR レジスタで INIT ビットをセットしている間、CAN バスとのメッセージ転送は停止し、CAN バス出力 FDCAN_TX のステータスはリセッシブ（ハイ）になります。EML（エラー管理ロジック）のカウンタは変更されません。FDCAN_CCCR の INIT ビットのセットによって変更される設定レジスタはありません。FDCAN_CCCR の INIT ビットのクリアで、ソフトウェア初期化は終了します。その後、ビットストリームプロセッサ（BSP）は 11個の連続したリセッシブビットの発生を待つことで（Bus_Idle）、CAN バスでのデータ転送と同期してから、バスアクティビティに移行してメッセージ転送を開始できます。

FDCAN 設定レジスタへのアクセスは、FDCAN_CCCR レジスタの INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみ有効です。

FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがセットされている間のみセット／クリアできます。FDCAN_CCCR レジスタの CCE ビットは、FDCAN_CCCR の INIT ビットがクリアされているとき、自動的にクリアされます。

FDCAN_CCCR レジスタの CCE ビットがセットされている場合、次のレジスタはリセットされます。

- FDCAN_HPMS - 高優先メッセージ･ステータス
- FDCAN_RXF0S - Rx FIFO 0 ステータス
- FDCAN_RXF1S - Rx FIFO 1 ステータス
- FDCAN_TXFQS - Tx FIFO／キュー･ステータス
- FDCAN_TXBRP - Tx バッファ･リクエスト･ペンディング
- FDCAN_TXBTO - Tx バッファ送信発生
- FDCAN_TXBCF - Tx バッファキャンセル終了
- FDCAN_TXEFS - Tx イベント FIFO ステータス

FDCAN_TOCV レジスタのタイムアウト･カウンタ値 TOC ビットは、FDCAN_CCCR の CCE ビットがセットされているとき、FDCAN_TOCC レジスタの TOP ビットによって設定された値にプリセットされます。

さらに、Tx ハンドラおよび Rx ハンドラのステート･マシンは、FDCAN_CCCR の CCE ビットがセットされている間はアイドル状態に保持されます。

FDCAN_CCCR レジスタの CCE ビットがクリアされている場合のみ、次のレジスタは書込み可能です。

- TXBAR - Tx バッファ追加リクエスト
- TXBCR - Tx バッファ･キャンセル･リクエスト

FDCAN_CCCR の TEST ビットおよび FDCAN_CCCR の MON ビットは、CCCR の INIT ビットと CCCR レジスタの CCE ビットの両方がセットされている場合にのみ、ソフトウェアによってセットできます。両ビットは常時リセットできます。FDCAN_CCCR の DAR ビットは、FDCAN_CCCR の INIT ビットと FDCAN_CCCR レジスタの CCE ビットの両方がセットされているときのみセット／クリアできます。

通常動作

ハードウェアリセット後の FDCAN のデフォルト動作モードは、イベント駆動 CAN 通信です。TT 動作モードはサポートされていません。

FDCAN が初期化され、FDCAN_CCCR レジスタの INIT ビットがクリアされると、FDCAN は CAN バスと同期し、通信できるようになります。

許可フィルタを通過した後、メッセージ ID と DLC を含む受信メッセージは、Rx FIFO 0 や Rx FIFO 1 に格納されます。

送信されるメッセージの場合、Tx FIFO や Tx キューを初期化または更新できます。リモートフレーム受信時の自動送信はサポートされません。

CAN FD 動作

FDCAN プロトコルは 2 種類あります。

1. CAN フレームのデータ･フィールドを 8 バイトより長くすることができるロング･フレーム･モード（LFM）
2. CAN フレームの制御フィールド、データ･フィールド、および CRC フィールドがフレームの開始時と終了時より高いビット･レートで送信される高速フレーム･モード（FFM）

高速フレーム･モードは、ロング･フレーム･モードと組み合わせて使用できます。

11 ビットの識別子の CAN フレームで以前に予約済みのビットと、29 ビットの識別子の CAN フレームで一番最初に予約済みのビットは、FDF ビットとしてデコードされます。FDF リセッシブは、CAN FD フレームを示し、FDF ドミナントはクラシック CAN フレームを示します。

CAN FD フレームでは、FDF に続く 2 つのビットの res および BRS によって、この CAN FD フレーム内のビットレートを切り替えるかどうかが決まります。CAN FD ビットレートの切り替えは、res ドミナントと BRS リセッシブによって示されます。res リセッシブのコーディングは、プロトコルを今後拡張するために予約済みとなります。FDF リセッシブおよび res リセッシブでフレームを受信した FDCAN は、ビット PSR.PXE をセットすることでプロトコル例外イベントを通知します。プロトコル例外処理が有効である場合（CCCR.PXHD = 0）、動作状態が次のサンプル･ポイントで Receiver（PSR.ACT = 10）から Integrating（PSR.ACT = 00）に変わります。プロトコル例外処理が無効である場合（CCCR.PXHD = 1）、FDCAN はリセッシブの res ビットをフォーム･エラーとして扱い、エラー･フレームで応答します。

CAN FD 動作は、CCCR.FDOE をプログラムすることで有効になります。CCCR.FDOE = 1 の場合、CAN FD フレームの送受信は有効です。クラシック CAN フレームの送受信はいつでも可能です。CAN FD フレームまたはクラシック CAN フレームのどちらを送信するかは、それぞれの Tx バッファ要素のビット FDF で設定できます。CCCR.FDOE = 0 では、受信フレームはクラシック CAN フレームとして解釈され、CAN FD フレームを受信するとエラー･フレームが送信されます。CAN FD 動作が無効である場合、Tx バッファ要素のビット FDF がセットされている場合でも CAN FD フレームは送信されません。CCCR.FDOE と CCCR.BRSE は、CCCR.INIT と CCCR.CCE の両方がセットされている状態でのみ変更できます。

CCCR.FDOE = 0 では、ビット FDF および BRS の設定は無視され、フレームはクラシック CAN フォーマットで送信されます。CCCR.FDOE = 1 で CCCR.BRSE = 0 である場合は、Tx バッファ要素のビット FDF のみが評価されます。CCCR.FDOE = 1 で CCCR.BRSE = 1 である場合は、ビット･レートを切り替えて CAN FD フレームを送信できます。ビット FDF と BRS がセットされたすべての Tx バッファ要素は、ビット･レートを切り替えて CAN FD フォーマットで送信されます。

CAN 動作中のモード変更は、次の条件下でのみ推奨されます。

- CAN FD データフェーズでの失敗率は、CAN FD アービトレーションフェーズより大幅に高くなります。この場合、送信用の CAN FD ビットレート切り替えオプションを無効にします。
- システム起動中、すべてのノードが CAN FD フォーマットで通信できることを検証するまでクラシック CAN メッセージを送信します。これが当てはまる場合、すべてのノードは CAN FD 動作に切り替わります。
- CAN パーシャル･ネットワーキングのウェイクアップ･メッセージは、クラシック CAN フォーマットで送信する必要があります。
- CAN FD に対応していないノードがある場合の EOL のプログラミング。非 CAN FD ノードは、プログラミングが完了するまでサイレントモードで保持されます。その後、すべてのノードがクラシック CAN 通信に切り替わります。

FDCAN フォーマットでは、DLC のコーディングは標準の CAN フォーマットの 1 つとは異なります。DLC コード 0～8 は標準 CAN と同じコーディングで、コード 9～15（標準 CAN ではすべて 8 バイトのデータフィールドのコーディング）は表 398 に従ってコーディングされます。

表 398. FDCAN での DLC コーディング

DLC	9	10	11	12	13	14	15
データバイト数	12	16	20	24	32	48	64

CAN FD 高速フレームでは、BRS（ビット･レート切り替え）ビットがリセッシブである場合、このビットの後のフレーム内でビット･タイミングが切り替わります。FDCAN アービトレーション･フェーズで BRS ビットまでは、ビット･タイミングおよびプリスケーラ･レジスタ BTP によって定義

されたとおりに、標準の CAN ビット･タイミングが使用されます。次の FDCAN データ･フェーズでは、高速ビット･タイミングおよびプリスケーラ･レジスタ FBTP によって定義されたとおりに、高速 CAN ビット･タイミングが使用されます。CRC デリミタまたはエラーのどちらかが検出された時点で、ビット･タイミングが高速タイミングから元に戻ります。

CAN FD データフェーズで設定可能な最大ビットレートは、FDCAN カーネルクロック周波数に依存します。たとえば、20 MHz の FDCAN カーネルクロック周波数で 4 タイムクォンタム（tq）の設定可能な最小ビット時間とした場合、データフェーズのビットレートは 5 Mbit/s となります。

CAN FD ロング･フレームと CAN FD 高速フレームのどちらのフレーム・フォーマットでも、ビット ESI（エラー･ステータス･インジケータ）の値は送信開始時のトランスミッタのエラー状態によって決まります。トランスミッタがエラー･パッシブである場合、ESI はリセッシブで送信され、その他の場合はドミナントで送信されます。CAN FD リモートフレームでは、ESI ビットはトランスミッタのエラー状態にかかわらず、常にドミナントで送信されます。CAN FD リモートフレームのデータ長コードは 0 で送信されます。

FDCAN Tx バッファが DLC > 8 で FDCAN を送信するように設定されている場合、最初の 8 バイトは Tx バッファで設定されたとおりに送信され、データ･フィールドの残りの部分は 0xCC でパディングされます。FDCAN が DLC > 8 で FDCAN フレームを受信する場合、フレームの最初の 8 バイトがそれに対応する Rx FIFO に格納されます。残りのバイトは破棄されます。

トランシーバ遅延補正

FDCAN 送信のデータフェーズの間、1 つのノードだけが送信して、その他すべてはレシーバになります。バスラインの長さに影響はありません。ピン FDCAN_TX 経由で送信する場合、プロトコルコントローラはピン FDCAN_RX 経由でローカル CAN トランシーバから送信されたデータを受信します。受信されるデータは、CAN トランシーバループ遅延により遅れます。この遅延が TSEG1（サンプル･ポイント前の時間セグメント）より長くなると、ビット･エラーが検出されます。トランシーバ遅延補正がない場合、FDCAN フレームのデータ･フェーズのビット･レートはトランシーバ･ループ遅延によって制限されます。

FDCAN は遅延補正メカニズムを実装して、CAN トランシーバループ遅延を補正することで、特定の CAN トランシーバの遅延にかかわらず、FDCAN データフェーズ中により高いビットレートでの送信を可能にします。

ノード送信のデータ･フェーズ中にビット･エラーをチェックするために、遅延した送信データはセカンダリ･サンプル･ポイント（SSP）で受信したデータと比較されます。ビット･エラーが検出されると、トランスミッタは次のレギュラ･サンプル･ポイントでこのビット･エラーに反応します。アービトレーション･フェーズ中、遅延補正は常に無効です。

トランスミッタ遅延補正は、データビット時間がトランスミッタ遅延より短い場合に設定が可能になります。詳細は、新しい ISO11898-1 に記載されています。ビット DBTP.TDC をセットすることで有効になります。

受信したビットは、SSP で送信したビットと比較されます。SSP の位置は、トランシーバ経由で FDCAN の送信出力ピン FDCAN_TX から受信入力ピン FDCAN_RX までに測定された遅延と、TDCR.TDCO によって設定されたトランスミッタ遅延補正オフセットの合計として定義されます。トランスミッタ遅延補正オフセットは、受信したビット内の SSP の位置を調整するために使用します（例：データフェーズのビット時間の半分）。セカンダリサンプルポイントの位置は、次の mtq（fdcan_tq_ck クロックの 1 周期である最小タイムクォンタム）の整数に切り捨てられます。

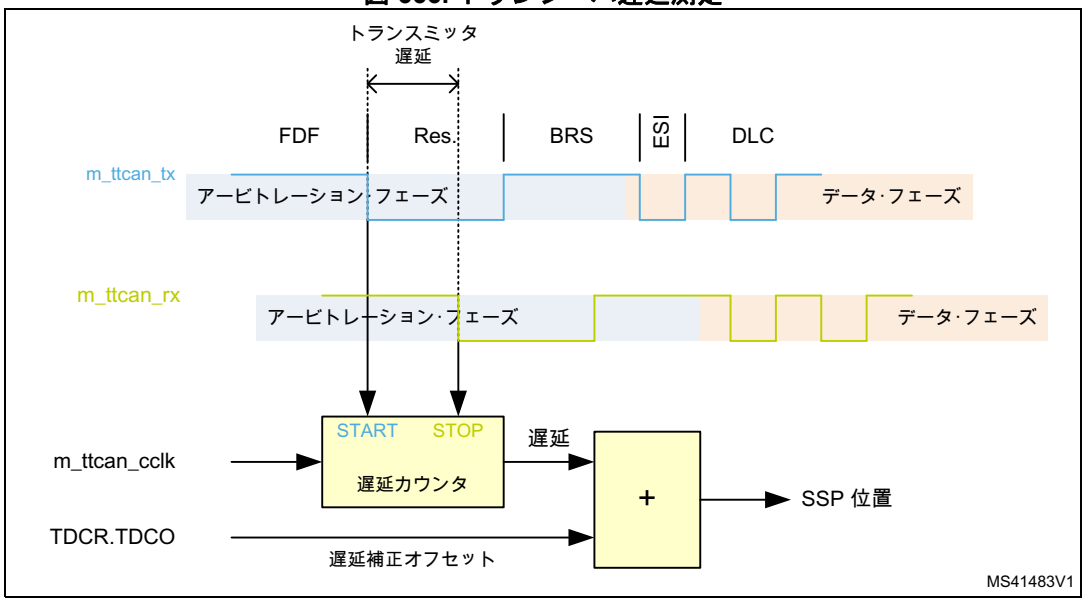
PSR.TDCV は、実際のトランスミッタ遅延補正値を示します。PSR.TDCV は、CCCR.INIT がセットされているとクリアされ、DBTP.TDC がセットされている間 FD フレームの各送信で更新されます。

FDCAN に実装されたトランスミッタ遅延補正では、次の境界条件を考慮する必要があります。

- FDCAN_Tx から FDCAN_Rx までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット TDCR.TDCO の合計は、データ･フェーズで 6 ビット時間未満でなければなりません。
- FDCAN_TX から FDCAN_RX までの測定された遅延と、設定されたトランスミッタ遅延補正オフセット TDCR.TDCO の合計は、127 mtq 以下でなければなりません。合計がこの値を超える場合、トランスミッタ遅延補正には最大値（127 mtq）が使用されます。
- データ･フェーズは、CRC デリミタのサンプル･ポイントで終了し、SSP で受信したビットのチェックを停止します。

DBTP.TDC = 1 をプログラムすることでトランスミッタ遅延補正を有効にしている場合、送信した各 CAN FD フレームの中で、ビット FDF がビット･リセットまで立ち下がるエッジで測定が開始されます。このエッジがトランスミッタの受信入力ピン FDCAN_TX で見られると測定は停止します。この測定の解析時間は、1 mtq です。

図 666. トランシーバ遅延測定



受信した FDF ビット内のドミナントグリッチによって、受信した res ビットの立ち下がりエッジ前に遅延補正測定を終了する（早い SSP の位置となる）ことを避けるために、TDCR.TDCF をプログラムすることでトランスミッタ遅延補正フィルタウィンドウの使用が有効になります。これで SSP の位置の最小値を定義します。より早い SSP の位置につながる FDCAN_RX のドミナントエッジは、トランスミッタ遅延測定では無視されます。SSP の位置が少なくとも TDCR.TDCF で、FDCAN_RX がローになると、測定は停止します。

制限動作モード

制限動作モードでは、ノードはデータとリモート･フレームを受信して有効なフレームに確認応答を返すことができますが、データ･フレーム、リモート･フレーム、アクティブ･エラー･フレーム、オーバーロード･フレームのいずれも送信しません。エラー状態やオーバーロード状態になった場合、ドミナント･ビットを送らずに、バス･アイドル状態の発生を待って CAN 通信と再同期します。エラー･カウンタ（ECR.REC、ECR.TEC）は、エラー･ロギング（ECR.CEL）がアクティブになっている間は停止します。ソフトウェアは、ビット CCCR.ASM をセットすることで FDCAN を制限動作モードに設定できます。このビットは、CCCR.CCE および CCCR.INIT の両方が 1 に設定されているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってクリアできます。

Tx ハンドラがメッセージ RAM から時間内にデータを読み出せなかったとき、自動的に制限動作モードに移行します。制限動作モードを終了するには、ソフトウェアで CCCR.ASM をリセットする必要があります。

制限動作モードは、さまざまな CAN ビット･レートに適応するアプリケーションで使用できます。この場合、アプリケーションはさまざまなビット･レートをテストし、有効なフレームを受信すると制限動作モードを終了します。

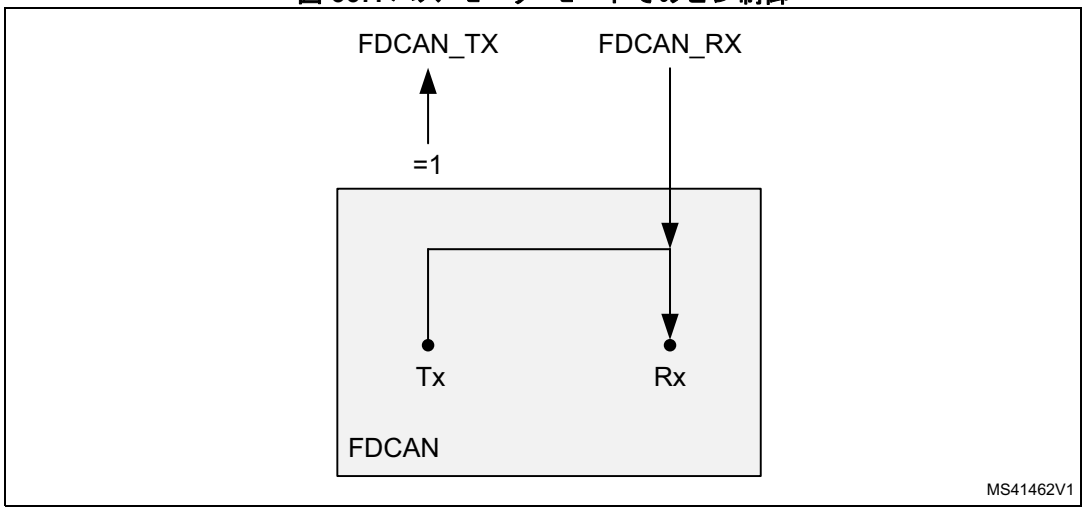
注： **制限動作モードを、ループ･バック･モード（内部または外部）と組み合わせないでください。**

バス･モニタ･モード

FDCAN は、CCCR.MON ビットをセットすることでバス･モニタ･モードに設定されます。バス･モニタ･モード（詳細は ISO11898-1、10.12 Bus monitoring を参照してください）では、FDCAN は有効なデータ･フレームとリモート･フレームを受信できますが、送信は開始できません。このモードでは、CAN バスでリセッシブ･ビットのみを送信します。FDCAN がドミナント･ビット（ACK ビット、オーバーロード･フラグ、アクティブ･エラー･フラグ）を送信する必要がある場合、CAN バスがリセッシブ状態にとどまることがあっても FDCAN がこのドミナント･ビットを監視できるように、このビットは内部でルート変更されます。バス･モニタ･モードでは、TXBRP レジスタがリセット状態のまま保持されます。

バス･モニタ･モードを使用すると、ドミナント･ビットの送信による影響を受けずに、CAN バス上のトラフィックを解析できます。図 667 は、バス･モニタ･モードでの FDCAN への FDCAN_TX および FDCAN_RX 信号の接続を示しています。

図 667. バス･モニタ･モードでのピン制御



自動再送信無効（DAR）モード

CAN 仕様（ISO11898-1、6.3.3 recovery management）に従い、FDCAN はアービトレーションを喪失したフレームや送信中にエラーで中断されたフレームを自動的に再送信する方法を提供します。デフォルトでは自動再送信は有効化されています。

DAR モードでのフレーム送信

DAR モードでは、すべての送信が CAN バスで開始した後で自動的にキャンセルされます。Tx バッファ Tx リクエスト･ペンディング･ビット TXBRP.TRPx は、転送が成功した後、転送がまだ始まっていない時点でのキャンセル時、アービトレーション喪失によるアボート時、もしくはフレーム転送中にエラーが発生した時にリセットされます。

- 送信が成功した場合
  - 対応する Tx バッファ送信発生ビット TXBTO[TOx] がセットされます
  - 対応する Tx バッファ･キャンセル終了ビット TXBCF[CFx] はセットされません
- キャンセルしたものの送信は成功した場合
  - 対応する Tx バッファ送信発生ビット TXBTO[TOx] がセットされます
  - 対応する Tx バッファ･キャンセル終了ビット TXBCF[CFx] がセットされます
- アービトレーション喪失またはフレーム送信中断の場合
  - 対応する Tx バッファ送信発生ビット TXBTO[TOx] はセットされません
  - 対応する Tx バッファ･キャンセル終了ビット TXBCF[CFx] がセットされます

フレーム送信が成功し、Tx イベントのストレージが有効になっている場合、Tx イベント FIFO 要素はイベント･タイプ ET = 10（キャンセルしたが送信）で書き込まれます。

パワー･ダウン（SLEEP モード）

FDCAN は、CC 制御レジスタ CCCR[CSR] 経由のクロック停止リクエスト入力によって制御されるパワーダウンモードにセットできます。クロック停止リクエストがアクティブである限り、ビット CCCR[CSR] は 1 で読み出されます。

すべてのペンディングされた送信リクエストが完了すると、FDCAN はバスアイドル状態が検出されるまで待ちます。その後、FDCAN は CCCR[INIT] を 1 にセットして、追加の CAN 転送を防ぎます。FDCAN は、CCCR[CSA] を 1 にセットすることでパワーダウンの準備が整ったことを確認します。この状態では、クロックがオフに切り替わる前に、追加のレジスタアクセスが可能です。CCCR[INIT] への書込みアクセスは効果がありません。これで、モジュール･クロック入力をオフに切り替えることができます。

パワーダウンモードを終了するには、アプリケーションが CC 制御レジスタフラグ CCCR.CSR をリセットする前にモジュールクロックをオンにしなければなりません。FDCAN は、CCCR[CSA] をリセットすることでこれを確認します。その後、アプリケーションは、ビット CCCR[INIT] をリセットすることで CAN 通信を再開できます。

テストモード

FDCAN テスト･レジスタ（FDCAN_TEST）への書込みアクセスを有効にするには、ビット CCCR.TEST を 1 にセットして、テストモードおよび機能の設定を有効にしなければなりません。

TEST.TX をプログラムすることで、CAN 送信ピン FDCAN_TX で 4 つの出力機能が使用できます。デフォルト機能（シリアルデータ出力）に加えて、CAN サンプルポイント信号を駆動して FDCAN ビットタイミングを監視し、一定のドミナントまたはリセッシブ値を駆動できます。ピン FDCAN_RX での実際の値は、TEST.RX から読み出せます。両方の機能を使用して、CAN バス物理層をチェックできます。

CAN カーネルクロックと APB クロックドメインとの同期メカニズムによって、新しい設定が FDCAN_TX 出力ピンで確認できるようになるまで、TEST.TX への書込み間で APB クロック周期数周期分の遅延が発生する場合があります。これは、TEST.RX 経由で FDCAN_RX 入力ピンを読み出すときにも適用されます。

注： **テスト･モードは製造検査または自己診断機能でのみ使用してください。FDCAN_TX ピンのソフトウェア制御は、すべての CAN プロトコル機能に干渉します。アプリケーションにテストモードを使用することは推奨されません。**

外部ループバックモード

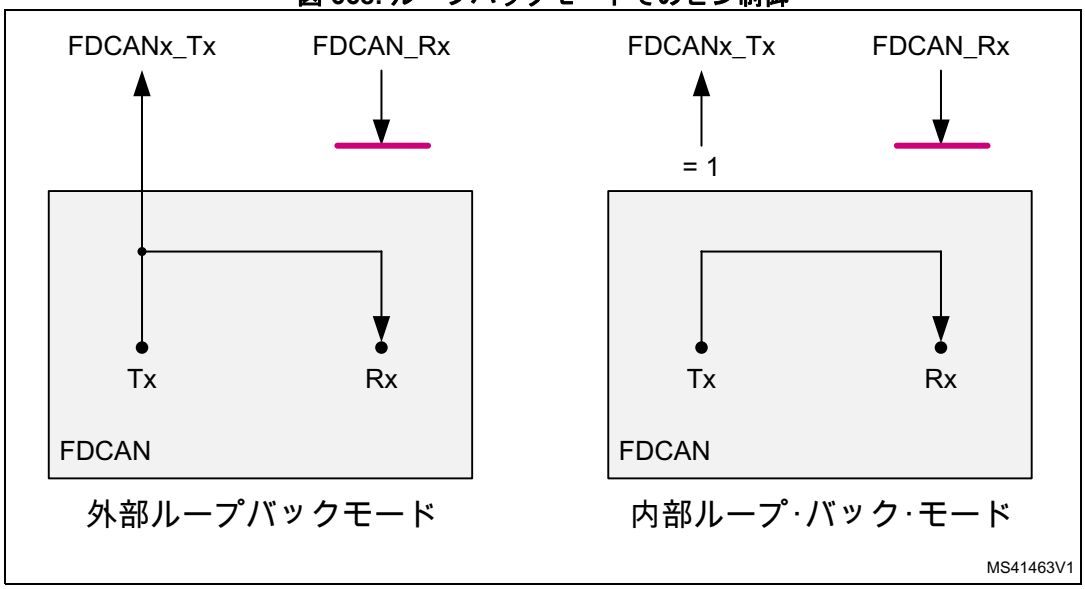
FDCAN を外部ループバックモードに設定するには、TEST.LBCK を 1 にセットします。ループバックモードでは、FDCAN はそれ自体が送信したメッセージを受信メッセージとして扱い、それらを Rx FIFO に格納します（許可フィルタを通過した場合）。図 668 では、外部ループバックモードでの FDCAN への送受信信号 FDCAN_TX および FDCAN_RX の接続を示します。

このモードは、ハードウェアの自己診断機能のために用意されています。外部通信から独立させるために、ループバックモードでは、FDCAN は ACK エラーを無視します（データ／リモートフレームの ACK スロットではリセッシブビットがサンプリングされます）。このモードでは、FDCAN は送信出力から受信入力へ内部フィードバックを行います。FDCAN は、FDCAN_RX 入力ピンの実際の値を無視します。送信メッセージは FDCAN_TX 送信ピンで監視できます。

内部ループバックモード

TEST.LBCK と CCCR.MON を 1 にプログラムすることによって、内部ループバックモードに移行します。このモードは「ホットセルフテスト」に使用できます。つまり、FDCAN_TX および FDCAN_RX ピンに接続されている実行中の CAN システムに影響を与えずに、FDCAN をテストできます。このモードでは、FDCAN_RX ピンは FDCAN から切断され、FDCAN_TX ピンはリセッシブに保持されます。図 668 では、内部ループバックモードの場合の FDCAN への FDCAN_TX および FDCAN_RX ピンの接続を示します。

図 668. ループバックモードでのピン制御



タイムスタンプの生成

タイムスタンプの生成について、FDCAN は 16 ビットラップアラウンドカウンタを提供します。プリスケーラ TSCC.TCP を設定して、CAN ビット時間の倍数（1～16）でカウンタにクロック供給できます。16)。カウンタは、TSCV[TCV] で読み出せます。レジスタ TSCV への書込みアクセスによって、カウンタは 0 にリセットされます。タイムスタンプカウンタが0に戻ると、割込みフラグ IR[TSW] がセットされます。

フレーム受信／送信の開始時、カウンタ値がキャプチャされ、Rx FIFO（RXTS[15:0]）要素または Tx イベント FIFO（TXTS[15:0]）要素のタイムスタンプ･セクションに格納されます。

ビット TSCC.TSS をプログラムすることで、16 ビットのタイムスタンプを使用できます。

デバッグモードの動作

デバッグモードでは、読出し機能のセット／リセットはデバッガレジスタアクセス中に自動的に無効になり、通常の MCU 動作中に有効になります。

タイムアウトカウンタ

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO のタイムアウト条件の信号を送信するために、FDCAN は 16-bit タイムアウトカウンタを提供します。ダウンカウンタとして動作し、タイムスタンプ カウンタの場合と同様に、TSCC[TCP] で制御されるプリスケーラを使用します。タイムアウト･カウンタは、レジスタ TOCC 経由で設定されます。実際のカウンタ値は、TOCV[TOC] から読み出すことができます。タイムアウト･カウンタは CCCR[INIT] = 0 である期間でのみ開始できます。FDCAN が Bus_Off 状態に移行したときなど、CCCR[INIT] = 1 の場合は停止します。

動作モードは TOCC[TOS] で選択されます。連続モードで動作時、CCCR[INIT] がリセットされるとカウンタが開始します。TOCV への書込みによって、カウンタは TOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。

タイムアウト･カウンタがいずれか 1つの FIFO で制御されている場合、空の FIFO は TOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。TOCV への書込みは無効です。

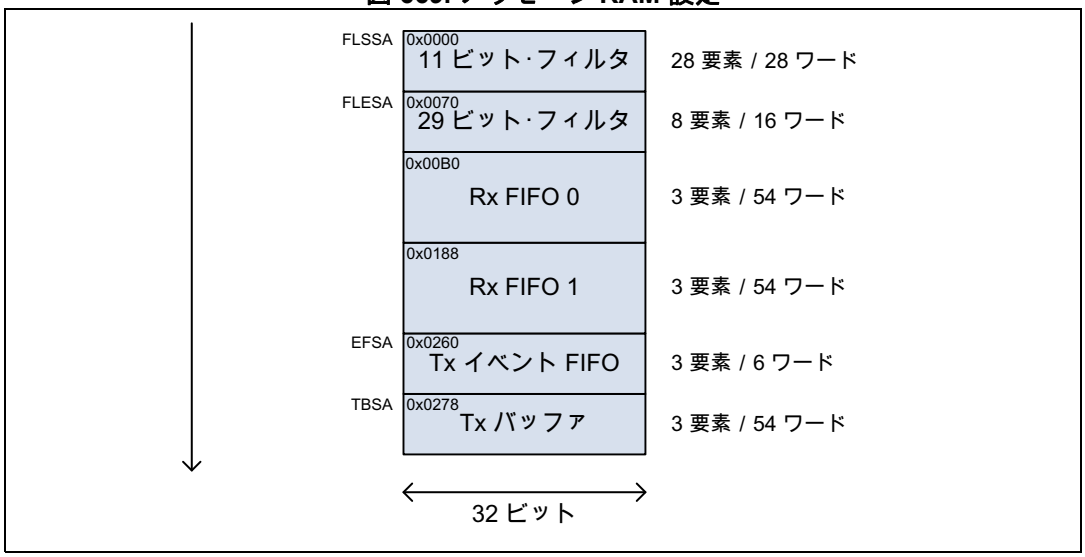
カウンタが 0 に達すると、割込みフラグ IR[TOO] がセットされます。連続モードでは、カウンタは TOCC[TOP] ですぐに再開されます。

注 : **タイムアウト･カウンタのクロック信号は、CAN コア･サンプル･ポイント信号から生成されます。そのため、タイムアウト･カウンタがデクリメントされるタイミングは、CAN コアの同期／再同期メカニズムによって異なる場合があります。FDCAN のボー･レート切り替え機能を使用している場合、タイムアウト･カウンタを駆動するクロックがアービトレーションとデータ･フィールドで異なります。**

44.3.3 メッセージ RAM

メッセージ RAM は 32 ビット幅で、これに最大 212 ワードを割り当てるために FDCAN モジュールが設定されます。図 669 に示された各セクションを設定する必要はありません。

図 669. メッセージ RAM 設定



FDCAN がメッセージ RAM のアドレスを指定する場合、シングルバイトではなく 32 ビットワード（整列）を指定します。RAM アドレスは 32 ビット･ワードです。つまり、ビット 15 ~ 2 だけが評価され、2 つの最下位ビットは無視されます。

複数のインスタンスがある場合、FDCANn の開始アドレスは（FDCANn-1 の終了アドレス + 4）で計算され、FDCANn の終了アドレスは（FDCANn の開始アドレス + 0x0350 - 4）で計算されます。

たとえば、インスタンスが 2 つの場合は、次のとおりです。

- FDCAN1：
  - 開始アドレス 0x0000
  - 終了アドレス 0x0350（図 669 参照）
- FDCAN2：
  - 開始アドレス = 0x0350 （FDCAN1 の終了アドレス）+ 4 = 0x0354
  - 終了アドレス = 0x0354 （FDCAN2 の開始アドレス）+ 0x0350 - 4 = 0x06A0

Rx の処理

Rx ハンドラは、許可フィルタ、2 つの Rx FIFO のうちの 1 つへの受信メッセージの転送、および Rx FIFO の Put と Get のインデックスを制御します。

許可フィルタ

FDCAN は、2 セットの許可フィルタを設定でき、1 つは標準識別子用で、もう 1 つは拡張識別子用です。これらのフィルタは、Rx FIFO 0 または Rx FIFO 1 に割り当てられます。許可フィルタでは、フィルタの各リストが要素 #0 から最初に一致する要素まで実行されます。許可フィルタは最初に一致する要素で停止します。次のフィルタ要素はこのメッセージでは評価されません。

主な機能：

- 各フィルタ要素は次のように設定できます。
  - 範囲フィルタ（〜から〜まで）
  - 1 つまたは 2 つの専用 ID のフィルタ
  - クラシックビットマスクフィルタ
- 各フィルタ要素は、許可フィルタまたは拒否フィルタで設定できます。
- 各フィルタ要素は個別に有効化／無効化できます。
- フィルタは連続でチェックされ、最初の一致フィルタ要素で実行を停止します。

関連する設定レジスタ：

- グローバルフィルタ設定（RXGFC）
- 拡張 ID ANDマスク（XIDAM）

フィルタ要素の設定（SFEC/EFEC）に応じて、一致すると次の操作のいずれかがトリガされます。

- FIFO 0 または FIFO 1 に受信フレームを格納
- 受信フレームを拒否
- 高優先メッセージ割込みフラグ IR[HPM] のセット
- 高優先メッセージ割込みフラグ IR[HPM] をセットし、FIFO 0 または FIFO 1 に受信フレームを格納

許可フィルタは、完了識別子を受信した後で開始されます。許可フィルタが完了した後、一致する Rx FIFO が見つかった場合、メッセージハンドラが 32 ビットの単位で受信したメッセージデータを一致する Rx FIFO に書込み始めます。CAN プロトコルコントローラがエラー条件（CRC エラーなど）を検出した場合、このメッセージは次の影響を与えつつ破棄されます。

- **Rx FIFO**
  一致する Rx FIFO の プット インデックスは更新されませんが、関連する FIFO 要素（一部）が受信したデータで上書きされます。エラー･タイプについては、PSR.LEC および PSR.DLEC を参照してください。一致する Rx FIFO が上書きモードで動作している場合、Rx FIFO 上書きモードに記載された境界条件を考慮する必要があります。

注： **許可したメッセージが 2 つの Rx FIFO のうちの 1 つに書き込まれると、変更されていない受信 ID は、使用されるフィルタとは別に格納されます。許可フィルタプロセスの結果は、設定されたフィルタ要素のシーケンスに強く依存します。**

範囲フィルタ

フィルタが SF1ID/SF2ID および EF1ID/EF2ID で定義された範囲のメッセージ ID を持つすべての受信フレームに一致します。

範囲フィルタを拡張フレームと使用する場合、2 つのことが可能です。

- EFT = 00：範囲フィルタが適用される前に、受信フレームのメッセージ ID は拡張 ID AND マスク（XIDAM）と AND でつながれます。
- EFT = 11：拡張 ID AND マスク（XIDAM）は、範囲フィルタに使用されません。

専用 ID のフィルタ

フィルタ要素を設定して、1 つまたは 2 つの固有 ID のフィルタでフィルタできます。1 つの固有メッセージ ID でフィルタするには、フィルタを SF1ID = SF2ID および EF1ID = EF2ID で設定する必要があります。

クラシックビットマスクフィルタ

クラシックビットマスクフィルタは、受信したメッセージ ID のシングルビットをマスクして、メッセージ ID のグループをフィルタすることを目的としています。クラシックビットマスクフィルタで、SF1ID/EF1ID はメッセージ ID フィルタとして使用され、SF2ID/EF2ID はフィルタマスクとして使用されます。

フィルタ･マスクでの 0 ビットは、設定された ID フィルタの対応するビット位置をマスクします。たとえば、受信したメッセージ ID の中で、このマスク･ビットが 0 であるビット位置の値には許可フィルタが作用しません。受信したメッセージ ID の各ビットのうち、対応するマスク ビットが 1 のビットにのみ許可フィルタが作用します。

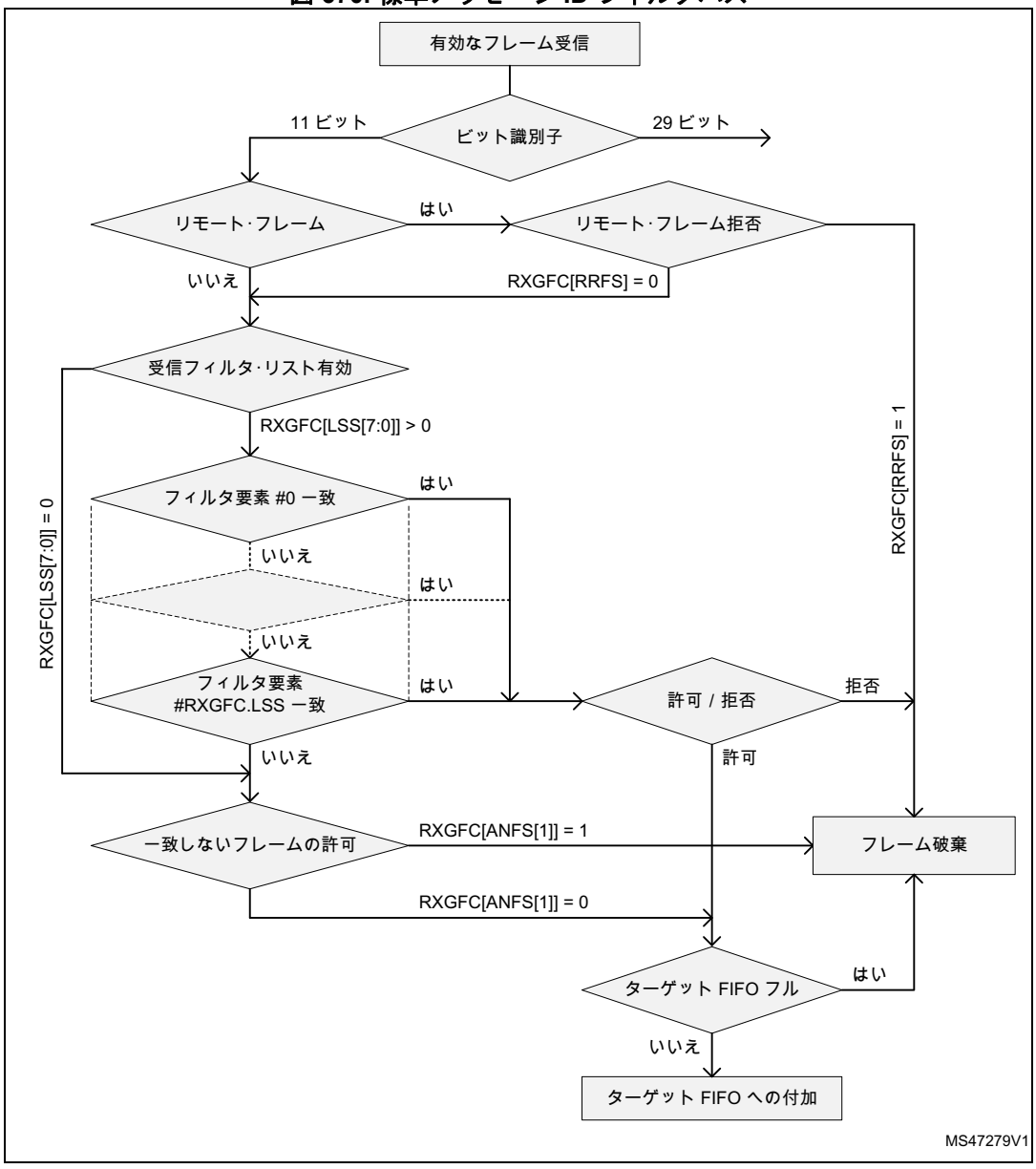
すべてのマスクビットが 1 の場合、受信したメッセージ ID とメッセージ ID フィルタが同一である場合のみ一致が発生します。すべてのマスクビットが 0 の場合、すべてのメッセージ ID が一致します。

標準メッセージ ID フィルタ

図 670 では、標準メッセージ ID（11 ビット識別子）フィルタのフローを示します。標準メッセージ ID フィルタ要素は、セクション 44.3.8： を参照してください。

グローバルフィルタ設定（RXGFC）メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）は設定されたフィルタ要素のリストと比較されます。

図 670. 標準メッセージ ID フィルタパス

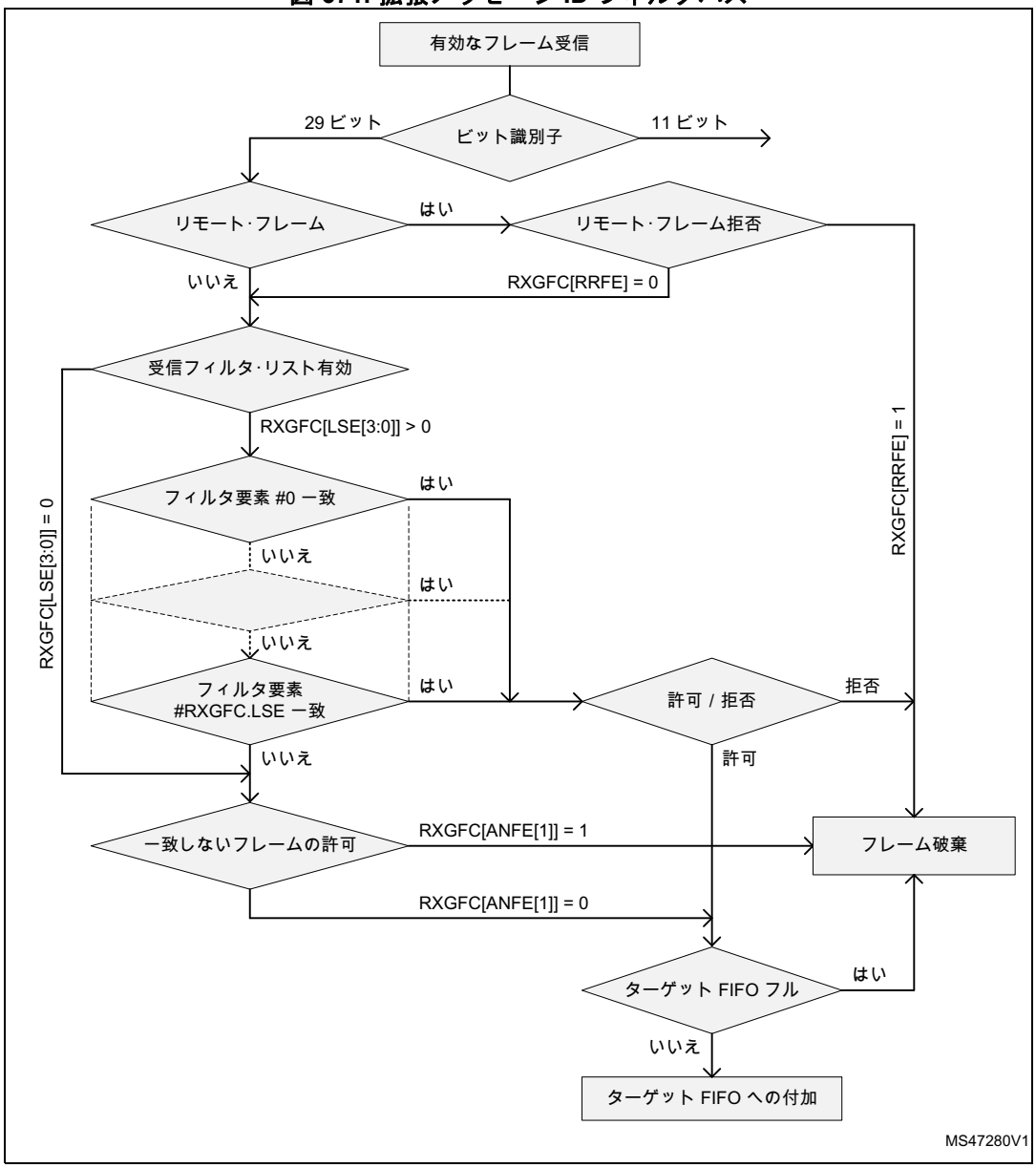


拡張メッセージ ID フィルタ

図 671 では、拡張メッセージ ID（29 ビット識別子）フィルタのフローを示します。拡張メッセージ ID フィルタ要素は、セクション 44.3.9 : を参照してください。

グローバルフィルタ設定 RXGFC および拡張 ID フィルタ設定 RXGFC メッセージ ID で制御されている場合、受信フレームのリモート送信リクエストビット（RTR）と識別子拡張ビット（IDE）は設定されたフィルタ要素のリストと比較されます。

図 671. 拡張メッセージ ID フィルタパス



フィルタリストが実行される前に、拡張 ID AND マスク（XIDAM）は受信識別子と AND でつながれます。

Rx FIFO

Rx FIFO 0 および Rx FIFO 1 は、それぞれ最大 3 個の要素を保持できます。

許可フィルタを通過した受信メッセージは、一致フィルタ要素で設定されたとおりに Rx FIFO に転送されます。Rx FIFO 0 および Rx FIFO 1 で使用できるフィルタメカニズムの詳細については、許可フィルタを参照してください。Rx FIFO 要素は、セクション 44.3.5 : を参照してください。

Rx FIFO フル条件が IR[RFnF] によって信号送信されると、最低 1 つのメッセージが読み出され、Rx FIFO Get インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、割込みフラグ IR[RFnL] がセットされます。

Rx FIFO から読み出すときに、Rx FIFO Get インデックス RXFnS[FnGI] + FIFO 要素サイズを対応する Rx FIFO 開始アドレス [FnSA] に追加する必要があります。

Rx FIFO ブロックモード

Rx FIFO ブロックモードは、RXGFC.FnOM = 0 で設定されます。これは、Rx FIFO のデフォルトの動作モードです。

Rx FIFO フル条件（RXFnS.FnPI = RXFnS.FnGI）に達すると、最低 1 つのメッセージが読み出され、Rx FIFO Get インデックスがインクリメントされるまで、対応する Rx FIFO に追加のメッセージは書き込まれません。Rx FIFO フル条件は、RXFnS.FnF = 1 で信号送信されます。さらに、割込みフラグ IR.RFnF がセットされます。

対応する Rx FIFO がフルになっているときにメッセージを受信した場合、このメッセージは破棄され、メッセージ喪失条件が RXFnS.RFnL = 1 で信号送信されます。さらに、割込みフラグ IR.RFnL がセットされます。

Rx FIFO 上書きモード

Rx FIFO 上書きモードは、RXGFC.FnOM = 1 で設定されます。

Rx FIFO がフルになっている状態（RXFnS.FnPI = RXFnS.FnGI）が RXFnS.FnF = 1 によって通知されると、FIFO に許可された次のメッセージによって一番古い FIFO メッセージが上書きされます。Put インデックスと Get インデックスはいずれも 1 でインクリメントされます。

Rx FIFO が上書きモードで動作しているときに、Rx FIFO がフルである状態が通知された場合、Rx FIFO 要素の読出しを Get インデックス + 1 以上で開始する必要があります。これは、CPU がメッセージ RAM（Get インデックス）から読み出しているときに、受信メッセージがメッセージ RAM（Put インデックス）に書き込まれる状況を防止するためです。この場合、一貫していないデータがそれぞれの Rx FIFO 要素から読み出される場合があります。Rx FIFO から読出し時に 取得インデックスにオフセットを加えることで、この問題を回避しています。このオフセットは、CPU が Rx FIFO にアクセスする速度に依存します。

Rx FIFO から読み出した後、最後の要素読出しの数を Rx FIFO 確認応答インデックス RXFnA.FnA に書き込む必要があります。これにより、Get インデックスがその要素数にインクリメントされます。プット インデックスがこの Rx FIFO 要素にインクリメントされなかった場合、Rx FIFO フル条件はリセットされます（RXFnS.FnF = 0）。

Tx の処理

Tx ハンドラは、Tx FIFO および Tx キューの送信リクエストを処理します。CAN コア、Put および Get インデックス、Tx イベント FIFO への送信メッセージの転送を制御します。最大 3 個の Tx バッファをメッセージ送信用に設定できます。CAN メッセージ･データ･フィールドは 64 バイトに設定され、Tx FIFO は Tx 要素の格納に 18 個の 32 ビット･ワードを配置します。

表 399. フレーム送信の可能な設定

CCCR		Tx バッファ要素		フレーム送信
BRSE	FDOE	FDF	BRS	
無視	0	無視	無視	クラシック CAN
0	1	0	無視	クラシック CAN
0	1	1	無視	ビットレート切り替えなしの FD
1	1	0	無視	クラシック CAN
1	1	1	0	ビットレート切り替えなしの FD
1	1	1	1	ビットレート切り替えありの FD

注： **AUTOSAR は最低 3つの Tx キュー･バッファと送信キャンセルのサポートが必須です。**

Tx ハンドラは、Tx バッファ･リクエスト･ペンディング･レジスタ TXBRP が更新されたときや、送信が開始されたときに、Tx スキャンを開始して、最優先のペンディング中の Tx リクエスト（一番小さいメッセージ ID の Tx バッファ）をチェックします。

送信一時停止

送信一時停止機能は、CAN メッセージ識別子が（完全に）固有の値に指定され、容易に変更できない CAN システムで使用することを目的とします。これらのメッセージ識別子は、他の定義されたメッセージより高い CAN アービトレーション優先度である場合がありますが、特定のアプリケーションでは、アービトレーションの相対的な優先度を逆にする必要があります。これにより、ある ECU が CAN メッセージをバースト送信すると、別の ECU CAN メッセージの処理が遅延することがあります。その別のメッセージの CAN アービトレーション優先度の方が低いためです。

たとえば、この機能を有効にしている CAN ECU-1 が、4 件のメッセージの送信をアプリケーション･ソフトウェアからリクエストされた場合、最初のメッセージ送信が成功した後、リクエストされた次のメッセージの開始が許可されるまでバス･アイドル状態で 2 CAN ビット時間待機します。ペンディング中のメッセージを保持する他の ECU がある場合、それらのメッセージはこのアイドル時間内に開始されるので、ECU-1 の次のメッセージとアービトレーションする必要がありません。メッセージを受信した後、受信メッセージが CAN バスを開放すると、ECU-1 は次の送信開始が許可されます。

この機能は、CCCR レジスタの TXP ビットで制御されます。このビットがセットされている場合、FDCAN は、メッセージの送信に成功するたびに、次の送信を開始する前に 2 CAN ビット時間、一時停止します。これにより、ネットワーク内の他の CAN ノードは、低い優先度識別子を持つメッセージであっても送信できます。デフォルトでは無効です（CCCR.TXP = 0）。

この機能は、1 つのノードからのバースト送信を緩和し、アプリケーションプログラムが誤って大量の送信をリクエストする「バブリングイディオット」シナリオから保護します。

Tx FIFO

Tx FIFO 動作は、TXBC[TFQM] を 0 にプログラムすることによって設定されます。Tx FIFO に格納されたメッセージは、Get インデックス TXFQS[TFGI] によって参照されるメッセージから送信されます。各送信後、Get インデックスは Tx FIFO がエンプティになるまで周期的にインクリメントされます。Tx FIFO は、メッセージが Tx FIFO に書き込まれた順番で、さまざまな Tx バッファからの同じメッセージ ID を持つメッセージを送信できます。FDCAN は、Tx FIFO フリー･レベル TXFQS[TFFL] を、Get インデックスと Put インデックスとの差として計算します。これは、使用可能な（空き）Tx FIFO 要素の数を示します。

新しい送信メッセージは、Put インデックス TXFQS[TFQPI] によって参照される Tx バッファを先頭として Tx FIFO に書き込まれる必要があります。リクエストの追加によって、次の空き Tx FIFO 要

素に Put インデックスがインクリメントされます。Put インデックスが Get インデックスに達すると、Tx FIFO フル（TXFQS[TFQF]= 1）の信号が送信されます。この場合、次のメッセージが送信され、Get インデックスがインクリメントされるまで、Tx FIFO に別のメッセージを書き込まないようにする必要があります。

1 つのメッセージが Tx FIFO に追加される場合、Tx FIFO Put インデックスによって参照される Tx バッファに関連した TXBAR ビットに 1 を書き込むことで送信がリクエストされます。

複数（n）件のメッセージが Tx FIFO に追加される場合、Put インデックスから始まる n 個の連続した Tx バッファに書き込まれます。送信は TXBAR 経由でリクエストされます。Put インデックスは周期的に n ずつインクリメントされます。リクエストされる Tx バッファの数は、Tx FIFO フリー･レベルで示された空きの Tx バッファの数を超えないことが必要です。

Get インデックスによって参照される Tx バッファに対する送信リクエストがキャンセルされた場合、Get インデックスは送信リクエストがペンディングされている次の Tx バッファまでインクリメントされ、Tx FIFO フリー･レベルが再計算されます。送信キャンセルが他の任意の Tx バッファに適用される場合、Get インデックスと FIFO フリー･レベルに変化はありません。

Tx FIFO 要素にはメッセージ RAM で 18 個の 32 ビット･ワードが割り当てられます。そのため、次に使用可能な（空き）Tx FIFO バッファの開始アドレスは、Put インデックス TXFQS[TFQPI]（0:2）を Tx バッファ開始アドレス [TBSA] に 4 回加算することで得られます。

Tx キュー

Tx キュー動作は、TXBC[TFQM] を 1 にプログラムすることによって設定されます。Tx キューに格納されたメッセージは、一番小さいメッセージ ID（最優先）のメッセージから送信されます。

標準メッセージ ID と拡張メッセージ ID が混在している場合、標準メッセージ ID は拡張メッセージ ID のビット [28:18] と比較されます。

複数のキュー･バッファに同じメッセージ ID が設定されている場合、一番小さいバッファ番号のキュー･バッファが先に送信されます。

新しいメッセージは、Put インデックス TXFQS[TFQPI] によって参照される Tx バッファに書き込まれる必要があります。リクエストを追加すると、次の空き Tx バッファまで周期的に Put インデックスがインクリメントされます。Tx キューがフル（TXFQS[TFQF] = 1）の場合、Put インデックスが有効ではないので、リクエストされているメッセージのうち最低 1 件が送信されるか、ペンディング中の送信リクエストがキャンセルされるまで、追加のメッセージを Tx キューに書き込まないようにする必要があります。

アプリケーションは Put インデックスの代わりにレジスタ TXBRP を使用し、送信リクエストをペンディングせずに 任意の Tx バッファにメッセージを配置できます。

Tx キュー･バッファには、メッセージ RAM で 18 個の 32 ビット･ワードが割り当てられます。そのため、次に使用可能な（空き）Tx キュー･バッファの開始アドレスは、Tx キュー Put インデックス TXFQS[TFQPI]（0:2）を Tx バッファ開始アドレス TBSA に 4 回加算することで得られます。

送信キャンセル

FDCAN は送信キャンセルをサポートします。リクエストされた送信を Tx キュー･バッファからキャンセルするには、ホストからレジスタ TXBCR の対応するビット位置（= Tx バッファ番号）に 1 を書き込む必要があります。送信キャンセルは Tx FIFO 動作を対象としません。

成功したキャンセルは､レジスタ TXBCF の対応するビットを 1 にセットすることで信号送信されます。

送信キャンセルがリクエストされても、Tx バッファからの送信がすでに進行中であれば、対応する TXBRP ビットはセットされたままとなります。送信に成功した場合、対応する TXBTO ビットと TXBCF ビットがセットされます。送信に成功しなかった場合、繰り返されず、対応する TXBCF ビットのみセットされます。

注： **ペンディング中の送信が開始される直前にキャンセルされた場合、別のメッセージがこのノードでペンディングされている場合でも送信が開始されない短いタイムウィンドウが続きます。これにより、別のノードはそのノードの 2 番目の優先度のメッセージより低い優先度のメッセージを送信できます。**

Tx イベントの処理

Tx イベントの処理をサポートするために、FDCAN は Tx イベント FIFO を実装しました。FDCAN が CAN バスでメッセージを送信すると、メッセージ ID およびタイムスタンプが Tx イベント FIFO 要素に格納されます。Tx イベントを Tx イベント FIFO 要素にリンクするために、送信された Tx バッファのメッセージ･マーカーが Tx イベント FIFO 要素にコピーされます。

Tx イベント FIFO は、3 個の要素に設定できます。Tx イベント FIFO 要素については、Tx FIFO を参照してください。

Tx イベント FIFO の目的は、Tx バッファには送信すべきメッセージのみを保持し、送信ステータスは Tx イベント FIFO に別に格納することにより、送信ステータス情報の処理と送信メッセージ処理を切り離すことにあります。これには、特に動的に管理される送信キューを操作する際に、Tx バッファが送信成功直後に新しいメッセージに使用できるという利点があります。Tx バッファを上書きする前に、その Tx バッファからの送信ステータス情報を保存する必要はありません。

Tx イベント FIFO がフルである状態が IR[TEFF] によって通知されると、最低 1 つの要素が読み出され、Tx イベント FIFO Get インデックスがインクリメントされるまで、Tx イベント FIFO に別のメッセージは書き込まれません。Tx イベント FIFO がフルになっているときに Tx イベントが発生した場合、このイベントは破棄され、割込みフラグ IR[TEFL] がセットされます。

Tx イベント FIFO から読み出すときに、Tx イベント FIFO Get インデックス TXEFS[EFGI] の 2 倍の値を Tx イベント FIFO 開始アドレス EFSA に加算する必要があります。

44.3.4 FIFO 確認応答の処理

Rx FIFO 0、Rx FIFO 1、および Tx イベント FIFO の Get インデックスは、対応する FIFO 確認応答インデックスへの書込みによって制御されます。セクション 44.4.23: およびセクション 44.4.25: を参照してください。FIFO 確認応答インデックスへの書込みによって FIFO Get インデックスが FIFO 確認応答インデックス +1 に設定され、これによって FIFO フィル･レベルが更新されます。これには、2 つの使用例があります。

1. 要素が 1 つだけ FIFO（Get インデックスによって指定されているもの）から読み出された場合、この Get インデックス値は FIFO 確認応答インデックスに書き込まれます。
2. 一連の要素が FIFO から読み出された場合、読出しシーケンスの最後に 1 回だけ FIFO 確認応答インデックス（値：最後の要素読出しのインデックス）を書き込んで、FIFO Get インデックスを更新するだけで十分です。

CPU が FDCAN メッセージ RAM に自由にアクセスできるため、任意の（Get インデックスが考慮されない）順番で FIFO 要素を読み出す際に特別な注意が必要です。これは、2つの Rx FIFO のうちの1つから高優先メッセージを読み出すときに役立つ場合があります。この場合、Get インデックスが誤った位置に設定され、FIFO フィル･レベルも変わるため、FIFO 確認応答インデックスには書き込まないようにする必要があります。この場合、古いほうの FIFO 要素の一部は失われます。

注： **アプリケーションは、有効な値が FIFO 確認応答インデックスに書き込まれていることを確認する必要があります。FDCAN は誤った値をチェックしません。**

44.3.5 FDCAN Rx FIFO 要素

2 つの Rx FIFO がメッセージ RAM に設定されています。各 Rx FIFO セクションを設定して、最大 3 個の受信メッセージを格納できます。Rx FIFO 要素の構造は、表 400 に示され、表 401 に説明が示されています。

表 400. Rx FIFO 要素

ビット	31			24	23			16	15	8	7	0
R0	ESI	XTD	RTR	ID[28:0]								
R1	ANMF	FIDX[6:0]			Res.	FDF	BRS	DLC[3:0]	RXTS[15:0]			
R2	DB3[7:0]				DB2 [7:0]				DB1 [7:0]		D[7:0]	
R3	DB7[7:0]				DB6 [7:0]				DB5 [7:0]		DB4 [7:0]	
⋮	⋮				⋮				⋮			
Rn	DBm[7:0]				DBm-1[7:0]				DBm-2[7:0]		DBm-3[7:0]	

CAN FD メッセージの格納について設定された要素サイズは、64 バイトのデータフィールドに設定されます。

表 401. Rx FIFO 要素の説明

フィールド	説明
R0 ビット 31 ESI	エラー状態インジケータ – 0：送信中のノードはエラーアクティブです。 – 1：送信中のノードはエラーパッシブです。
R0 ビット 30 XTD	拡張識別子 受信フレームに標準識別子または拡張識別子があるかどうかをホストに信号送信します。 – 0：11 ビット標準識別子です。 – 1：29 ビット拡張識別子です。
R0 ビット 29 RTR	リモート送信リクエスト 受信フレームがデータフレームかリモートフレームかどうかをホストに信号送信します。 – 0：受信フレームはデータフレームです。 – 1：受信フレームはリモートフレームです。
R0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に保存されます。
R1 ビット 31 ANMF	許可される不一致フレーム 一致しないフレームの許可は、RXGFC[ANFS] および RXGFC[ANFE] 経由で有効にできます。 – 0：受信フレームがフィルタインデックスに一致します。 – 1：受信フレームがどの Rx フィルタ要素にも一致しません。
R1 ビット 30:24 FIDX[6:0]	フィルタインデックス 0-27 = 一致している Rx 許可フィルタ要素のインデックス（ANMF = 1 の場合は無効）。 範囲は、0～RXGFC[LSS] - 1 または RXGFC[LSE] - 1 です。
R1 ビット 21 FDF	FD フォーマット – 0：標準フレームフォーマットです。 – 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC）
R1 ビット 20 BRS	ビット･レート切り替え – 0：ビット･レート切り替えなしで受信したフレームです。 – 1：ビット･レート切り替えありで受信したフレームです。

表 401. Rx FIFO 要素の説明 （続き）

フィールド	説明
R1 ビット 19:16 DLC[3:0]	データ長コード – 0～8：クラシック CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 – 9～15：クラシック CAN：受信フレームに 8 データバイトがあります。 – 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
R1 ビット 15:0 RXTS[15:0]	Rx タイムスタンプ フレーム受信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ TSCC[TCP] の設定に依存します。
R2 ビット 31:24 DB3[7:0]	データバイト 3
R2 ビット 23:16 DB2[7:0]	データバイト 2
R2 ビット 15:8 DB1[7:0]	データバイト 1
R2 ビット 7:0 D[7:0]	データバイト 0
R3 ビット 31:24 DB7[7:0]	データバイト 7
R3 ビット 23:16 DB6[7:0]	データバイト 6
R3 ビット 15:8 DB5[7:0]	データバイト 5
R3 ビット 7:0 DB4[7:0]	データバイト 4
⋮	⋮
Rn ビット 31:24 DBm[7:0]	データバイト m
Rn ビット 23:16 DBm-1[7:0]	データバイト m-1
Rn ビット 15:8 DBm-2[7:0]	データバイト m-2
Rn ビット 7:0 DBm-3[7:0]	データバイト m-3

44.3.6 FDCAN の Tx バッファ要素

Tx バッファ･セクション（3 つの要素）は Tx FIFO または Tx キューを保持するように設定できます。Tx ハンドラは Tx FIFO と Tx キューを Tx バッファ設定 FDCAN_TXBC.TFQM を使用して区別します。要素サイズは、最大 64 バイトのデータがある CAN FD メッセージを格納できるように設定されます。

表 402. Tx バッファおよび FIFO 要素

ビット	31 24	23 16	15 8	7 0
T0	ESI / XTD / RTR / ID[28:0]			
T1	MM[7:0]	EFC / Res. / FDF / BPS / DLC[3:0]	Res.	
T2	DB3[7:0]	DB2 [7:0]	DB1 [7:0]	D[7:0]
T3	DB7[7:0]	DB6 [7:0]	DB5 [7:0]	DB4 [7:0]
⋮	⋮	⋮	⋮	
Tn	DBm[7:0]	DBm-1[7:0]	DBm-2[7:0]	DBm-3[7:0]

表 403. Tx バッファ要素の説明

フィールド	説明
T0 ビット 31 ESI(1)	エラー状態インジケータ – 0：CAN FD フォーマットの ESI ビットはエラーパッシブフラグにのみ依存します。 – 1：CAN FD フォーマットの ESI ビットはリセッシブで送信されました。
T0 ビット 30 XTD	拡張識別子 – 0：11 ビット標準識別子です。 – 1：29 ビット拡張識別子です。
T0 ビット 29 RTR(2)	リモート送信リクエスト – 0：データフレームを送信します。 – 1：リモートフレームを送信します。
T0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
T1 ビット 31:24 MM[7:0]	メッセージマーカー Tx バッファ設定中に CPU によって書き込まれます。Tx メッセージステータスの識別のために Tx イベント FIFO 要素にコピーされます。
T1 ビット 23 EFC	イベント FIFO 制御 – 0：Tx イベントを保存しません。 – 1：Tx イベントを保存します。
T1 ビット 21 FDF	FD フォーマット – 0：クラシック CAN フォーマットでフレームが送信されます。 – 1：CAN FD フォーマットでフレームが送信されます。
T1 ビット 20 BRS(3)	ビットレート切り替え – 0：ビット･レート切り替えなしで送信された CAN FD フレームです。 – 1：ビット･レート切り替えありで送信された CAN FD フレームです。

表 403. Tx バッファ要素の説明 （続き）

フィールド	説明
T1 ビット 19:16 DLC[3:0]	データ長コード – 0～8：クラシック CAN + CAN FD：受信フレームに 0～8 のデータバイトがあります。 – 9～15：クラシック CAN：受信フレームに 8 データバイトがあります。 – 9～15：CAN FD：受信フレームに 12/16/20/24/32/48/64 のデータバイトがあります。
T2 ビット 31:24 DB3[7:0]	データバイト 3
T2 ビット 23:16 DB2[7:0]	データバイト 2
T2 ビット 15:8 DB1[7:0]	データバイト 1
T2 ビット 7:0 D[7:0]	データバイト 0
T3 ビット 31:24 DB7[7:0]	データバイト 7
T3 ビット 23:16 DB6[7:0]	データバイト 6
T3 ビット 15:8 DB5[7:0]	データバイト 5
T3 ビット 7:0 DB4[7:0]	データバイト 4
⋮	⋮
Tn ビット 31:24 DBm[7:0]	データバイト m
Tn ビット 23:16 DBm-1[7:0]	データバイト m-1
Tn ビット 15:8 DBm-2[7:0]	データバイト m-2
Tn ビット 7:0 DBm-3[7:0]	データバイト m-3

1. 送信バッファの ESI ビットは、エラー･パッシブ･フラグと論理和がとられ、送信される FD フレームでの ESI ビットの値を決定します。CAN FD プロトコルの仕様によって必須とされるとおり、エラー･アクティブ･ノードでは必要に応じて ESI ビット･リセッシブを送信できますが、エラー･パッシブ･ノードでは必ず ESI ビット･リセッシブを送信します。
2. RTR = 1 のとき、CCCR.FDOE が CAN FD フォーマットでの送信を有効にしている場合でも、FDCAN は ISO11898-1 に従ってリモート･フレームを送信します。
3. ビット ESI、FDF、BRS は CAN FD 動作が有効（CCCR.FDOE = 1）のときのみ評価されます。ビット BRS はさらに CCCR.BRSE = 1 のときのみ評価されます。

44.3.7 FDCAN Tx イベント FIFO 要素

各要素には、送信されたメッセージに関する情報が保存されています。Tx イベント FIFO を読み出すことで、ホスト CPU はメッセージが送信された順にこの情報を取得します。Tx イベント FIFO に関するステータス情報は、レジスタ TXEFS から取得できます。

表 404. Tx イベント FIFO 要素

ビット	31			24	23			16	15	8	7	0
E0	ESI	XTD	RTR		ID[28:0]							
E1	MM[7:0]				ET[1:0]	EDL	BRS	DLC[3:0]	TXTS[15:0]			

表 405. Tx イベント FIFO 要素の説明

フィールド	説明
E0 ビット 31 ESI	エラー状態インジケータ – 0：送信中のノードはエラーアクティブです。 – 1：送信中のノードはエラーパッシブです。
E0 ビット 30 XTD	拡張識別子 – 0：11 ビット標準識別子です。 – 1：29 ビット拡張識別子です。
E0 ビット 29 RTR	リモート送信リクエスト – 0：データフレームを送信します。 – 1：リモートフレームを送信します。
E0 ビット 28:0 ID[28:0]	識別子 ビット XTD に応じた標準識別子または拡張識別子です。標準識別子は、ID[28:18] に書き込まれる必要があります。
E1 ビット 31:24 MM[7:0]	メッセージマーカー Tx メッセージステータスの識別のために Tx バッファから Tx イベント FIFO 要素にコピーされます。
E1 ビット 23:22 EFC	イベント･タイプ – 00：予約済みです。 – 01：Tx イベント – 10：キャンセル時でも送信（DAR モードでの送信では常にセット） – 11：予約済みです。
E1 ビット 21 EDL	拡張データ長 – 0：標準フレームフォーマットです。 – 1：FDCAN フレームフォーマット（新しい DLC コーディングおよび CRC）
E1 ビット 20 BRS	ビット･レート切り替え – 0：ビット･レート切り替えなしで送信されたフレームです。 – 1：ビット･レート切り替えありで送信されたフレームです。
T1 ビット 19:16 DLC[3:0]	データ長コード 0～8：0～8 のデータバイトのあるフレームが送信されます。 9～15：8データバイトのあるフレームが送信されます。
E1 ビット 15:0 TXTS[15:0]	Tx タイムスタンプ フレーム送信の開始時にキャプチャされるタイムスタンプカウンタ値です。分解能はタイムスタンプカウンタプリスケーラ TSCC[TCP] の設定に依存します。

44.3.8 FDCAN 標準メッセージ ID フィルタ要素

最大 28 のフィルタ要素を 11 ビット標準 ID に設定できます。標準メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの標準開始アドレス FLSSA とフィルタ要素のインデックス（0～27）です。

表 406. 標準メッセージ ID フィルタ要素

ビット	31 24		23 16	15 8	7 0
S0	SFT[1:0]	SFEC[2:0]	SFID1 [10:0]	Res.	SFID2 [10:0]

表 407. 標準メッセージ ID フィルタ要素フィールドの説明

フィールド	説明
ビット 31:30 SFT[1:0](1)	標準フィルタタイプ – 00：SFID1 から SFID2 までの範囲のフィルタ – 01：SFID1 または SFID2 のデュアル ID フィルタ – 10：クラシックフィルタ：SFID1 = フィルタ、SFID2 = マスク – 11：フィルタ要素無効化
ビット 29:27 SFEC[2:0]	標準フィルタ要素の設定 すべての有効化されたフィルタ要素は、標準フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。SFEC = 100、101、または 110 の場合、一致すると割込みフラグ IR.HPM がセットされ、有効であれば、割込みが生成されます。この場合、レジスタ HPMS は優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合に Rx FIFO 0 に格納 – 010：フィルタが一致した場合に Rx FIFO 1 に格納 – 011：フィルタが一致した場合に ID を拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 – 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 – 111：未使用
ビット 26:16 SFID1 [10:0]	標準フィルタ ID 1 標準 ID フィルタ要素の最初の ID
ビット 10:0 SFID2 [10:0]	標準フィルタ ID 2 標準 ID フィルタ要素の 2 番目の ID

1. SFT = 11 でフィルタ要素は無効化され、許可フィルタの適用が継続します（SFEC = 000 と同じ動作）。

注： **予約済みの値が設定されている場合、フィルタ要素は無効とみなされます。**

44.3.9 FDCAN 拡張メッセージ ID フィルタ要素

最大 8 のフィルタ要素を 29 ビット拡張 ID に設定できます。拡張メッセージ ID フィルタ要素にアクセスするとき、そのアドレスはフィルタリストの拡張開始アドレス FLESA に要素のインデックス（0～7）の 2 倍を加えたものです。

表 408. 拡張メッセージ ID フィルタ要素

ビット	31		24	23 16	15 8	7 0
F0	EFEC[2:0]		EFID1 [28:0]			
F1	EFTI[1:0]	Res.	EFID2 [28:0]			

表 409. 拡張メッセージ ID フィルタ要素フィールドの説明

フィールド	説明
F0 ビット 31:29 EFEC[2:0]	拡張フィルタ要素の設定 すべての有効化されたフィルタ要素は、拡張フレームの許可フィルタに使用されます。許可フィルタは、有効化されたフィルタ要素に最初に一致したとき、またはフィルタの最後に達した場合に停止します。EFEC = 100、101、または 110 の場合、一致すると割込みフラグ IR[HPM] がセットされ、有効であれば、割込みが生成されます。この場合、レジスタ HPMS は優先度一致のステータスで更新されます。 – 000：フィルタ要素の無効化 – 001：フィルタが一致した場合に Rx FIFO 0 に格納 – 010：フィルタが一致した場合に Rx FIFO 1 に格納 – 011：フィルタが一致した場合に ID を拒否 – 100：フィルタが一致した場合に優先度をセット – 101：フィルタが一致した場合に優先度をセットして Rx FIFO 0 に格納 – 110：フィルタが一致した場合に優先度をセットして Rx FIFO 1 に格納 – 111：未使用
F0 ビット 28:0 EFID1[28:0]	拡張フィルタ ID 1 拡張 ID フィルタ要素の最初の ID Rx FIFO のフィルタ時、このフィールドで格納する拡張メッセージの ID を定義します。受信した識別子は完全一致とし、XIDAM マスキングメカニズムのみです。
F1 ビット 31:30 EFT[1:0]	拡張フィルタタイプ – 00：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID） – 01：EF1ID または EF2ID のデュアル ID フィルタ – 10：クラシックフィルタ：EF1ID = フィルタ、EF2ID = マスク – 11：EF1ID から EF2ID までの範囲のフィルタ（EF2ID ≥ EF1ID）、XIDAM マスク適用なし
F1 ビット 29	未使用
F1 ビット 28:0 EFID2[28:0]	拡張フィルタ ID 2 標準 ID フィルタ要素の 2 番目の ID

44.4 FDCAN レジスタ

44.4.1 FDCAN コア開放レジスタ（FDCAN_CREL）

アドレスオフセット：0x0000

リセット値：0x3214 1218

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REL[3:0]				STEP[3:0]				SUBSTEP[3:0]				YEAR[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MON[7:0]								DAY[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:28 **REL[3:0]**：3

ビット 27:24 **STEP[3:0]**：2

ビット 23:20 **SUBSTEP[3:0]**：1

ビット 19:16 **YEAR[3:0]**：4

ビット 15:8 **MON[7:0]**：12

ビット 7:0 **DAY[7:0]**：18

44.4.2 FDCAN エンディアン･レジスタ（FDCAN_ENDN）

アドレスオフセット：0x0004

リセット値：0x8765 4321

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ETV[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETV[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **ETV[31:0]**：エンディアン･テスト値

エンディアン･テスト値は 0x8765 4321 です。

注： **エンディアンに問題がないことを確認するには、レジスタからリセット値を読み出す必要があります。**

44.4.3 FDCAN データ･ビット･タイミングおよびプリスケーラ･レジスタ（FDCAN_DBTP）

アドレスオフセット：0x000C

リセット値：0x0000 0A33

このレジスタは、CCCR.CCE および CCCR.INIT がセットされている場合のみ書込みできます。CAN タイムクォンタムは、1～32 FDCAN 周期の範囲でプログラム可能です。tq = (DBRP + 1) FDCAN クロック周期。

DTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。DTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さは（プログラムされた値）[DTSEG1 + DTSEG2 + 3] tq または（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDC	Res.	Res.	DBRP[4:0]				
								rw			rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0]			
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **TDC**：トランシーバ遅延補正

0：トランシーバ遅延補正は無効です。

1：トランシーバ遅延補正は有効です。

ビット 22:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **DBRP[4:0]**：データ･ビット･レート･プリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です。ビット時間は、このクォンタムの倍数で構築されます。ボーレートプリスケーラに有効な値は 0～31 です。ハードウェアはこの値をプログラムされた値 +1 として解釈します。

ビット 15:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **DTSEG1[4:0]**：サンプルポイント前のデータ時間セグメント

有効な値は 0～31 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS1 = (DTSEG1 + 1) x tq です。

ビット 7:4 **DTSEG2[3:0]**：サンプルポイント後のデータ時間セグメント

有効な値は 0～15 です。ハードウェアによって使用される値はプログラムされたもので、1ずつ増加します。例：tBS2 = (DTSEG2 + 1) x tq です。

ビット 3:0 **DSJW[3:0]**：同期ジャンプ幅

必ず DTSEG2 より小さくする必要があり、有効値は 0 ~ 15 です。ハードウェアによって使用される値はプログラムされたもので、1 ずつ増加します。例：t_{SJW} = (DSJW + 1) x tq です。

44.4.4 FDCAN テスト･レジスタ（FDCAN_TEST）

このレジスタへの書込みアクセスは、CCCR[TEST] を 1 に設定することで有効になります。CCCR[TEST] がリセットされると、すべてのレジスタ機能がそれぞれのリセット値に設定されます。

Tx ピン FDCANx_TX のループ･バック･モードおよびソフトウェア制御はハードウェア･テスト･モードです。TX を 00 以外にプログラムすると、CAN バスでのメッセージ転送を妨害します。

アドレスオフセット：0x0010

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX[1:0]		LBCK	Res.	Res.	Res.	Res.
								r	rw	rw	rw				

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **RX**：受信ピン
ピン FDCANx_RX の実際の値を監視します。
0：CAN バスはドミナントです（FDCANx_RX = 0）。
1：CAN バスはリセッシブです（FDCANx_RX = 1）。

ビット 6:5 **TX[1:0]**：送信ピンの制御
00：リセット値、FDCANx_TX TX は CAN コアによって制御され、CAN ビット時間の最後に更新されます。
01：サンプルポイントはピン FDCANx_TX で監視できます。
10：ピン FDCANx_TX でドミナント（0）レベルです。
11：ピン FDCANx_TX でリセッシブ（1）です。

ビット 4 **LBCK**：ループ･バック･モード
0：リセット値、ループ･バック･モードは無効です。
1：ループ･バック･モードは有効です（パワー･ダウン（SLEEP モード）を参照）。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

44.4.5 FDCAN RAM ウォッチドッグレジスタ（FDCAN_RWD）

RAM ウォッチドッグはメッセージ RAM の READY 出力を監視します。メッセージ RAM アクセスは、RWD[WDC] ビットで設定された値でメッセージ RAM ウォッチドッグカウンタを開始します。

メッセージ RAM が READY 出力を有効にすることで完了成功の信号を送信すると、カウンタは RWD[WDC] ビットで再ロードされます。カウンタが 0 までのカウントダウンを完了するまでにメッセージ RAM からレスポンスがない場合、カウンタは停止し、割込みフラグ IR[WDI] ビットがセットされます。RAM ウォッチドッグカウンタは、fdcan_pclk クロックによってクロック供給されます。

アドレス･オフセット：0x0014

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDV[7:0]								WDC[7:0]							
r	r	r	r	r	r	r	r	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:8 **WDV[7:0]**：ウォッチドッグの値

実際のメッセージ RAM ウォッチドッグカウンタの値です。

ビット 7:0 **WDC[7:0]**：ウォッチドッグの設定

メッセージ RAM ウォッチドッグカウンタの開始値です。00 のリセット値で、カウンタは無効になります。

これらは書込み保護（P）ビットで、FDCAN_CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 に設定されているときのみ書込みアクセス可能です。

44.4.6 FDCAN CC 制御レジスタ（FDCAN_CCCR）

アドレスオフセット：0x0018

リセット値：0x0000 0001

シングル･ビットのセットおよびリセットの詳細については、ソフトウェアの初期化を参照してください。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INIT
rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	r	rw	rw	rw

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **NISO**：非 ISO 動作

このビットがセットされた場合、FDCAN は Bosch CAN FD Specification V1.0 によって指定された CAN FD フレームフォーマットを使用します。

0：ISO11898-1 に準拠した CAN FD フレームフォーマットです。

1：Bosch CAN FD Specification V1.0 に準拠した CAN FD フレームフォーマットです。

ビット 14 **TXP**：

このビットがセットされていると、FDCAN はフレームの送信成功後、次の送信を開始する前に 2 CAN ビット時間、一時停止します。

0：無効

1：有効

ビット 13 **EFBI**：バス統合中のエッジ･フィルタ

0：エッジフィルタは無効です。

1：2 つの連続したドミナント tq は、ハード同期のためにエッジを検出する必要があります。

ビット 12 **PXHD**：プロトコル例外処理ディスエーブル

0：プロトコル例外処理は有効です。

1：プロトコル例外処理は無効です。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **BRSE**：FDCAN ビット･レート切り替え

0：送信のビット･レート切り替えは無効です。

1：送信のビット･レート切り替えは有効です。

ビット 8 **FDOE**：FD 動作イネーブル

0：FD 動作は無効です。

1：FD 動作は有効です。

ビット 7 **TEST**：テスト･モード･イネーブル

0：通常動作、レジスタ TEST はリセット値を保持します。

1：テスト･モード、レジスタ TEST への書込みアクセスは有効です。

ビット 6 **DAR**：自動再送信の無効化

0：正常に送信されなかったメッセージの自動再送信は有効です。

1：自動再送信は無効です。

ビット 5 **MON**：バス･モニタ･モード

MON ビットは、CCE および INIT の両方が 1 にセットされているときのみソフトウェアによってセットできます。このビットはいつでもホストによってリセットできます。

0：バス･モニタ･モードは無効です。

1：バス･モニタ･モードは有効です。

ビット 4 **CSR**：クロック停止リクエスト

0：クロック停止はリクエストされていません。

1：クロック停止がリクエストされました。クロック停止がリクエストされたとき、すべてのペンディング中の転送リクエストが完了して CAN バスがアイドルに達した後、まず INIT がセットされ、次に CSA がセットされます。

ビット 3 **CSA**：クロック停止の確認応答

0：クロック停止の確認応答はありません。

1：FDCAN は、APB クロックおよびカーネル･クロックを停止することで、パワー･ダウンに設定できます。

ビット 2 **ASM**：ASM 制限動作モード

制限動作モードは、さまざまな CAN ビット･レートに適応するアプリケーションを対象とします。アプリケーションはさまざまなビット･レートをテストし、有効なフレームを受信すると制限動作モードを終了します。オプションの制限動作モードでは、ノードはデータとリモート･フレームを送信および受信でき、有効なフレームに確認応答を返しますが、アクティブ･エラー･フレームとオーバーロード･フレームは送信しません。エラー状態やオーバーロード状態になった場合、ドミナント･ビットを送らずに、バス･アイドル状態の発生を待って CAN 通信と再同期します。エラーカウンタはインクリメントされません。ASM ビットは、CCE および INIT の両方が 1 にセットされているときのみソフトウェアによってセットできます。このビットはいつでもソフトウェアによってリセットできます。

0：通常 CAN 動作です。

1：制限動作モードはアクティブです。

ビット 1 **CCE**：設定変更の有効化

0：保護された設定レジスタへの書込みアクセス権が CPU にありません。

1：保護された設定レジスタへの書込みアクセス権が CPU にあります（CCCR.INIT = 1 のとき）。

ビット 0 **INIT**：初期化

0：通常動作

1：初期化が開始されました。

注： **2 つのクロック･ドメインの同期メカニズムによって、INIT に書き込まれた値を読み戻せるようになるまでに遅延が発生する場合があります。そのため、プログラマは INIT を新しい値にセットする前に、INIT を読み出して INIT に書き込まれた前の値が許可されたことを確認する必要があります。**

44.4.7 FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）

アドレスオフセット：0x001C

リセット値：0x0600 0A03

このレジスタは､CCCR[CCE] および CCCR[INIT] がセットされている場合のみ書込みできます｡CAN ビット時間は､4～81 tq の範囲でプログラム可能です｡CAN タイムクォンタムは､[1～1024] FDCAN カーネルクロック周期の範囲でプログラム可能です。

tq = (BRP + 1) FDCAN クロック周期 fdcan_clk

NTSEG1 は、Prop_Seg と Phase_Seg1 の合計です。NTSEG2 は、Phase_Seg2 です。そのため、ビット時間の長さは（プログラムされた値）[NTSEG1 + NTSEG2 + 3] tq または（機能の値）[Sync_Seg + Prop_Seg + Phase_Seg1 + Phase_Seg2] tq となります。

情報処理時間（IPT）は 0 で、次のビットのデータはサンプルポイント後の最初のクロックエッジで使用できることを示します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NSJW[6:0]							NBRP[8:0]								
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NTSEG1 [7:0]								Res.	NTSEG2 [6:0]						
rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:25 **NSJW[6:0]**：公称（再）同期ジャンプ幅

有効な値は 0 ~ 127 です。この値のハードウェアによる実際の解釈は、使用される値が 1 でインクリメントされたプログラムされた値です。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 に設定されているときのみ書込みアクセス可能です。

ビット 24:16 **NBRP[8:0]**：ビット･レート･プリスケーラ

ビット時間のクォンタムを生成するためにオシレータ周波数を分周する値です｡ビット時間は､このクォンタムの倍数で構築されます｡有効な値は 0～511 です｡この値のハードウェアによる実際の解釈は、ここでプログラムされた値より 1 つ大きな値が使用されます。

これらは書込み保護（P）ビットで､CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:8 **NTSEG1[7:0]**：サンプルポイント前の公称時間セグメント

有効な値は 0～255 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

これらは書込み保護（P）ビットで､CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ書込みアクセス可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **NTSEG2[6:0]**：サンプルポイント後の公称時間セグメント

有効な値は 0～127 です。この値のハードウェアによる実際の解釈は、プログラムされた値より 1 つ大きな値が使用されます。

注： **48 MHz の CAN カーネル･クロックでは、0x06000A03 のリセット値によって FDCAN のビット･レートが 3 Mbit/s に設定されます。**

44.4.8 FDCAN タイムスタンプ･カウンタ設定レジスタ（FDCAN_TSCC）

アドレスオフセット：0x0020

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS[1:0]	
														rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:16 **TCP[3:0]**：タイムスタンプカウンタプリスケーラ

タイムスタンプとタイムアウト･カウンタの時間単位を CAN ビット時間の倍数 [1 ~ 16] で設定します。ハードウェアによるこの値の実際の解釈は、ここでプログラムされた値より 1 大きい値が使用されるということです。

CAN FD モードでは、アービトレーションフェーズとデータフェーズで異なる CAN ビット時間のため、内部タイムスタンプカウンタ TCP は一定の時間ベースを提供しません。CAN FD では、タイムスタンプ生成用の外部カウンタが必要です（TSS = 10）。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ書込みアクセス可能です。

ビット 15:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **TSS[1:0]**：タイムスタンプの選択

00：タイムスタンプカウンタ値は常に 0x0000 です。

01：タイムスタンプカウンタ値は TCP に従ってインクリメントされます。

10：TIM3 の値（tim3_cnt[0:15]）から始まる外部タイムスタンプ･カウンタ。

11：00 と同様です。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ書込みアクセス可能です。

44.4.9 FDCAN タイムスタンプカウンタ値レジスタ（FDCAN_TSCV）

アドレスオフセット：0x0024

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSC[15:0]															
rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TSC[15:0]**：タイムスタンプカウンタ

内部／外部タイムスタンプカウンタ値は、フレーム受信の開始時（Rx および Tx の両方）にキャプチャされます。TSCC[TSS] = 01 の場合、タイムスタンプ･カウンタは、TSCC[TCP] の設定に応じて CAN ビット時間の倍数 [1 ~ 16] でインクリメントされます。ラップアラウンドによって、割込みフラグ IR[TSW] がセットされます。書込みアクセスによって、カウンタは 0 にリセットされます。

TSCC.TSS = 10 の場合、TSC には外部タイムスタンプ･カウンタ値が反映されます。書込みアクセスによる影響はありません。

注： **「ラップアラウンド」とは、TSCV への書込みアクセス以外で、0 以外から 0 へのタイムスタンプカウンタ値の変化を指します。**

44.4.10 FDCAN タイムアウトカウンタ設定レジスタ（FDCAN_TOCC）

アドレスオフセット：0x0028

リセット値：0xFFFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TOP[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS[1:0]		ETOC
													rw	rw	rw

ビット 31:16 **TOP[15:0]**：タイムアウト期間

タイムアウトカウンタ（ダウンカウンタ）の開始値です。タイムアウト期間を設定します。

ビット 15:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:1 **TOS[1:0]**：タイムアウト選択

連続モードで動作している場合、TOCV への書込みによって、カウンタは TOCC[TOP] で設定された値にプリセットされ、カウントダウンを続行します。タイムアウト･カウンタがいずれか 1つの FIFO で制御されている場合、空の FIFO は TOCC[TOP] で設定された値にカウンタをプリセットします。カウントダウンは最初の FIFO 要素が格納されたときに開始します。

00：連続動作

01：Tx イベント FIFO によって制御されるタイムアウト

10：Rx FIFO 0 によって制御されるタイムアウト

11：Rx FIFO 1 によって制御されるタイムアウト

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ書込みアクセス可能です。

ビット 0 **ETOC**：タイムアウト･カウンタ有効化

0：タイムアウトカウンタは無効です。

1：タイムアウトカウンタは有効です。

これは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 に設定されているときのみ書込みアクセス可能です。

詳細については、タイムアウトカウンタを参照してください。

44.4.11 FDCAN タイムアウトカウンタ値レジスタ（FDCAN_TOCV）

アドレスオフセット：0x002C

リセット値：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOC[15:0]															
rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w	rc_w

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **TOC[15:0]**：タイムアウトカウンタ

タイムアウト･カウンタは、TSCC.TCP の設定に応じて CAN ビット時間の倍数 [1 ~ 16] でデクリメントされます。0 までデクリメントされると、割込みフラグ IR.TOO がセットされ、タイムアウト･カウンタが停止します。開始条件とリセット／再開条件は、TOCC.TOS で設定します。

44.4.12 FDCAN エラーカウンタレジスタ（FDCAN_ECR）

アドレスオフセット：0x0040

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]							
								rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r	rc_r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RP	REC[6:0]							TEC[7:0]							
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **CEL[7:0]**：CAN エラーロギング

CAN プロトコルエラーによって送信エラーカウンタまたは受信エラーカウンタがインクリメントされるたびにカウンタがインクリメントされます。CEL の読出しアクセスすることによってリセットされます。カウンタは 0xFF で停止し、TEC または REC の次のインクリメントによって割込みフラグ IR[ELO] がセットされます。

アクセスタイプは RX で、読出し時にリセットされます。

ビット 15 **RP**：受信エラー･パッシブ

0：受信エラー･カウンタが 128 のエラー･パッシブ･レベル未満です。

1：受信エラー･カウンタが 128 のエラー･パッシブ･レベルに達しています。

ビット 14:8 **REC[6:0]**：受信エラーカウンタ

受信エラーカウンタの実際の状態、値の範囲は 0～127 です。

ビット 7:0 **TEC[7:0]**：送信エラーカウンタ

送信エラーカウンタの実際の状態、値の範囲は 0～255 です。

CCCR.ASM がセットされている場合、CAN プロトコル･エラーが検出されても、CEL が引き続きインクリメントされていれば、CAN プロトコル･コントローラは TEC と REC をインクリメントしません。

44.4.13 FDCAN プロトコルステータスレジスタ（FDCAN_PSR）

アドレスオフセット：0x0044

リセット値：0x0000 0707

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]						
									r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PXE	REDL	RBRS	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	rc_r	rc_r	rc_r	rc_r	rs	rs	rs	r	r	r	r	r	rs	rs	rs

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **TDCV[6:0]**：トランスミッタ遅延補正値

セカンダリ･サンプル･ポイントの位置で、FDCAN_TX から FDCAN_RX までで測定された遅延と TDCR.TDCO の合計で定義されます。SSP の位置は、データフェーズでは送信されたビットとセカンダリサンプルポイントの間の最小タイムクォンタム（mtq）の数です。有効な値は 0～127 mtq です。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **PXE**：プロトコル例外イベント

0：最後の読出しアクセス以降に発生したプロトコル例外イベントはありません。

1：プロトコル例外イベントが発生しました。

ビット 13 **REDL**：受信 FDCAN メッセージ

このビットは、許可フィルタに関係なくセットされます。

0：このビットが CPU によってリセットされた時点以降、受信した FDCAN メッセージはありません。

1：EDL フラグがセットされた FDCAN フォーマットのメッセージを受信しました。

アクセス･タイプは RX で、読出し時にリセットされます。

ビット 12 **RBRS**：最後に受信した FDCAN メッセージの BRS フラグ

このビットは、許可フィルタに関係なく REDL とともにセットされます。

0：最後に受信した FDCAN メッセージには BRS フラグがセットされていません。

1：最後に受信した FDCAN メッセージには BRS フラグがセットされています。

アクセス･タイプは RX で、読出し時にリセットされます。

ビット 11 **RESI**：最後に受信した FDCAN メッセージの ESI フラグ

このビットは、許可フィルタに関係なく REDL とともにセットされます。

0：最後に受信した FDCAN メッセージには ESI フラグがセットされていません。

1：最後に受信した FDCAN メッセージには ESI フラグがセットされています。

アクセス･タイプは RX で、読出し時にリセットされます。

ビット 10:8 **DLEC[2:0]**：データの最後のエラー･コード

BRS フラグをセットした FDCAN フォーマット･フレームのデータ･フェーズで発生した最後のエラーのタイプです。コーディングは LEC と同じです。このフィールドは、BRS フラグをセットした FDCAN フォーマット･フレームがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。

アクセス･タイプは RS で、読出し時にセットされます。

ビット 7 **BO**：Bus_Off ステータス

0：FDCAN は、Bus_Off ではありません。

1：FDCAN は、Bus_Off 状態です。

ビット 6 **EW**：警告ステータス

0：両方のエラー･カウンタが 96 の Error_Warning 制限未満です。

1：少なくとも 1 つのエラー･カウンタが 96 の Error_Warning 制限に達しました。

ビット 5 **EP**：エラー･パッシブ

0：FDCAN は、Error_Active 状態です。通常、バス通信に参加して、エラーを検出した場合にアクティブ･エラー･フラグを送信します。

1：FDCAN は、Error_Passive 状態です。

ビット 4:3 **ACT[1:0]**：動作

モジュールの CAN 通信状態を監視します。

00：同期中：ノードが CAN 通信で同期しています。

01：アイドル：ノードはレシーバでもトランスミッタでもありません。

10：レシーバ：ノードはレシーバとして動作しています。

11：トランスミッタ：ノードはトランスミッタとして動作しています。

ビット 2:0 **LEC[2:0]**：最終エラーコード

LEC は、CAN バスで発生した最後のエラーのタイプを示します。このフィールドは、メッセージがエラーなしで転送（受信または送信）されたときに 0 にクリアされます。

000：エラーなし：正常な受信または送信によって LEC がリセットされて以降に発生したエラーはありません。

001：スタッフ･エラー：シーケンス上で許容されていない6ビット以上の同一レベルの連続ビットが受信メッセージに含まれている

010：フォーム･エラー：受信フレームの固定フォーマット部分に誤ったフォーマットがあります。

011：Ack(確認応答)エラー：FDCAN によって送信されたメッセージが他のノードによって確認応答されませんでした。

100：ビット 1 エラー：メッセージ（アービトレーションフィールドを除く）送信中、デバイスがリセッシブレベル（論理値のビット 1）の送信を求めたが、監視されたバスの値がドミナントでした。

101：ビット 0 エラー：メッセージ（または確認応答ビット、アクティブエラーフラグ、オーバーロードフラグ）送信中、デバイスがドミナントレベル（データまたは識別子ビットの論理値 0）の送信を求めたが、監視されたバスの値がリセッシブでした。Bus_Off リカバリ中、11 のリセッシブビットのシーケンスが監視されるたびに、このステータスがセットされます。これにより、CPU が（バスがドミナントでスタックしていない、または継続的に妨害されていないことを示す）Bus_Off リカバリシーケンスの進行を監視できます。

110：CRC エラー：受信メッセージの CRC チェックサムが正しくありません。着信メッセージの CRC が、受信したデータから計算された CRC と一致しません。

111：変更なし：プロトコルステータスレジスタの読出しアクセスによって、LEC が 7 に再初期化されます。LEC が値 '7' を示している場合、プロトコルステータスレジスタの最後の CPU 読出しアクセス以降に CAN バスイベントは検出されませんでした。

アクセスタイプは RS で、読出し時にセットされます。

注： **FDCAN フォーマットのフレームが、BRS フラグがセットされたデータ･フェーズに達したとき、次の CAN イベント（エラーまたは有効なフレーム）は LEC の代わりに FLEC で示されます。FDCAN CRC シーケンスの固定スタッフ･ビットでのエラーは、スタッフ･エラーではなく、フォーム･エラーとして示されます。**

注： **Bus_Off リカバリ･シーケンス（CAN Specification Rev. 2.0 または ISO11898-1参照）は、CCCR[INIT] のセットやリセットで短縮できません。デバイスが Bus_Off になると、それ自体で CCCR.INIT をセットし、すべてのバス動作を停止します。CCCR[INIT] が CPU によってクリアされると、デバイスは通常動作を再開する前に、129 のバス･アイドルの発生（129 × 11 の連続リセッシブ･ビット）を待ちます。Bus_Off リカバリ･シーケンスの最後に、エラー管理カウンタがリセットされます。CCCR[INIT] リセット後の待機時間中、11 のリセッシブ･ビットで構成されたシーケンスを監視するたびに、ビット 0 のエラー･コードが PSR[LEC] に書き込まれます。これにより、CPU から、ドミナントへの固定や継続的な中断が CAN バスに発生していないかを容易に確認し、Bus_Off リカバリ･シーケンスを監視できます。ECR[REC] はこれらのシーケンスをカウントするために使用されます。**

44.4.14 FDCAN トランスミッタ遅延補正レジスタ（FDCAN_TDCR）

アドレスオフセット：0x0048

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:8 **TDCO[6:0]**：トランスミッタ遅延補正オフセット

FDCAN_TX から FDCAN_RX までに測定された遅延とセカンダリサンプルポイントの距離を定義するオフセット値です。有効な値は 0～127 mtq です。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:0 **TDCF[6:0]**：トランスミッタ遅延補正フィルタのウィンドウ長

SSP の位置の最小値を定義し、より早い SSP の位置につながる FDCAN_RX のドミナントエッジは、トランスミッタ遅延測定では無視されます。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 に設定されているときのみ、このビットによる書込みアクセスが可能です。

44.4.15 FDCAN 割込みレジスタ（FDCAN_IR）

リストされた条件が検出された場合、このフラグがセットされます（エッジ区別）。フラグはホストがクリアするまでセットされたままとなります。フラグは、対応するビット位置に 1 を書き込むことによってクリアされます。

0 を書き込んでも、ビットの値は変化しません。ハード･リセットでレジスタはクリアされます。IE の設定は、割込みが生成されるかどうかを制御します。ILS の設定は、割込みが信号送信される割込みラインを制御します。

アドレスオフセット：0x0050

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO
								rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOO	MRAF	TSW	TEFL	TEFF	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1N	RF0L	RF0F	RF0N
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ARA**：予約済みアドレスへのアクセス

0：予約済みアドレスへのアクセスは発生していません。

1：予約済みアドレスへのアクセスが発生しました。

ビット 22 **PED**：データ･フェーズでのプロトコル･エラー（データ･ビット時間が使用されます）

0：データフェーズでのプロトコルエラーはありませんでした。

1：データフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.DLEC）。

ビット 21 **PEA**：アービトレーション･フェーズでのプロトコル･エラー（公称ビット時間が使用されます）

0：アービトレーションフェーズでのプロトコルエラーはありませんでした。

1：アービトレーションフェーズでのプロトコルエラーが検出されました（0、7 ではない PSR.LEC）。

ビット 20 **WDI**：ウォッチドッグ割込み

0：メッセージ RAM ウォッチドッグイベントは発生していません。

1：不足した READY によるメッセージ RAM ウォッチドッグイベントがありました。

ビット 19 **BO**：Bus_Off ステータス

0：Bus_Off ステータスは変更されません。

1：Bus_Off ステータスは変更されます。

ビット 18 **EW**：警告ステータス

0：Error_Warning ステータスは変更されません。

1：Error_Warning ステータスは変更されます。

ビット 17 **EP**：エラー･パッシブ

0：Error_Passive ステータスは変更されません。

1：Error_Passive ステータスは変更されます。

ビット 16 **ELO**：エラー･ロギング･オーバーフロー

0：CAN エラー･ロギング･カウンタはオーバーフローしていません。

1：CAN エラー･ロギング･カウンタのオーバーフローが発生しました。

ビット 15 **TOO**：タイムアウト発生

0：タイムアウトなし

1：タイムアウト到達

ビット 14 **MRAF**：メッセージ RAM アクセス失敗

このフラグは、Rx ハンドラが次の場合にセットされます。

- 次のメッセージのアービトレーション･フィールドを受信するまでに、許可フィルタまたは許可メッセージの格納を完了しなかった場合。この場合、許可フィルタまたは許可メッセージの格納はアボートされ、Rx ハンドラが次のメッセージの処理を開始します。
- メッセージをメッセージ RAM に書き込めなかった場合。この場合、メッセージ格納はアボートされます。

どちらの場合も、FIFO Put インデックスは更新されません。部分的に格納されたメッセージは、次のメッセージがこの位置に格納される際に上書きされます。

Tx ハンドラがメッセージ RAM から時間内にメッセージを読み出せなかったときもこのフラグがセットされます。この場合、メッセージ送信はアボートされます。Tx ハンドラ･アクセス失敗時、FDCAN は制限動作モードに切り替わります（制限動作モードを参照）。制限動作モードを終了するには、ホスト CPU で CCCR.ASM をリセットする必要があります。

0：メッセージ RAM アクセス失敗は発生していません。

1：メッセージ RAM アクセス失敗が発生しました。

ビット 13 **TSW**：タイムスタンプ･ラップアラウンド

0：タイムスタンプ･カウンタのラップアラウンドはありません。

1：タイムスタンプ･カウンタのラップアラウンドが発生しました。

ビット 12 **TEFL**：Tx イベント FIFO 要素の喪失
0：Tx イベント FIFO 要素の喪失はありません。
1：Tx イベント FIFO 要素の喪失が発生しました。

ビット 11 **TEFF**：Tx イベント FIFO フル
0：Tx イベント FIFO はフルではありません。
1：Tx イベント FIFO はフルです。

ビット 10 **TEFN**：Tx イベント FIFO 新規エントリ
0：Tx イベント FIFO は変更されません。
1：Tx ハンドラが Tx イベント FIFO 要素に書き込みました。

ビット 9 **TFE**：Tx FIFO エンプティ
0：Tx FIFO はエンプティではありません。
1：Tx FIFO はエンプティです。

ビット 8 **TCF**：送信キャンセル終了
0：終了した送信キャンセルはありません。
1：送信キャンセルが終了しました。

ビット 7 **TC：**送信完了
0：完了した送信はありません。
1：送信が完了しました。

ビット 6 **HPM**：高優先メッセージ
0：受信した高優先メッセージはありません。
1：高優先メッセージを受信しました。

ビット 5 **RF1L**：Rx FIFO 1 メッセージ喪失
0：喪失した Rx FIFO 1 メッセージはありません。
1：Rx FIFO 1 メッセージを喪失しました。

ビット 4 **RF1F**：Rx FIFO 1 フル
0：Rx FIFO 1 はフルではありません。
1：Rx FIFO 1 はフルです。

ビット 3 **RF1N**：Rx FIFO 1新規メッセージ
0：Rx FIFO 1 に書き込まれた新規メッセージはありません。
1：新規メッセージが Rx FIFO 1 に書き込まれました。

ビット 2 **RF0L**：Rx FIFO 0 メッセージ喪失
0：喪失した Rx FIFO 0 メッセージはありません。
1：Rx FIFO 0 メッセージを喪失しました。

ビット 1 **RF0F**：Rx FIFO 0 フル
0：Rx FIFO 0 はフルではありません。
1：Rx FIFO 0 はフルです。

ビット 0 **RF0N**：Rx FIFO 0新規メッセージ
0：Rx FIFO 0 に書き込まれた新規メッセージはありません。
1：新規メッセージが Rx FIFO 0 に書き込まれました。

44.4.16 FDCAN 割込み有効レジスタ（FDCAN_IE）

割込み有効レジスタの設定によって、割込みレジスタのどのステータス変化が割込みラインで通知されるかが決まります。

アドレス･オフセット：0x0054

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1NE	RF0LE	RF0FE	RF0NE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **ARAE**：予約済みアドレスへのアクセス･イネーブル

ビット 22 **PEDE**：データ･フェーズでのプロトコル･エラー･イネーブル

ビット 21 **PEAE**：アービトレーション･フェーズでのプロトコル･エラー･イネーブル

ビット 20 **WDIE**：ウォッチドッグの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 19 **BOE**：Bus_Off ステータス

0：割込みは無効です。

1：割込みは有効です。

ビット 18 **EWE**：警告ステータスの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 17 **EPE**：エラー･パッシブの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 16 **ELOE**：エラー･ロギング･オーバーフローの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 15 **TOOE**：タイムアウト発生の割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 14 **MRAFE**：メッセージ RAM アクセス失敗の割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 13 **TSWE**：タイムスタンプ･ラップアラウンドの割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 12 **TEFLE**：Tx イベント FIFO 要素の喪失の割込みイネーブル

0：割込みは無効です。

1：割込みは有効です。

ビット 11 **TEFFE**：Tx イベント FIFO フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 10 **TEFNE**：Tx イベント FIFO 新規エントリの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 9 **TFEE**：Tx FIFO エンプティの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 8 **TCFE**：送信キャンセル終了の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 7 **TCE**：転送完了の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 6 **HPME**：高優先メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 5 **RF1LE**：Rx FIFO 1メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 4 **RF1FE**：Rx FIFO 1フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 3 **RF1NE**：Rx FIFO 1新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 2 **RF0LE**：Rx FIFO 0メッセージ喪失の割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 1 **RF0FE**：Rx FIFO 0フルの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

ビット 0 **RF0NE**：Rx FIFO 0新規メッセージの割込みイネーブル
0：割込みは無効です。
1：割込みは有効です。

44.4.17 FDCAN 割込みライン選択レジスタ（FDCAN_ILS）

このレジスタは、割込みレジスタの割込みフラグの特定のグループによって生成される割込みを、2つのモジュール割込みラインのうちの 1 つに割り当てます。割込み生成には、それぞれの割込みラインを ILE[EINT0] および ILE[EINT1] で有効にする必要があります。

アドレスオフセット：0x0058

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERR	BERR	MISC	TFERR	SMSG	RXFIFO1	RXFIFO0
									rw	rw	rw	rw	rw	rw	rw

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **PERR**：プロトコル･エラーのグループは以下の割込み
- ARAL：予約済みアドレスへのアクセスのライン
- PEDL：データ･フェーズでのプロトコル･エラーのライン
- PEAL：アービトレーション･フェーズでのプロトコル･エラーのライン
- WDIL：ウォッチドッグの割込みライン
- BOL：Bus_Off ステータス
- EWL：警告ステータスの割込みライン

ビット 5 **BERR**：ビットおよびラインのエラーのグループは以下の割込み
- EPL：エラー･パッシブの割込みライン
- ELOL：エラー･ロギング･オーバーフローの割込みライン

ビット 4 **MISC**：割込みの再グループは以下の割込み
- TOOL：タイムアウト発生の割込みライン
- MRAFL：メッセージ RAM アクセス失敗の割込みライン
- TSWL：タイムスタンプ･ラップアラウンドの割込みライン

ビット 3 **TFERR**：Tx FIFO エラーのグループは以下の割込み
- TEFLL：Tx イベント FIFO 要素の喪失の割込みライン
- TEFFL：Tx イベント FIFO フルの割込みライン
- TEFNL：Tx イベント FIFO 新規エントリの割込みライン
- TFEL：Tx FIFO エンプティの割込みライン

ビット 2 **SMSG**：ステータス･メッセージ･ビットのグループは以下の割込み
- TCFL：送信キャンセル終了の割込みライン
- TCL：送信完了の割込みライン
- HPML：高優先メッセージの割込みライン

ビット 1 **RXFIFO1**：RX FIFO ビットのグループは以下の割込み
- RF1LL：Rx FIFO 1 メッセージ喪失の割込みライン
- RF1FL：Rx FIFO 1 フルの割込みライン
- RF1NL：Rx FIFO 1 新規メッセージの割込みライン

ビット 0 **RXFIFO0**：RX FIFO ビットは以下の割込み
- RF0LL：Rx FIFO 0 メッセージ喪失の割込みライン
- RF0FL：Rx FIFO 0 フルの割込みライン
- RF0NL：Rx FIFO 0 新規メッセージの割込みライン

44.4.18 FDCAN 割込みライン有効レジスタ（FDCAN_ILE）

CPU への 2 つの割込みラインは、それぞれビット EINT0 および EINT1 をプログラムすることで有効化／無効化できます。

アドレスオフセット：0x005C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **EINT1**：割込みライン 1 イネーブル
0：割込みライン fdcan_intr0_it は無効です。
1：割込みライン fdcan_intr0_it は有効です。

ビット 0 **EINT0**：割込みライン 0 イネーブル
0：割込みライン fdcan_intr1_it は無効です。
1：割込みライン fdcan_intr1_it は有効です。

44.4.19 FDCAN グローバルフィルタ設定レジスタ（FDCAN_RXGFC）

メッセージ ID フィルタのグローバル設定です。グローバルフィルタ設定は、図 670 および図 671 に示されたとおり、標準メッセージおよび拡張メッセージのフィルタパスを制御します。

アドレスオフセット：0x0080

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	LSE[3:0]				Res.	Res.	Res.	LSS[4:0]				
				rw	rw	rw	rw				rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F0OM	F1OM	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
						rw	rw			rw	rw	rw	rw	rw	rw

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **LSE[3:0]**：リストサイズ拡張
0：拡張メッセージ ID フィルタはありません。
1 から 8：拡張メッセージ ID フィルタ要素の数です。
>8：8 を超える値は 8 として解釈されます。
これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 23:21 予約済みであり、リセット値に保持する必要があります。

ビット 20:16 **LSS[4:0]**：リストサイズ標準

0：標準メッセージ ID フィルタはありません。

1 から 28：標準メッセージ ID フィルタ要素の数です。

>28：28 を超える値は 28 として解釈されます。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 **F0OM**：FIFO 0 動作モード（上書きまたはブロック）

これは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 8 **F1OM**：FIFO 1 動作モード（上書きまたはブロック）

これは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **ANFS[1:0]**：一致しないフレーム（標準）の許可

フィルタ・リストの要素のいずれとも一致しない 11 ビット ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 3:2 **ANFE[1:0]**：一致しないフレーム（拡張）の許可

フィルタ・リストの要素のいずれとも一致しない 29 ビット ID を持つ受信メッセージの処理方法を定義します。

00：Rx FIFO 0 で許可

01：Rx FIFO 1 で許可

10：拒否

11：拒否

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 1 **RRFS**：リモート・フレーム（標準）の拒否

0：11 ビット標準 ID を持つリモート・フレームをフィルタします。

1：11 ビット標準 ID を持つすべてのリモート・フレームを拒否します。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

ビット 0 **RRFE**：リモート・フレーム（拡張）の拒否

0：29 ビット標準 ID を持つリモート・フレームをフィルタします。

1：29 ビット標準 ID を持つすべてのリモート・フレームを拒否します。

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

44.4.20 FDCAN 拡張 ID およびマスクレジスタ（FDCAN_XIDAM）

アドレスオフセット：0x0084

リセット値：0x1FFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	EIDM[28:16]												
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIDM[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:29 予約済みであり、リセット値に保持する必要があります。

ビット 28:0 **EIDM[28:0]**：拡張 ID マスク

拡張フレームの許可フィルタでは､拡張 ID AND マスクと受信フレームのメッセージ ID の論理積（AND）が計算されます。SAE J1939 での 29 ビット ID のマスキングを対象とします。すべてのビットのリセット値を 1 にすると、マスクはアクティブではなくなります。

これらは書込み保護（P）ビットで､CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

44.4.21 FDCAN 高優先メッセージ･ステータス･レジスタ（FDCAN_HPMS）

このレジスタは、メッセージ ID フィルタ要素が優先度イベント一致を生成するように設定するたびに更新されます。これを使用して、着信高優先メッセージのステータスを監視し、これらのメッセージへの高速アクセスが可能になります。

アドレスオフセット：0x0088

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLST	Res.	Res.	FIDX[4:0]					MSI[1:0]		Res.	Res.	Res.	BIDX[2:0]		
r			r	r	r	r	r	r	r				r	r	r

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **FLST**：フィルタ･リスト

一致フィルタ要素のフィルタリストを示します。

0：標準フィルタリスト

1：拡張フィルタリスト

ビット 14:13 予約済みであり、リセット値に保持する必要があります。

ビット 12:8 **FIDX[4:0]**：フィルタインデックス

一致フィルタ要素のインデックスです。範囲は、0～RXGFC[LSS] - 1 または RXGFC[LSE] - 1 です。

ビット 7:6 **MSI[1:0]**：メッセージ･ストレージ･インジケータ
00：選択された FIFO がありません。
01：FIFO オーバーランです。
10：メッセージが FIFO 0 に格納されます。
11：メッセージが FIFO 1 に格納されます。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **BIDX[2:0]**：バッファインデックス
メッセージが格納された Rx FIFO 要素のインデックスです。MSI[1] = 1 の場合のみ有効です。

44.4.22 FDCAN Rx FIFO 0 ステータスレジスタ（FDCAN_RXF0S）

アドレス･オフセット：0x0090

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	Res.	Res.	Res.	Res.	F0PI [1:0]	
						r	r							r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F0GI [1:0]		Res.	Res.	Res.	Res.	F0FL [3:0]			
						r	r					r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **RF0L**：Rx FIFO 0 メッセージ喪失
このビットは割込みレジスタ IR[RF0L] のコピーです。IR[RF0L] がリセットされると、このビットもリセットされます。
0：喪失した Rx FIFO 0 メッセージはありません。
1:Rx FIFO 0 メッセージを喪失しました。サイズ 0 の Rx FIFO 0 への書込み試行後にもセットされます。

ビット 24 **F0F**：Rx FIFO 0 フル
0：Rx FIFO 0 はフルではありません。
1：Rx FIFO 0 はフルです。

ビット 23:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **F0PI[1:0]**：Rx FIFO 0 プット インデックス
Rx FIFO 0 書込みインデックスポインタで、範囲は 0〜2 です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **F0GI[1:0]**：Rx FIFO 0 取得インデックス
Rx FIFO 0 読出しインデックスポインタで、範囲は 0〜2 です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **F0FL[3:0]**：Rx FIFO 0 フィルレベル
Rx FIFO 0 に格納される要素の数で、範囲は 0 ~ 3 です。

44.4.23 CAN Rx FIFO 0 確認応答レジスタ（FDCAN_RXF0A）

アドレス･オフセット：0x0094

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **F0AI[2:0]**：Rx FIFO 0確認応答インデックス

ホストが Rx FIFO 0 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 0 から読み出された最後の要素のバッファインデックスを F0AI に書き込む必要があります。これにより、Rx FIFO 0 Get インデックス RXF0S[F0GI] が F0AI + 1 に設定され、FIFO 0 フィル･レベル RXF0S[F0FL] が更新されます。

44.4.24 FDCAN Rx FIFO 1 ステータス･レジスタ（FDCAN_RXF1S）

アドレス･オフセット：0x0098

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.	Res.	Res.	Res.	F1PI [1:0]	
						r	r							r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	F1GI [1:0]		Res.	Res.	Res.	Res.	F1FL [3:0]			
						r	r					r	r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **RF1L**：Rx FIFO 1 メッセージ喪失

このビットは割込みレジスタ IR[RF1L] のコピーです。IR[RF1L] がリセットされると、このビットもリセットされます。

0：喪失した Rx FIFO 1 メッセージはありません。

1:Rx FIFO 1 メッセージを喪失しました。サイズ 0 の Rx FIFO 1 への書込み試行後にもセットされます。

ビット 24 **F1F**：Rx FIFO 1 フル

0：Rx FIFO 1 はフルではありません。

1：Rx FIFO 1 はフルです。

ビット 23:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **F1PI[1:0]**：Rx FIFO 1 プット インデックス

Rx FIFO 1 書込みインデックスポインタで、範囲は 0～2 です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **F1GI[1:0]**：Rx FIFO 1 取得インデックス
Rx FIFO 1 読出しインデックスポインタで、範囲は 0〜2 です。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **F1FL[3:0]**：Rx FIFO 1 フィルレベル
Rx FIFO 1 に格納される要素の数で、範囲は 0 ~ 3 です。

44.4.25 FDCAN Rx FIFO 1 確認応答レジスタ（FDCAN_RXF1A）

アドレス･オフセット：0x009C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **F1AI[2:0]**：Rx FIFO 1確認応答インデックス
ホストが Rx FIFO 1 からメッセージまたは一連のメッセージを読み出した後、Rx FIFO 1 から読み出された最後の要素のバッファインデックスを F1AI に書き込む必要があります。これにより、Rx FIFO 1 Get インデックス RXF1S[F1GI] が F1AI + 1 に設定され、FIFO 1 フィル･レベル RXF1S[F1FL] が更新されます。

44.4.26 FDCAN Tx バッファ設定レジスタ（FDCAN_TXBC）

アドレスオフセット：0x00C0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
							rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24 **TFQM**：Tx FIFO／キュー･モード
0：Tx FIFO 動作
1：Tx キュー動作
これは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 に設定されているときのみ、このビットによる書込みアクセスが可能です。

ビット 23:0 予約済みであり、リセット値に保持する必要があります。

44.4.27 FDCAN Tx FIFO／キュー･ステータス･レジスタ（FDCAN_TXFQS）

Tx FIFO／キューステータスは、レジスタ TXBRP に示されたペンディング中の Tx リクエストに関連します。そのため、追加／キャンセルリクエストの効果は、実行中の Tx スキャン（TXBRP がまだ更新されていない）によって遅延する場合があります。

アドレスオフセット：0x00C4

リセット値：0x0000 0003

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	Res.	Res.	Res.	TFQPI[1:0]	
										r				r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TFGI[1:0]		Res.	Res.	Res.	Res.	Res.	TFFL[2:0]		
						r	r						r	r	r

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **TFQF**：Tx FIFO／キュー･フル
0：Tx FIFO／キューはフルではありません。
1：Tx FIFO／キューがフルです。

ビット 20:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **TFQPI[1:0]**：Tx FIFO／キュー プット インデックス
Tx FIFO／キュー書込みインデックスポインタで、範囲は 0〜3 です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **TFGI[1:0]**：Tx FIFO Get インデックス
Tx FIFO 読出しインデックス･ポインタで、範囲は 0 ~ 3 です。Tx キュー動作が設定された場合（TXBC.TFQM = 1）、0 が読み出されます。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TFFL[2:0]**：Tx FIFO フリーレベル
連続するフリー Tx FIFO 要素の数は TFGI から開始し、範囲は 0〜3 です。Tx キュー動作が設定された場合（TXBC[TFQM] = 1）、0 が読み出されます。

44.4.28 FDCAN Tx バッファ･リクエスト･ペンディング･レジスタ（FDCAN_TXBRP）

アドレスオフセット：0x00C8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRP[2:0]		
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TRP[2:0]**：送信リクエスト･ペンディング

各 Tx バッファには独自の送信リクエスト･ペンディング･ビットがあります。このビットはレジスタ TXBAR 経由でセットされます。リクエストされた送信が完了した後か、レジスタ TXBCR 経由でキャンセルされた後に、ビットがリセットされます。

TXBRP ビットがセットされた後、Tx スキャンが開始され、最高優先度を持つペンディング中の Tx リクエスト（一番小さいメッセージ ID の Tx バッファ）がないかチェックされます。

キャンセル･リクエストで、レジスタ TXBRP の対応する送信リクエスト･ペンディング･ビットがリセットされます。キャンセルがリクエストされたときに送信をすでに開始していた場合、送信が成功したかどうかに関係なく、送信の最後に実行されます。対応する TXBRP ビットがリセットされた直後に、キャンセルリクエストビットがリセットされます。

キャンセルがリクエストされた後、終了したキャンセルは次のタイミングで TXBCF 経由で信号送信されます。

対応する TXBTO ビットとともに送信が成功した後

キャンセル時点で送信がまだ開始されていなかったとき

ロストアービトレーションによって送信がアボートされたとき

フレーム送信中にエラーが発生したとき

DAR モードでは、すべての送信が成功しなかった場合、自動的にキャンセルされます。すべての成功しなかった送信に対応する TXBCF ビットがセットされます。

0：送信リクエストペンディングはありません。

1：送信リクエストペンディングがあります。

注： **Tx スキャンが進行中のときにセットされた TXBRP ビットは、この特定の Tx スキャン中は考慮されません。このような Tx バッファに対してキャンセルがリクエストされた場合、この追加リクエストはすぐにキャンセルされ、対応する TXBRP ビットはリセットされます。**

44.4.29 FDCAN Tx バッファ追加リクエスト･レジスタ（FDCAN_TXBAR）

アドレスオフセット：0x00CC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR[2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **AR[2:0]**：追加リクエスト

各 Tx バッファには独自の追加リクエスト･ビットがあります。1 を書き込むと、対応する追加リクエスト･ビットがセットされ、0 を書き込んでも影響はありません。これにより、ホストは複数の Tx バッファに対する送信リクエストを TXBAR への 1 回の書込みで設定できます。実行中の Tx スキャンがないときはビットがすぐにリセットされ、それ以外の場合は Tx スキャン･プロセスが完了するまでビットがセットされたままとなります。

0：追加された送信リクエストはありません。

1：送信リクエストが追加されました。

注： ペンディング中の送信リクエストを持つ（対応する TXBRPビットがすでにセットされている）Tx バッファに追加リクエストを適用しても、そのリクエストは無視されます。

44.4.30 FDCAN Tx バッファ･キャンセル･リクエスト･レジスタ（FDCAN_TXBCR）

アドレスオフセット：0x00D0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CR[2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CR[2:0]**：キャンセルリクエスト

各 Tx バッファには独自のキャンセルリクエストビットがあります。1 を書き込むと、対応する CR ビットがセットされ、0 を書き込んでも影響はありません。

これにより、ホストは複数の Tx バッファに対するキャンセル･リクエストを TXBCR への 1 回の書込みで設定できます。対応する TXBRP ビットがリセットされるまで、ビットはセットされたままとなります。

0：ペンディング中のキャンセルはありません。

1：キャンセルがペンディングされています。

44.4.31 FDCAN Tx バッファ送信発生レジスタ（FDCAN_TXBTO）

アドレスオフセット：0x00D4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TO[2:0]		
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TO[2:0]**：送信発生

各 Tx バッファには独自の TO ビットがあります。送信成功後に対応する TXBRP ビットがクリアされたとき、このビットがセットされます。レジスタ TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：発生した送信はありません。

1：送信が発生しました。

44.4.32 FDCAN Tx バッファキャンセル終了レジスタ（FDCAN_TXBCF）

アドレスオフセット：0x00D8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CF[2:0]		
													r	r	r

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CF[2:0]**：キャンセル終了

各 Tx バッファには独自の CF ビットがあります。キャンセルが TXBCR ビットによってリクエストされた後に、対応する TXBRP ビットがクリアされたとき、このビットがセットされます。対応する TXBRP ビットがキャンセル時点でセットされなかった場合、CF がすぐにセットされます。レジスタ TXBAR の対応するビットに 1 を書き込むことによって新しい送信がリクエストされると、ビットがリセットされます。

0：送信バッファキャンセルはありません。

1：送信バッファキャンセルが終了しました。

44.4.33 FDCAN Tx バッファ送信割込み有効レジスタ（FDCAN_TXBTIE）

アドレスオフセット：0x00DC

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE[2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **TIE[2:0]**：送信割込みは有効です。
各 Tx バッファには独自の TIE ビットがあります。
0：送信割込みは無効です。
1：送信割込みは有効です。

44.4.34 FDCAN Tx バッファ･キャンセル終了割込みイネーブル･レジスタ（FDCAN_TXBCIE）

アドレスオフセット：0x00E0

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFIE[2:0]		
													rw	rw	rw

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **CFIE[2:0]**：キャンセル終了の割込みイネーブル
各 Tx バッファには独自の CFIE ビットがあります。
0：キャンセル終了割込みは無効です。
1：キャンセル終了割込みは有効です。

44.4.35 FDCAN Tx イベント FIFO ステータスレジスタ（FDCAN_TXEFS）

アドレスオフセット：0x00E4

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	Res.	Res.	Res.	EFPI[1:0]	
						r	r							r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	EFGI[1:0]		Res.	Res.	Res.	Res.	Res.	EFFL[2:0]		
						r	r						r	r	r

ビット 31:26 予約済みであり、リセット値に保持する必要があります。

ビット 25 **TEFL**：Tx イベント FIFO 要素の喪失

このビットは割込みフラグ IR[TEFL] のコピーです。IR[TEFL] がリセットされると、このビットもリセットされます。

0：Tx イベント FIFO 要素の喪失はありません。

1：Tx イベント FIFO 要素を喪失しました。サイズ 0 の Tx イベント FIFO への書込み試行後にもセットされます。

ビット 24 **EFF**：イベント FIFO フル

0：Tx イベント FIFO はフルではありません。

1：Tx イベント FIFO はフルです。

ビット 23:18 予約済みであり、リセット値に保持する必要があります。

ビット 17:16 **EFPI[1:0]**：イベント FIFO Put インデックス

Tx イベント FIFO 書込みインデックス･ポインタで、範囲は 0 ~ 3 です。

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **EFGI[1:0]**：イベント FIFO Get インデックス

Tx イベント FIFO 読出しインデックス･ポインタで、範囲は 0 ~ 3 です。

ビット 7:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **EFFL[2:0]**：イベント FIFO フィル･レベル

Tx イベント FIFO に格納されている要素の数で、範囲は 0 ~ 3 です。

44.4.36 FDCAN Tx イベント FIFO 確認応答レジスタ（FDCAN_TXEFA）

アドレスオフセット：0x00E8

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[1:0]	
														rw	rw

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **EFAI[1:0]**：イベント FIFO 確認応答インデックス

Tx イベント FIFO から要素または一連の要素を読み出したホストは、Tx イベント FIFO から読み出した最後の要素のインデックスを EFAI に書き込む必要があります。これにより、Tx イベント FIFO Get インデックス TXEFS[EFGI] が EFAI + 1 に設定され、FIFO 0 フィル･レベル TXEFS[EFFL] が更新されます。

44.4.37 FDCAN CFG クロック分周回路レジスタ（FDCAN_CKDIV）

アドレスオフセット：0x0100

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PDIV[3:0]			
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:0 **PDIV[3:0]**：入力クロック分周器

APB クロックは、CAN サブシステムで使用する前に分周できます。分周比は、分周回路出力クロックを使用して計算する必要があります。

0000：分周比 1
0001：分周比 2
0010：分周比 4
0011：分周比 6
0100：分周比 8
0101：分周比 10
0110：分周比 12
0111：分周比 14
1000：分周比 16
1001：分周比 18
1010：分周比 20
1011：分周比 22
1100：分周比 24
1101：分周比 26
1110：分周比 28
1111：分周比 30

これらは書込み保護（P）ビットで、CCCR レジスタのビット 1 [CCE] およびビット 0 [INIT] が 1 にセットされているときのみ、このビットによる書込みアクセスが可能です。

44.4.38 FDCAN レジスタ･マップ

表 410. FDCAN レジスタのマップとリセット値(1)

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000	FDCAN_CREL	REL[3:0]				STEP[3:0]				SUBSTEP [3:0]				YEAR[3:0]				MON[7:0]								DAY[7:0]							
	リセット値	0	0	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	1	1	0	1	1	1	1	1	0	1	0	0	0	1	0
0x0004	FDCAN_ENDN	ETV[31:0]																															
	リセット値	1	0	0	0	0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0	0	0	1	1	0	0	1	0	0	0	0	1
0x0008	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x000C	FDCAN_DBTP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	TDC	Res.	Res.	DBRP[4:0]					Res.	Res.	Res.	DTSEG1 [4:0]					DTSEG2 [3:0]				DSJW[3:0]			
	リセット値									0	0	0	0	0	0	0	0				0	1	0	1	0	0	0	1	1	0	0	1	1
0x0010	FDCAN_TEST	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RX	TX [1:0]		LBCK	Res.	Res.	Res.	Res.
	リセット値																									0	0	0	0				
0x0014	FDCAN_RWD	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WDV[7:0]								WDC[7:0]							
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0018	FDCAN_CCCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	NISO	TXP	EFBI	PXHD	Res.	Res.	BRSE	FDOE	TEST	DAR	MON	CSR	CSA	ASM	CCE	INT
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x001C	FDCAN_NBTP	NSJW[6:0]								NBRP[8:0]								NTSEG1 [7:0]								Res.	NTSEG2 [6:0]						
	リセット値	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	1
0x0020	FDCAN_TSCC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TCP[3:0]				Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSS [1:0]	
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0024	FDCAN_TSCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TSC[15:0]															
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0028	FDCAN_TOCC	TOP[15:0]																Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOS [1:0]		ETOC
	リセット値	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x002C	FDCAN_TOCV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TOC[15:0]															
	リセット値																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0030 から 0x003C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0040	FDCAN_ECR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CEL[7:0]								RP	REC[6:0]							TEC[7:0]							
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0044	FDCAN_PSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCV[6:0]							Res.	PXE	REDL	RBRSRESI1	RESI	DLEC[2:0]			BO	EW	EP	ACT[1:0]		LEC[2:0]		
	リセット値										0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1

表 410. FDCAN レジスタのマップとリセット値(1)（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0048	FDCAN_TDCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TDCO[6:0]							Res.	TDCF[6:0]						
	リセット値																		0	0	0	0	0	0		0	0	0	0	0	0	0	0
0x004C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0050	FDCAN_IR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARA	PED	PEA	WDI	BO	EW	EP	ELO	TOO	MRAF	TSW	TEFL	TEFF	TEFN	TFE	TCF	TC	HPM	RF1L	RF1F	RF1N	RF0L	RF0F	RF0N
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0054	FDCAN_IE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	ARAE	PEDE	PEAE	WDIE	BOE	EWE	EPE	ELOE	TOOE	MRAFE	TSWE	TEFLE	TEFFE	TEFNE	TFEE	TCFE	TCE	HPME	RF1LE	RF1FE	RF1NE	RF0LE	RF0FE	RF0NE
	リセット値									0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0058	FDCAN_ILS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PERR	BERR	MISC	TFERR	SMSG	RXFIFO1	RXFIFO0
	リセット値	0																									0	0	0	0	0	0	0
0x005C	FDCAN_ILE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EINT1	EINT0
	リセット値																															0	0
0x0060 から 0x007C	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																
0x0080	FDCAN _RXGFC	Res.	Res.	Res.	Res.	LSE[3:0]				Res.	Res.	Res.	LSS[4:0]					Res.	Res.	Res.	Res.	Res.	Res.	F0OM	F1OM	Res.	Res.	ANFS[1:0]		ANFE[1:0]		RRFS	RRFE
	リセット値					0	0	0	0				0	0	0	0	0							0	0			0	0	0	0	0	0
0x0084	FDCAN _XIDAM	Res.	Res.	Res.	EIDM[28:0]																												
	リセット値				1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x0088	FDCAN_HPMS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FLST	Res.	Res.	FIDX[4:0]					MSI [1:0]		Res.	Res.	Res.	BIDX [2:0]		
	リセット値																	0			0	0	0	0	0	0	0				0	0	0
0x0090	FDCAN _RXF0S	Res.	Res.	Res.	Res.	Res.	Res.	RF0L	F0F	Res.	Res.	Res.	Res.	Res.	Res.	F0PI [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	F0GI [1:0]		Res.	Res.	Res.	Res.	F0FL [3:0]			
	リセット値							0	0							0	0							0	0						0	0	0
0x0094	FDCAN _RXF0A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F0AI [2:0]		
	リセット値																														0	0	0
0x0098	FDCAN _RXF1S	Res.	Res.	Res.	Res.	Res.	Res.	RF1L	F1F	Res.	Res.	Res.	Res.	Res.	Res.	F1PI [1:0]		Res.	Res.	Res.	Res.	Res.	Res.	F1GI [1:0]		Res.	Res.	Res.	Res.	F1FL [3:0]			
	リセット値							0	0							0	0							0	0						0	0	0
0x009C	FDCAN _RXF1A	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	F1AI [2:0]		
	リセット値																														0	0	0
0x00A0 から 0x00BC	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値																																

表 410. FDCAN レジスタのマップとリセット値(1)（続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00C0	FDCAN_TXBC	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQM	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値								0	0																							
0x00C4	FDCAN_TXFQS	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TFQF	Res.	Res.	Res.	TFQPI[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	TFGI[1:0]		Res.	Res.	Res.	Res.	Res.	TFFL[2:0]		
	リセット値											0				0	0							0	0						0	1	1
0x00C8	FDCAN_TXBRP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRP2	TRP1	TRP0
	リセット値																														0	0	0
0x00CC	FDCAN_TXBAR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	AR2	AR1	AR0
	リセット値																														0	0	0
0x00D0	FDCAN_TXBCR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CR2	CR1	CR0
	リセット値																														0	0	0
0x00D4	FDCAN_TXBTO	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TO2	TO1	TO0
	リセット値																														0	0	0
0x00D8	FDCAN_TXBCF	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CF2	CF1	CF0
	リセット値																														0	0	0
0x00DC	FDCAN_TXBTIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TIE2	TIE1	TIE0
	リセット値																														0	0	0
0x00E0	FDCAN_TXBCIE	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CFIE2	CFIE1	CFIE0
	リセット値																														0	0	0
0x00E4	FDCAN_TXEFS	Res.	Res.	Res.	Res.	Res.	Res.	TEFL	EFF	Res.	Res.	Res.	Res.	Res.	Res.	EFPI[1:0]		Res.	Res.	Res.	Res.	Res.	Res.	EFG[1:0]		Res.	Res.	Res.	Res.	Res.	EFFL[2:0]		
	リセット値							0	0							0	0							0	0						0	0	0
0x00E8	FDCAN_TXEFA	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EFAI[1:0]	
	リセット値																															0	0
0x0100	FDCAN_CKDIV	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PDIV[3:0]			
	リセット値																													0	0	0	0

1. R = 読出し、S = 読出し時にセット、X = 読出し時にリセット、W = 書込み、P = 書込み保護、p = セット保護、C = 書込み時にクリア／プリセット

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

45 ユニバーサル･シリアル･バス･フルスピード･デバイス･インタフェース（USB）

45.1 概要

USB ペリフェラルは、フルスピードの USB 2.0 バスと APB1 バスの間のインタフェースを実装します。

USB サスペンド／レジュームがサポートされるため､節電のためにデバイスのクロックを停止できます。

45.2 USB の主な機能

- USB 仕様バージョン 2.0 フルスピード準拠
- 1 から 8 まで設定可能なエンドポイント数
- 1024 バイトの専用パケットバッファメモリ（SRAM）
- 巡回冗長検査（CRC）の生成／チェック、NRZI（Non-return-to-zero Inverted）方式のエンコード／デコード、およびビットスタッフィング
- アイソクロナス転送サポート
- ダブルバッファバルク／アイソクロナスエンドポイントサポート
- USB サスペンド／レジューム操作
- フレームロッククロックパルス生成
- USB 2.0 リンク電源管理サポート
- バッテリ充電仕様 Revision 1.2 サポート
- USB 接続／切断機能（USB DP ライン上の制御可能な組み込みプルアップレジスタ）

45.3 USB の実装

表 411 に、デバイスでの USB 実装を示します。

表 411. STM32G4 シリーズ の USB の実装

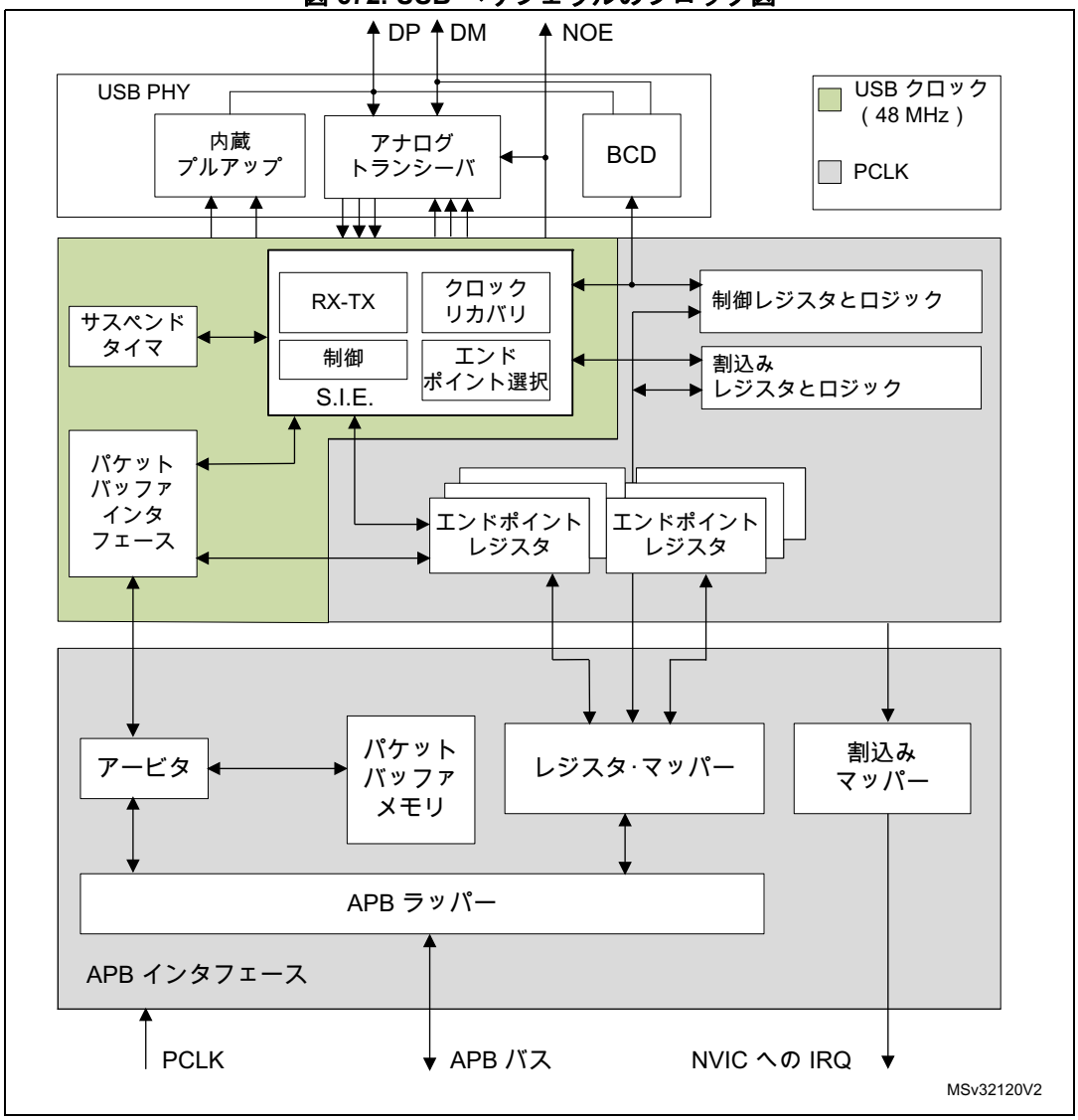
USB の機能(1)	USB
エンドポイント数	8
専用パケットバッファメモリ SRAM のサイズ	1024 バイト
専用パケットバッファメモリ SRAM のアクセス方式	2 x 16 ビット/ワード
USB 2.0 リンク電源管理（LPM）のサポート	X
バッテリ充電検出（BCD）のサポート	X
USB_DP ライン上の組み込みプルアップレジスタ	X

1. X = サポート

45.4 USB の機能詳細

図 672 に、USB ペリフェラルのブロック図を示します。

図 672. USB ペリフェラルのブロック図



USB ペリフェラルは、ホスト PC とマイクロコントローラによって実装される機能の間に USB 準拠接続を提供します。ホストPC とシステムメモリ間のデータ転送は、USB ペリフェラルによって直接アクセスされる専用パケットバッファメモリを通じて行われます。この専用メモリのサイズは1024 バイトであり、最大 16 の単方向または 8 つの双方向エンドポイントを使用できます。USB ペリフェラルは USB ホストとインタフェースして、トークンパケットを検出し、データの送受信を処理し、USB 標準の要求に従ってハンドシェイクパケットを処理します。トランザクションのフォーマット化は、CRC の生成とチェックも含めて、ハードウェアによって行われます。

各エンドポイントにはバッファ記述ブロックが関連付けられ、そこにはエンドポイント関連のメモリ領域の位置、サイズ、送信する必要があるバイト数などが示されています。機能とエンドポイントの有効なペアのトークンが USB ペリフェラルで検出されると、関連するデータが転送されます（データの転送が必要で、エンドポイントが設定されている場合）。USB ペリフェラルによってバッファされたデータは、内部 16 ビットレジスタに配置され、専用バッファへのメモリアクセスが行われます。

すべてのデータが転送されると、必要な場合には、転送方向に従って、適切なハンドシェイクパケットが USB 経由で生成され、または転送の方向によって期待されます。

トランザクションの終了時には、エンドポイント固有の割込みが生成され、ステータスレジスタを読み出すか、別の割込み応答ルーチンを使用します。マイクロコントローラは、以下を決めることができます。

- 処理するエンドポイント
- エラーが発生した場合に行うトランザクションのタイプ（ビットスタッフィング、フォーマット、CRC、プロトコル、ACK の欠如、オーバーラン／アンダーランなど）

アイソクロナス転送と高スループットバルク転送に対する特別なサポートが提供され、ダブルバッファ私用が実装されることにより、USB ペリフェラルが使用できるバッファが常にあり、マイクロコントローラ用のバッファが別にあります。

必要なときには、制御レジスタに書き込むことによって、ユニットを低電力モード（SUSPEND モード）にできます。このとき、すべての静的消費電流が回避され、USB クロックをスローダウンまたは停止できます。低電力モード中に USB 入力でアクティビティが検出されると、デバイスは非同期でウェイクアップします。特殊な割込みソースをウェイクアップラインに直接接続することにより、システムは通常のクロック生成をただちに再開し、直接クロックの開始／停止をサポートできます。

45.4.1 USB ブロックの説明

USB ペリフェラルは USB インタフェースに関するすべての機能を実装し、これには以下のブロックが含まれます。

- USB 物理インタフェース（USB PHY）; このブロックは、外部 USB ホストへの電気インタフェースを維持します。ディファレンシャルアナログトランシーバ本体、制御可能な組み込みプルアップレジスタ（USB_DP ラインに接続）、およびバッテリ充電検出（BCD）のサポート、同じ USB_DP および USB_DM ラインでの多重化を含みます。アナログトランシーバの出力有効制御信号（アクティブロー）は、USB_NOE により外部から供給されます。これを使用して、アクティビティ LED を駆動したり、実際の通信方向に関する情報を他の回路に提供することができます。
- シリアルインタフェースエンジン（SIE）: このブロックの機能には、同期パターン認識、ビットスタッフィング、CRC 生成およびチェック、PID 確認／生成、およびハンドシェイク評価が含まれます。USB トランシーバとインタフェースし、ローカルデータストレージ用のパケットバッファインタフェースによって提供される仮想バッファを使用する必要があります。このユニットは、フレーム開始（SOF）、USB_Reset、データエラーなどの USB ペリフェラルイベントと、送信終了、パケットの正しい受信などのエンドポイント関連イベントに従って信号を生成し、これらの信号は割込みの生成に使用されます。
- タイマ : このブロックは、フレーム開始ロッククロックパルスを生成し、トラフィックが 3 ms の間、受信されなかったときにはグローバルサスペンドを（ホストから）検出します。
- パケットバッファインタフェース : このブロックは、ローカルメモリを管理して、送信と受信の両方のためのバッファのセットを柔軟に実装します。SIE から着信したリクエストに従って適切なバッファを選び、エンドポイントレジスタによって示されるメモリアドレスに配置します。パケットの終わりまで、各バイトの交換後にアドレスをインクリメントして、交換バイト数を追跡し、バッファが最大容量までオーバーランするのを防ぎます。
- エンドポイント関連レジスタ : 各エンドポイントには、エンドポイントのタイプと現在のステータスを含んだレジスタが関連付けられます。単方向／シングルバッファエンドポイントの場合、1 つのレジスタで 2 つのエンドポイントを実装できます。レジスタの数は 8 つであり、最大 16 の単方向／シングルバッファまたは最大 7 つのダブルバッファエンドポイントを任意の組み合わせで使用できます。たとえば、USB ペリフェラルをプログラムして、4 つのダブルバッファエンドポイントと 8 つのシングルバッファ／単方向エンドポイントを使用できます。

- 制御レジスタ：これらは USB ペリフェラル全体のステータスに関する情報を含むレジスタであり、レジュームやパワーダウンなどの USB イベントを強制するために使用されます。
- 割込みレジスタ：これらは割込みマスクとイベントのレコードを含みます。割込み理由、割込みステータスの問い合わせや、保留中の割込みのステータスをクリアするために使用できます。

注： *** エンドポイント 0 は、常にシングルバッファモードでの転送の制御に使用されます。**

USB ペリフェラルは APB1 バスに APB1 インタフェースを通じて接続され、以下のブロックを含みます。

- パケットメモリ：これは、パケットバッファを物理的に含むローカルメモリです。パケットバッファインタフェースによって使用して、データ構造を作成し、アプリケーションソフトウェアから直接アクセスできます。パケットメモリのサイズは、1024 バイトであり、16 ビットの 512 個のハーフワードで構成されます。
- アービタ：このブロックは、APB1 バスおよび USB インタフェースから着信するリクエストを受け入れます。APB1 アクセスに優先順位を付けることによってコンフリクトを解決するとともに、常にメモリ帯域幅の半分をすべての USB 転送を完了するために確保しています。このタイムデュプレックス方式によって仮想デュアルポート SRAM が実装され、USB トランザクション中のメモリアクセスを可能にします。この方式により、任意の長さのマルチワード APB1 転送も可能です。
- レジスタマッパー：このブロックは、USB ペリフェラルのさまざまなバイト長およびビット長のレジスタをAPB1によってアドレス指定され構造化された 16 ビット長のハーフワードセットに集めます。
- APB1 ラッパー：これは、メモリとレジスタに APB1 へのインタフェースを提供します。また、USB ペリフェラル全体を APB1 アドレス空間にマップします。
- 割込みマッパー：このブロックは、USB イベントによって割込みを生成する方法を選択し、それらを NVIC にマップするために使用されます。

45.5 プログラミングに関する考慮事項

以下のセクションでは、アプリケーションソフトウェア開発を容易にするために、USB ペリフェラルとアプリケーションプログラムの間の予期される相互作用について説明します。

45.5.1 汎用 USB デバイスのプログラミング

このパートでは、USB 準拠動作を得るためにアプリケーションソフトウェアに必要な主なタスクについて説明します。ほとんどの汎用 USB イベントに関連するアクションが考慮に入れられており、ダブルバッファエンドポイントとアイソクロナス転送という特殊ケースについて、それぞれのパラグラフで説明します。システムリセットを別にすると、アクションは常に USB ペリフェラルによって開始され、以下に述べる USB イベントの 1 つによって駆動されます。

45.5.2 システムリセットとパワーオンリセット

システムリセットおよびパワーオンリセットの際、アプリケーションソフトウェアが最初に実行すべき操作は、必要なクロック信号のすべてを USB ペリフェラルに与えることであり、その後、リセット信号をネゲートして、レジスタにアクセスできるようにしなければなりません。初期化シーケンス全体は、以下のとおりです。

最初のステップとして、アプリケーションソフトウェアは、デバイスクロック管理ロジックによって提供される関連の制御ビットを使用して、レジスタマクロセルクロックを有効にし、マクロセル固有のリセット信号をネゲートする必要があります。

その後、CNTR レジスタの PDWN ビットを使用して、USB トランシーバに関連するデバイスのアナログ部分をオンにする必要があり、これには特別な処理が必要です。このビットは、ポートトランシーバに供給される内部電圧基準をオンにすることを目的としています。この回路にはスタートアップ時間が定義されており（データシートで指定されている $t_{STARTUP}$）、この間の USB トランシーバの動作は定義されていません。したがって、CNTR レジスタの PDWN ビットをセットした後、この時間だけ待つ必要があり、その後、USB 部分のリセット条件を解除します（CNTR レジスタの FRES ビットをクリアすることによって）。ISTR レジスタをクリアすると、他のマクロセル動作が有効になる前に、誤って保留中の割込みが削除されます。

システムリセット時には、マイクロコントローラは必要なすべてのレジスタとパケットバッファ記述テーブルを初期化して､USB ペリフェラルが割込みとデータ転送を正しく生成できるようにする必要があります。エンドポイント固有ではないレジスタはすべて、アプリケーションソフトウェアの必要に応じて初期化する必要があります（有効にする割込みの選択、パケットバッファのアドレスの選択など）。USB リセットの場合、その後、プロセスが続行します（以下のパラグラフを参照）。

USB リセット（RESET 割込み）

このイベントが発生すると､USB ペリフェラルは､前のパラグラフで説明した初期化後のシステムリセットと同じ条件になります。すなわち、すべてのエンドポイントレジスタで通信が無効になります（USB ペリフェラルはパケットに応答しません）。USB リセットイベントに対して、USB 機能を有効にして、USB アドレスを 0 にし、デフォルトのコントロールエンドポイントのみを実装します（エンドポイントアドレスも 0）。これは、USB_DADDR レジスタの機能有効（EF）ビットをセットし、EP0R レジスタと関連するパケットバッファを初期化することによって行われます。USB エニュメレーションプロセス時、ホストは一意なアドレスをこのデバイスに割り当て、これが USB_DADDR レジスタの ADD[6:0] ビットに書き込まれなければならず、他に必要なエンドポイントを設定する必要があります。

RESET 割込みが受信されると､アプリケーションソフトウェアは､割込みをトリガしたリセットシーケンスの終了から 10 ms 以内に、USB 機能 0 のデフォルトのエンドポイントを再び有効にする必要があります。

パケットバッファの構造と用途

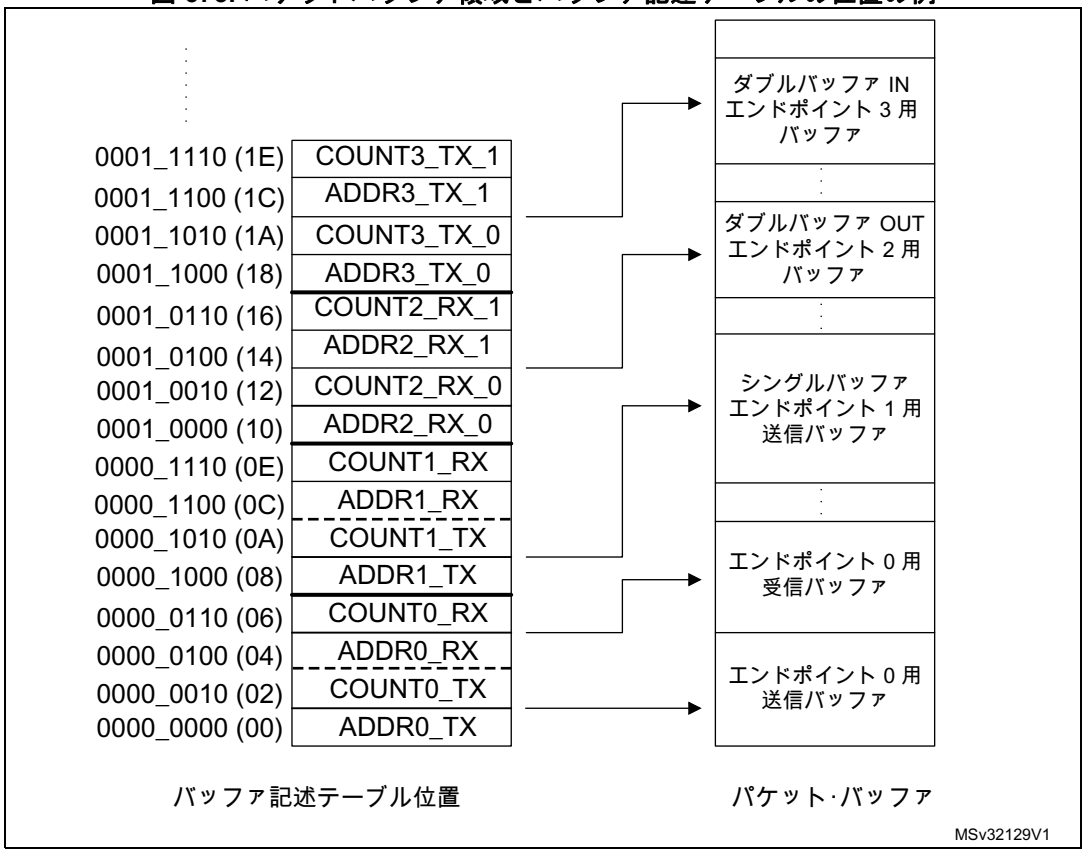
各双方向エンドポイントは、ホストとの間でデータを送受信できます。受信データは、そのエンドポイント用に予約された専用のメモリバッファに格納され、別のメモリバッファにはエンドポイントによって送信されるデータが格納されます。このメモリへのアクセスは、パケットバッファインタフェースブロックによって行われ､これはメモリアクセスリクエストを配信し､確認応答を待ちます。マイクロコントローラもパケットバッファメモリにアクセスする必要があるので、アクセスコンフリクトを避けるためにアービトレーションロジックが使用され、APB1 サイクルの半分がマイクロコントローラのアクセスに使用され、残りの半分が USB ペリフェラルのアクセスに使用されます。このように、パケットメモリがデュアルポート SRAM であるかのように両方のエージェントが動作でき、マイクロコントローラが連続アクセスを実行中でも、コンフリクトを気にする必要がありません。USB ペリフェラルロジックは、専用クロックを使用します。この専用クロックの周波数は、USB 標準の要件により 48 MHz に固定されており、これは APB1 バスへのインタフェースに使用されるクロックと異なる場合があります。APB1 クロック周波数が USB ペリフェラルのクロック周波数より高い、または低い場合、さまざまなクロック設定が可能です。

注： **USB アクティブバス接続中の USB スループットとシステム性能の理由から、48 MHz 以上の APB1 クロックを動作させることをお勧めします。**

各エンドポイントには 2 つのパケットバッファが関連付けられます（通常は、送信用に 1 つ、受信用に 1 つ）。バッファの位置とサイズはバッファ記述テーブルで指定されるので、バッファはパケットメモリ内の任意の場所に配置でき、このテーブルも USB_BTABLE レジスタによって示されるパケットメモリ内のアドレスに配置されます。各テーブルエントリがエンドポイントレジスタに関連付けられ、4 つの 16 ビットハードワードで構成されるため、テーブル開始アドレスは常に 8 バイトの境界

に揃えられます（USB_BTABLE レジスタの下位 3 ビットは常に 000）。バッファディスクリプタテーブルのエントリについては、セクション 45.6.2 : バッファディスクリプタテーブルで説明します。エンドポイントが単方向であり、アイソクロナスでもダブルバッファバルクでもない場合、必要なパケットバッファは 1 つだけです（サポートされる転送方向に応じた 1 つのバッファ）。サポートされない転送方向または未使用のエンドポイントに関連するテーブル位置は、ユーザが使用できます。アイソクロナスおよびダブルバッファバルクエンドポイントには、パケットバッファの特別な取り扱いがあります（それぞれ、セクション 45.5.4 : アイソクロナス転送およびセクション 45.5.3 : ダブルバッファエンドポイントを参照）。バッファ記述テーブルのエントリとパケットバッファ領域の関係を図 673 に示します。

図 673. パケットバッファ領域とバッファ記述テーブルの位置の例



各パケットバッファは、始めから終わりまで、受信または送信のいずれかに使用されます。USB ペリフェラルは、割り当てられたメモリバッファに隣接するメモリ位置の内容を変えることはありません。割り当てられたバッファ長を超えるパケットが受信された場合（バッファオーバーラン条件）、使用可能な最期の位置までしか、データはメモリにコピーされません。

エンドポイントの初期化

エンドポイントを初期化するための最初のステップは、ADDRn_TX/ADDRn_RX レジスタに適切な値を書き込んで、送信データがすでに使用可能になったことと、受信データをバッファできることを USB ペリフェラルに知らせることです。USB_EPnR レジスタの EP_TYPE ビットをエンドポイントのタイプに応じてセットし、最終的には EP_KIND ビットを使用して、特に必要な機能を有効にする必要があります。送信側では、USB_EPnR レジスタの STAT_TX ビットを使用してエンドポイントを有効にする必要があり、COUNTn_TX を初期化する必要があります。受信の場合、STAT_RX ビットをセットして受信を有効にする必要があり、BL_SIZE および NUM_BLOCK フィールドを使用して、割り当てられたバッファサイズを COUNTn_RX に書き込む必要があります。単方向エンドポイント

は、アイソクロナスおよびダブルバッファバルクエンドポイントを除き、サポートされる方向に関するビットとレジスタのみを初期化する必要があります。送信または受信が有効になると、レジスタ USB_EPnR と位置 ADDRn_TX/ADDRn_RX、COUNTn_TX/COUNTn_RX（それぞれ）をアプリケーションソフトウェアで変更しないでください。これらの値はハードウェアが直接変更できます。データ転送操作が完了し、CTR 割込みイベントによって通知されると、再びアクセスできるようになり、新しい操作を再び有効にできます。

IN パケット（データ送信）

IN トークンパケットを受信するとき、受信したアドレスが設定済みの有効なエンドポイントと一致した場合、USB ペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内の ADDRn_TX および COUNTn_TX 位置にアクセスします。これらの位置の内容は、内部 16 ビットレジスタ ADDR および COUNT に格納されます（ソフトウェアからはアクセスできません）。パケットメモリが再びアクセスされ、最初の送信バイトを読出し（1977 ページのパケットバッファの構造と用途を参照）、USB_EPnR のビット DTOG_TX に従って、DATA0 または DATA1 PID の送信を開始します。PID が完了すると、バッファメモリから読み出された最初のバイトが出力シフトレジスタにロードされ、USB バスで送信されます。最後のデータバイトが送信された後、計算された CRC が送信されます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_EPnR レジスタの STAT_TX ビットに従って、NAK または STALL ハンドシェイクパケットがデータパケットの代わりに送信されます。

ADDR 内部レジスタは、現在のバッファメモリ位置を指すポインタとして使用され、COUNT は、残りの送信バイト数をカウントするために使用されます。パケットバッファメモリから読み出された各ハーフワードは、最下位バイトから順に USB バスで送信されます。送信バッファメモリは、COUNTn_TX/2 ハーフワードの場合、ADDRn_TX によって示されるアドレスから順に読み出されます。送信パケットが奇数個のバイトで構成される場合、最後にアクセスされたハーフワードの下位半分のみが使用されます。

ホストが ACK を受信した場合、USB_EPnR レジスタは次のように更新されます。DTOG_TX ビットがトグルされ、STAT_TX=10（NAK）をセットすることによってエンドポイントが無効化され、ビット CTR_TX がセットされます。アプリケーションソフトウェアは、まず、USB_ISTR レジスタの EP_ID および DIR ビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。CTR_TX イベントの処理は、割込みビットをクリアすることから始まります。次に、アプリケーションソフトウェアは、送信データでいっぱいの別のバッファを準備し、COUNTn_TX テーブル位置を次の転送時に送信されるバイト数で更新し、最後に、STAT_TX を 11（VALID）にセットして、送信を再び有効にします。STAT_TX ビットが 10（NAK）に等しいときには、そのエンドポイントにアドレス指定された IN リクエストは NAK され、フロー制御条件を示します。USB ホストは、成功するまで、トランザクションを再試行します。CTR 割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された 2 番目の IN トランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。

OUT および SETUP パケット（データ受信）

この 2 つのトークンは、USB ペリフェラルによっておおむね同じように扱われます。SETUP パケットの取り扱いにおける違いについては、制御転送に関する以下のパラグラフで詳しく述べます。OUT/SETUP PID を受信するとき、アドレスが有効なエンドポイントと一致した場合、USB ペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内の ADDRn_RX および COUNTn_RX 位置の内容にアクセスします。ADDRn_RX の内容は、内部レジスタ ADDR に直接格納されます。COUNT がリセットされ、BL_SIZE および NUM_BLOCK ビットフィールドの値が COUNTn_RX 内に読み出されると、それを使用して BUF_COUNT が初期化され、これは、バッファオーバーラン条件のチェックに使用される内部 16 ビットカウンタです（これらの内部レジスタはすべて、ソフトウェアからはアクセスできません）。その後、USB ペリフェラルによって受信されるデータバイトは、ハーフワード単位でパケット化され（最初の受信バイトが最下位バイトとして格納されます）、内部 ADDR レジスタに含まれるアドレスから順にパケットバッファに転送

され、各バイト転送により、BUF_COUNT はデクリメントされ、COUNT はインクリメントされます。DATA パケットの終わりが検出されると、受信された CRC の正しさがテストされ、受信時にエラーが発生しなかった場合のみ、ACK ハンドシェイクパケットが送信元ホストに送り返されます。

CRC が正しくなかった場合、または他の種類のエラー（ビットスタッフ違反、フレームエラーなど）が発生した場合、データバイトは、少なくともエラー検出ポイントまではパケットメモリにコピーされますが、ACK パケットは送信されず、USB_ISTR レジスタの ERR ビットがセットされます。ただし、この場合、通常はソフトウェアのアクションは不要です。USB ペリフェラルは受信エラーから回復し、次のトランザクションを処理する準備ができた状態を保ちます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_EPnR レジスタの STAT_RX ビットに従って、NAK または STALL ハンドシェイクパケットがACKの代わりに送信され、データは受信メモリバッファに書き込まれません。

受信メモリバッファの位置は、ADDRn_RX に含まれるアドレスから、CRC を含めた受信データパケット長に対応するバイト数（データペイロード長 + 2）だけ、または、BL_SIZE および NUM_BLOCK によって定義された割り当て済みの最後のメモリ位置のいずれか先に来る方まで書き込まれます。このように、USB ペリフェラルは、割り当てられた受信メモリバッファ領域の終わりを超えて書き込むことはありません。データパケットペイロードの長さ（アプリケーションによって使用された実際のバイト数）が割り当てられたバッファより大きい場合、USB ペリフェラルはバッファオーバーラン条件を検出します。この場合、通常の ACK の代わりに STALL ハンドシェイクが送信されて、問題をホストに通知し、割込みは生成されず、トランザクションは失敗したとみなされます。

ACK ハンドシェイクパケットを送信することにより、トランザクションが正常に完了すると、内部 COUNT レジスタがバッファ記述テーブルエントリ内の COUNTn_RX 位置にコピーされ、影響を受けなかった BL_SIZE および NUM_BLOCK フィールドはそのままに保たれ（通常は再書込みの必要はありません）、USB_EPnR レジスタが次のように更新されます。DTOG_RX ビットがトグルされ、STAT_RX=10（NAK）をセットすることによってエンドポイントが無効化され、ビット CTR_RX がセットされます。エラーまたはバッファオーバーランによってトランザクションが失敗した場合、上記のアクションはいずれも実行されません。アプリケーションソフトウェアは、まず、USB_ISTR レジスタの EP_ID および DIR ビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。CTR_RX イベントの処理は、まず、トランザクションのタイプを決定することによって行われます（USB_EPnR レジスタの SETUP ビット）。アプリケーションソフトウェアは、割込みフラグビットをクリアして、処理されるエンドポイントに関連するバッファ記述テーブルエントリ内の COUNTn_RX 位置を読み出して、受信バイト数を取得する必要があります。受信データが処理された後、アプリケーションは USB_EPnR の STAT_RX ビットを 11（有効）にセットして、次のトランザクションを有効にする必要があります。STAT_RX ビットが 10（NAK）に等しいときには、そのエンドポイントにアドレス指定された OUT リクエストは NAK され、フロー制御条件を示します。USB ホストは、成功するまで、トランザクションを再試行します。CTR 割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された 2 番目の OUT トランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。

制御転送

制御転送は、SETUP トランザクション、すべて同じ方向のゼロ個以上のデータステージ、およびステータスステージ（反対方向のゼロバイトの転送）で構成されます。SETUP トランザクションはコントロールエンドポイントのみによって処理され、OUT トランザクション（データ受信）と非常に似ていますが、制御転送を初期化するために、アドレス指定されたエンドポイントレジスタの DTOG_TX および DTOG_RX ビットの値がそれぞれ 1 と 0 にセットされる点と、ソフトウェアが後続のトランザクションが IN か OUT かを SETUP の内容に応じて決定できるように、STAT_TX と STAT_RX の両方が 10（NAK）にセットされる点が異なります。制御エンドポイントは CTR_RX イベントごとに USB_EPnR レジスタの SETUP ビットをチェックして、通常の OUT トランザクションと SETUP トランザクションを区別する必要があります。USB デバイスは、SETUP ステージで転送されたデータを解釈することによって、データステージの数と方向を判断でき、エラーの場合にはト

ランザクションを STALL する必要があります。そのためには、最後以外のすべてのデータステージで、未使用の方向を STALL にセットして、ホストが転送方向を反転するのが速過ぎた場合にはステータスステージとして STALL されるようにする必要があります。

最後のデータステージを有効にするときには、反対方向を NAK にセットして、ホストが（ステータスステージを実行するために）転送方向をすぐに反転した場合、制御操作の完了を待つようにする必要があります。制御操作が正常に完了した場合、ソフトウェアは NAK を VALID に変更し、そうでない場合は STALL に変更します。同時に、ステータスステージが OUT の場合、STATUS_OUT（USB_EPnR レジスタの EP_KIND）ビットをセットして、ステータストランザクションが非ゼロのデータで実行された場合はエラーが生成されるようにする必要があります。ステータストランザクションが処理されると、アプリケーションは STATUS_OUT ビットをクリアし、STAT_RX を VALID にセットし（新しいコマンドを受け入れるため）、STAT_TX を NAK にセットします（次のセットアップの直後のステータスステージを遅延するため）。

USB 仕様では、SETUP パケットには ACK 以外のハンドシェイクでは応答できず、結果として、以前に発行されたコマンドをアボートして新しいコマンドを開始すると規定されているため、USB ロジックでは、コントロールエンドポイントはホストから受信した SETUP トークンに対して NAK または STALL パケットで応答することはできません。

STAT_RX ビットが 01（STALL）または 10（NAK）にセットされ、SETUP トークンが受信されると、USB はデータを受け入れ、要求されたデータ転送を実行し、ACK ハンドシェイクを返送します。そのエンドポイントに以前に発行された CTR_RX リクエストがあり、アプリケーションによってまだ確認応答されていない場合（すなわち、CTR_RX ビットが以前に完了した受信からセットされたままになっている場合）、USB は SETUP トランザクションを破棄し、状態にかかわらずハンドシェイクパケットで応答せず、受信エラーをシミュレートして、ホストが SETUP トークンを再び送信しなければならないようにします。これは、CTR_RX 割込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された SETUP トランザクションの通知が失われるのを避けるために行われます。

45.5.3 ダブルバッファエンドポイント

USB 標準によって定義されているすべてのエンドポイントタイプは、さまざまなトラフィックモデルを表し、各種のデータ転送操作の典型的な要件を記述しています。データの大部分をホスト PC と USB 機能の間で転送する必要があるときには、バルクエンドポイントタイプが最適のモデルです。ホストはバルクトランザクションをスケジュールして、フレーム内の使用可能な帯域幅のすべてをいっぱいにして、アドレス指定された USB 機能がバルクトランザクションを処理できる限り、実際の転送速度を最大化します。次のトランザクションが着信したときに、USB 機能が前のトランザクションでビジーであった場合は、NAK ハンドシェイクで応答し、ホスト PC は、USB 機能が処理できるようになるまで、同じトランザクションを再び発行するので、再送信によって帯域幅が占有されるため、実際の転送速度は低下します。このため、バルクエンドポイントでは、「ダブルバッファリング」と呼ばれる専用機能を使用できます。

「ダブルバッファリング」が有効なときには、データトグルシーケンシングが使用されて、USB ペリフェラルが要求されたデータ転送を実行するために使用するバッファが選択され、送信と受信の両方のパケットメモリ領域を使用して、トランザクションの成功ごとのバッファスワッピングを管理するため、アプリケーションは常に完全なバッファを使用でき、USB ペリフェラルは他方のバッファをいっぱいにします。たとえば、「受信」ダブルバッファバルクエンドポイント宛ての OUT トランザクション時、1 つのバッファが USB ホストから着信する新しいデータで満たされている間、もう 1 つのバッファはマイクロコントローラソフトウェアが使用できます（「送信」ダブルバッファバルクエンドポイントと IN トランザクションでも同様です）。

スワップバッファ管理には、アドレスポインタと割り当て済みメモリバッファの長さをホストする 4 つすべてのバッファ記述テーブル位置を使用する必要があるため、ダブルバッファバルクエンドポイントの実装に使用される USB_EPnR レジスタは、強制的に単方向として使用されます。したがって、1 つの STAT ビットのみを 00（無効）以外の値にセットする必要があります。ダブルバッファバルク

エンドポイントが受信について有効な場合は STAT_RX、ダブルバッファバルクエンドポイントが送信について有効な場合は STAT_TX。ダブルバッファバルクエンドポイントを受信と送信の両方について有効にする必要がある場合は、2 つの USB_EPnR レジスタを使用する必要があります。

ダブルバッファリング機能を利用して、可能な限り高い転送速度を達成するには、前の章で説明したエンドポイントフロー制御構造を変更して、トランザクションの正常官僚の旅ではなく、USB ペリフェラルとアプリケーションの間でバッファコンフリクトが発生したときのみ、エンドポイントのステータスを NAK に切り替える必要があります。USB ペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関連する DTOG ビットによって定義されます。すなわち、「受信」ダブルバッファバルクエンドポイントの場合は DTOG_RX（USB_EPnR レジスタのビット 14）、または「送信」ダブルバッファバルクエンドポイントの場合は DTOG_TX（USB_EPnR レジスタのビット 6）です。新しいフロー制御方式を実装するには、USB ペリフェラルはアプリケーションソフトウェアによって使用されているパケットバッファを知り、コンフリクトに注意する必要があります。USB_EPnR レジスタには 2 つの DTOG ビットがありますが、データおよびバッファシーケンシングのために USB ペリフェラルによって使用されるのは 1 つだけであり（ダブルバッファリング機能に必要な単方向成約のため）、もう 1 つはアプリケーションソフトウェアが、使用中のバッファを示すために使用できます。この新しいバッファフラグは、SW_BUF と呼ばれます。次の表で、「送信」および「受信」ダブルバッファバルクエンドポイントの場合の USB_EPnR レジスタのビットと DTOG/SW_BUF 定義の対応を説明します。

表 412. ダブルバッファリングバッファフラグの定義

バッファフラグ	「送信」エンドポイント	「受信」エンドポイント
DTOG	DTOG_TX（USB_EPnR ビット 6）	DTOG_RX（USB_EPnR ビット 14）
SW_BUF	USB_EPnR ビット 14	USB_EPnR ビット 6

USB ペリフェラルによって現在使用されているメモリバッファは、DTOG バッファフラグによって定義され、アプリケーションソフトウェアによって現在使用されているバッファは、SW_BUF バッファフラグによって示されます。バッファフラグの値と使用されているパケットバッファの関係は、どちらの場合も同じであり、次の表に示されています。

表 413. バルクダブルバッファリングメモリバッファの用途

エンドポイントタイプ	DTOG	SW_BUF	使用されるパケットバッファ（USB ペリフェラル）	アプリケーションソフトウェアによって使用されるパケットバッファ
入力	0	1	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。	ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置。
	1	0	ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。
	0	0	なし (1)	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。
	1	1	なし (1)	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。

表 413. バルクダブルバッファリングメモリバッファの用途 （続き）

エンドポイントタイプ	DTOG	SW_BUF	使用されるパケットバッファ（USB ペリフェラル）	アプリケーションソフトウェアによって使用されるパケットバッファ
OUT	0	1	ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。	ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。
	1	0	ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。	ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。
	0	0	なし (1)	ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。
	1	1	なし (1)	ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。

1. NAK ステータスのエンドポイント。

バルクエンドポイントのダブルバッファリング機能は、次のようにして有効化されます。

- USB_EPnR レジスタの EP_TYPE ビットフィールドに 00 を書き込み、エンドポイントをバルクとして定義します。
- 同じレジスタの EP_KIND ビットを 1（DBL_BUF）にセットします。

アプリケーションソフトウェアは､最初に使用されるバッファに応じて､DTOG および SW_BUF ビットを初期化する責任があります。これは、この 2 つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。DBL_BUF のセット後に発生した初めてのトランザクションの終了により、ダブルバッファバルクエンドポイントの特殊なフロー制御がトリガされ、DBL_BUF がセットされたままである限り、このエンドポイントにアドレス指定された他のすべてのトランザクションについて、これが使用されます。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイント USB_EPnR レジスタの CTR_RX または CTR_TX ビットがセットされます。同時に､USB_EPnR レジスタの影響を受けた DTOG ビットがハードウェアによってトグルされ、USB ペリフェラルのバッファスワッピングを完全にソフトウェアから独立させます。通常のトランザクション、および DBL_BUF のセット後の初めてのトランザクションと違って、STAT ビットペアはトランザクション終了の影響を受けず、値は 11（有効）のままです。ただし、新しいトランザクションのトークンパケットが受信されると､実際のエンドポイントステータスは､USB ペリフェラルとアプリケーションソフトウェアの間のバッファコンフリクトが検出されたときには 10（NAK）としてマスクされます（この条件は DTOG と SW_BUF が同じ値を持つことによって示されます。1982 ページの表 413 を参照）。アプリケーションソフトウェアは、CTR イベント通知に応答して、割込みフラグをクリアし、完了したトランザクションの要求された処理を開始します。アプリケーションパケットバッファの仕様が終了すると、ソフトウェアは SW_BUF ビットに 1 を書き込んでトグルし、そのバッファが使用可能になったことを USB ペリフェラルに通知します。このように、NAK されるトランザクションの数は、トランザクションデータのアプリケーションエラボレーション時間によってのみ制限されます。エラボレーション時間が USB バスでのトランザクションの完了に必要な時間より短かった場合、フロー制御による再送信は行われず、実際の転送速度はホスト PC によってのみ制限されます。

アプリケーションソフトウェアは､11（有効）とは異なる明示的なステータスを､関連する USB_EPnR レジスタの STAT ビットペアに書き込むことによって、ダブルバッファバルクエンドポイントのために実装される特殊なフロー制御をオーバーライドできます。この場合、USB ペリフェラルは、バッファ使用条件に関係なく、プログラムされたエンドポイントステータスを常に使用します。

45.5.4 アイソクロナス転送

USB 標準は、固定された正確なデータ生成／消費周波数を必要とするフルスピードペリフェラルをサポートし、この種類のトラフィックを「アイソクロナス」として定義しています。このデータの典型的な例は、オーディオサンプル、圧縮ビデオストリーム、および一般に、供給される周波数の正確さについて厳密な要件を持つ、あらゆる種類のサンプルデータです。エニュメレーションフェーズでエンドポイントが「アイソクロナス」として定義されると、ホストは必要な帯域幅をフレームに割り当て、エンドポイントの方向に応じて、各フレームに正確に 1 つの IN または OUT パケットを供給します。帯域幅要件を制限するために、アイソクロナストラフィックについては、失敗したトランザクションの再送信はできません。このため、アイソクロナストランザクションにはハンドシェイクフェーズはなく、ACK パケットは予期されず、データパケット後に送信されません。同じ理由で、アイソクロナス転送はデータトグルシーケンシングをサポートせず、常に DATA0 PID を使用してデータパケットを開始します。

エンドポイントのアイソクロナス動作は、USB_EPnR レジスタの EP_TYPE ビットを 10 にセットすることによって選択されます。ハンドシェイクフェーズがないため、STAT_RX/STAT_TX ビットの唯一の正当な値は 00（無効）と 11（有効）であり、他の値の場合は、USB 標準に準拠しない結果になります。アイソクロナスエンドポイントはダブルバッファリングを実装して、アプリケーションソフトウェア開発を容易にし、「送信」と「受信」の両方のパケットメモリ領域を使用して、トランザクションの成功のたびにバッファスワッピングを管理し、アプリケーションは常に完全なバッファを使用でき、USB ペリフェラルはもう 1 つのバッファを満たします。

USB ペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関する DTOG ビットによって、表 414 に従って定義されます（「受信」アイソクロナスエンドポイントの場合は DTOG_RX、「送信」アイソクロナスエンドポイントの場合は DTOG_TXであり、両方とも、関連する USB_EPnR レジスタ）。

表 414. アイソクロナスメモリバッファの使用

エンドポイントタイプ	DTOG ビットの値	使用されるパケットバッファ（USB ペリフェラル）	使用されるパケットバッファ（アプリケーションソフトウェア）
入力	0	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置	ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置
	1	ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置	ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置
出力	0	ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置	ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置
	1	ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置	ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置

ダブルバッファアイソクロナスエンドポイントで発生すると、アイソクロナスエンドポイントの実装に使用される USB_EPnR レジスタは強制的に単方向として使用されます。アイソクロナスエンドポイントを受信と送信の両方について有効にする必要がある場合は、2 つの USB_EPnR レジスタを使用する必要があります。

アプリケーションソフトウェアは、最初に使用されるバッファに応じて、DTOG ビットを初期化する責任があります。これは、この 2 つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイント USB_EPnR レジスタの CTR_RX または CTR_TX ビットがセットされます。同時に、USB_EPnR レジスタの影響を受けた DTOG ビットがハードウェアによってトグルされ、バッファスワッピングを完全にソフトウェアから独立させます。STAT ビットペアはトランザクションの完了に影響を受けません。アイソクロナス転送ではハンドシェイクフェーズがないため、フロー制御は不可能であり、エンドポイントは常に 11（有効）のままです。CRC エラーまたはバッファオーバーラン

条件がアイソクロナス OUT 転送時に発生した場合、常に正しいトランザクションとみなされ、常に CTR_RX イベントをトリガします。ただし、CRC エラーが発生すると、常に USB_ISTR レジスタの ERR ビットがセットされて、データ破損の可能性をソフトウェアに通知します。

45.5.5 サスペンド／レジュームイベント

USB 標準では、SUSPEND と呼ばれる特殊なペリフェラル状態が定義され、USB バスからの平均電流は 2.5 mA より大きくなってはなりません。この要件はバスパワードデバイスにとって基本的な重要性を持ちますが、セルフパワードデバイスは、この厳格な消費電力成約を順守する必要はありません。サスペンドモードでは、ホスト PC は 3 ms を超えて USB バスにトラフィックを送信しないことによって通知を送信します。SOF パケットは、通常動作では 1 ms ごとに送信されなければならないので、USB ペリフェラルは 3 回連続の SOF パケットの欠落をホスト PC からのサスペンドリクエストとして検出し、USB_ISTR レジスタの SUSP ビットを 1 にセットし、有効な場合は割込みが生成されます。デバイスがサスペンドされると、いわゆる RESUME シーケンスによって通常操作が復元され、これはホスト PC から、またはペリフェラルそのものから直接開始できますが、常にホスト PC によって終了されます。サスペンドされた USB ペリフェラルは、常に RESET シーケンスを検出して、通常の USB リセットイベントとして、このイベントに対応できなければなりません。

USB ペリフェラルをサスペンドするために使用される実際の手順は、デバイスの構成によって合計消費量を削減するために必要なアクションが異なるため、デバイスに依存します。

USB ペリフェラルの SUSP 通知に応答するアプリケーションソフトウェアルーチンの USB 関連の局面を中心にして、典型的なサスペンド手順の簡単な説明を以下に示します。

1. USB_CNTR レジスタの FSUSP ビットを 1 にセットします。このアクションによって、USB ペリフェラル内のサスペンドモードが有効になります。サスペンドモードが有効になると、USB がサスペンドされている間に SUSP 割込みが発行されるのを回避するために、SOF 受信時のチェックが無効になります。
2. USB ペリフェラルとは異なるブロックの静的消費電力を除去または削減します。
3. USB_CNTR レジスタの LP_MODE ビットを 1 にセットして、アナログ USB トランシーバの静的消費電力を除去しつつ、レジュームアクティビティを検出できるままにしておきます。
4. オプションで、外部オシレータとデバイス PLL をオフにして、デバイス内部のアクティビティを停止します。

デバイスが SUSPEND モード中に USB イベントが発生したときには、RESUME 手順が呼び出されて、通常のチェックと復元し、通常の USB 動作を復元する必要があります。ウェイクイベントが USB リセットシーケンスである時には、このプロセスが 10 ms 以内で終了するように特に注意が必要です（詳細については、「ユニバーサルシリアルバス仕様」を参照してください）。USB ペリフェラルのサスペンド中にレジュームまたはリセットシーケンスが開始されると、USB_CNTR レジスタの LP_MODE ビットが非同期でクリアされます。レジューム手順は、サスペンド手順の終了直後に実行する必要があります。そのため、システムクロックが再起動するとすぐにそのコードが実行されます。したがって、WKUP イベントを割込みとして処理する必要はありません。複数のイベントが原因でシステムが（EXTI を介して）レジュームする可能性がある場合は、できるだけ早く USB 48 MHz クロックを再起動することをお勧めします。このクロックがアクティブになると、イベントソースが USB 再開（つまり、WKUP ビット）であったかどうかを確認できます。システムをウェイクアップする際の ESD 放電またはその他の種類のノイズを防止するために（サスペンドモードの終了は、非同期イベントです）、サスペンド中にデータラインステータスの適切なアナログフィルタを有効にします。フィルタ幅は約 70 ns です。

以下は、レジューム手順で行うべきアクションのリストです。

1. オプションで、外部オシレータおよび/またはデバイス PLL をオンにし、さらに USB 48 MHz クロックの起動を待ちます（USB レジスタのアクセスを行う前にこれが必要です）。
2. USB_CNTR レジスタの FSUSP ビットをクリアし、さらに USB 48 MHz クロックの起動を待ちます（USB レジスタのアクセスを行う前にこれが必要です）。

3. レジュームをトリガしたイベントを識別する必要がある場合は、USB_FNR レジスタのビット RXDP および RXDM を 表 415 に従って使用でき、この表には、あらゆる場合のソフトウェアのアクションもリストされています。必要な場合には、上記のビットのステータスを監視することによって、レジュームまたはリセットシーケンスの終了を検出でき、これらが「10」設定（アイドルバス状態を表します）に達したことをチェックします。さらに、リセットシーケンスの終了時には、USB_ISTR レジスタの RESET ビットが 1 にセットされ、有効な場合は割込みが発行され、通常通りに処理されます。

表 415. レジュームイベント検出

[RXDP,RXDM] ステータス	ウェイクアップイベント	必要なレジュームソフトウェアアクション
“00”	ルートのリセット	なし
“10”	なし（バス上のノイズ）	サスペンドモードに戻る
“01”	ルートのレジューム	なし
“11”	不可（バス上のノイズ）	サスペンドモードに戻る

デバイスは、USB プロトコルに直接関係しない特定のイベントに対する応答として、サスペンドモードを終了しなければならない場合があります（たとえば、マウスの移動はシステム全体をウェイクアップします）。この場合、USB_CNTR レジスタの RESUME ビットを 1 にセットし、1 ms から 15 ms までのインターバル後に 0 にリセットすることによって、レジュームシーケンスを開始できます（このインターバルは、システムクロックが公称周波数で動作しているときには 1 ms 周期で発生する ESOF 割込みによって計測できます）。RESUME ビットがクリアされると、レジュームシーケンスはホスト PC によって完了され、その終了は、USB_FNR レジスタの RXDP および RXDM ビットを使用して監視できます。

注： **RESUME ビットは、USB ペリフェラルがサスペンドモードにされた（USB_CNTR レジスタの FSUSP ビットを 1 にセットした）後でのみ使用する必要があります。**

複数のイベントが原因でシステムが（EXTI を介して）レジュームする可能性がある場合は、できるだけ早く USB 48 MHz クロックを再起動することをお勧めします。このクロックがアクティブになれば、イベント･ソースが USB レジューム（たとえば、WKUP ビット）であったかどうかを確認できます。

45.6 USB および USB SRAM レジスタ

USB ペリフェラルレジスタは、以下のグループに分けることができます。

- 共通レジスタ：割込みおよび制御レジスタ。
- エンドポイントレジスタ：エンドポイント設定およびステータス

USB SRAM レジスタは以下をカバーします。

- バッファディスクリプタテーブル：データバッファの配置に使用されたパケットメモリの位置（USB SRAM のベースアドレスは、セクション 2.2：メモリ構成を参照してください）。

すべてのレジスタアドレスは、USB ペリフェラルレジスタのベースアドレスに対するオフセットとして表されますが、バッファディスクリプタテーブル位置は USB_BTABLE レジスタで指定された値だけオフセットされた USB SRAM ベースアドレスから始まります。

レジスタの説明で使用されている略語のリストについては、72ページのセクション 1.2 を参照してください。

ペリフェラル･レジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスできます。

45.6.1 共通レジスタ

これらのレジスタは USB ペリフェラルの全般的動作に影響し、動作モード、割込み処理、デバイスアドレス、およびホスト PC によって更新される現在のフレーム番号へのアクセス付与を定義します。

USB 制御レジスタ（USB_CNTR）

アドレスオフセット：0x40

リセット値：0x0003

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTRM	PMAOVRM	ERRM	WKUPM	SUSPM	RESETM	SOFM	ESOFM	L1REQM	Res.	L1RESUME	RESUME	FSUSP	LP_MODE	PDWN	FRES
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

ビット 15 **CTRM**：正しい転送割込みマスク
0：正しい転送（CTR）割込みは無効です。
1：CTR 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 14 **PMAOVRM**：パケットメモリ領域のオーバー／アンダーラン割込みマスク
0：PMAOVR 割込みは無効です。
1：PMAOVR 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 13 **ERRM**：エラー割込みマスク
0：ERR 割込みは無効です。
1：ERR 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 12 **WKUPM**：ウェイクアップ割込みマスク
0：WKUP 割込みは無効です。
1：WKUP 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 11 **SUSPM**：サスペンド･モード割込みマスク

0：サスペンド･モード･リクエスト（SUSP）割込みは無効です。

1：SUSP 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 10 **RESETM**：USB リセット割込みマスク

0：RESET 割込みは無効です。

1：RESET 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 9 **SOFM**：フレーム開始割込みマスク

0：SOF 割込みは無効です。

1：SOF 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 8 **ESOFM**：予期されるフレーム開始割込みマスク

0：予期されるフレーム開始（ESOF）割込みは無効です。

1：ESOF 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 7 **L1REQM**：LPM L1 状態リクエスト割込みマスク

0：LPM L1 状態リクエスト（L1REQ）割込みは無効です。

1：L1REQ 割込みは有効であり、USB_ISTR レジスタの対応するビットがセットされている場合は割込みリクエストが生成されます。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **L1RESUME**：LPM L1 レジュームリクエスト

マイクロコントローラはこのビットをセットして、ホストに LPM L1 レジューム信号を送信できます。このビットは、信号の送信終了後にハードウェアによってクリアされます。

ビット 4 **RESUME**：レジューム･リクエスト

マイクロコントローラはこのビットをセットして、ホストにレジューム信号を送信できます。USB 仕様に従って、ホスト PC がレジュームシーケンスを終了まで駆動する準備ができてから 1 ms 以上 15 ms 以下で有効にする必要があります。

ビット 3 **FSUSP**：強制サスペンド

ソフトウェアは、SUSP 割込みが受信されたときにこのビットをセットする必要があり、USB ペリフェラルが 3 ms 間、トラフィックを受信しなかったときに発行されます。

0：影響はありません。

1：サスペンド･モードに入ります。アナログトランシーバのクロックと静的消費電流は影響を受けません。サスペンド消費電力が要件の場合（バスパワードデバイス）、アプリケーションソフトウェアは FSUSP の後、以下の説明に従って LP_MODE ビットをセットする必要があります。

ビット 2 **LP_MODE**：低電力モード

このモードは、サスペンドモードの電力制約から、外部プルアップレジスタに供給される電力を除き、すべての静的消費電流を回避されるときに使用されます。USB サスペンド条件の消費電力要件を満たすために、アプリケーションがすべてのシステムクロックを停止するか、周波数を削減する準備ができたときに、この条件に入るべきです。サスペンドモード中に USB アクティビティ（WKUP イベント）が発生すると、このビットは非同期でリセットされます（ソフトウェアによってリセットすることもできます）。

0：低電力モードではありません。

1：低電力モードに入ります。

ビット 1 **PDWN**：パワー･ダウン

このビットは、何らかの理由で USB ペリフェラルを完全に無効にする必要がある場合、USB 関連のすべてのアナログ部品を完全にオフにするために使用されます。このビットがセットされると、USB ペリフェラルはトランシーバから切断され、使用できなくなります。

0：パワー･ダウンを終了します。

1：パワー･ダウン･モードに入ります。

ビット 0 **FRES**：強制的 USB リセット

0：USB リセットをクリアします。

1：USB に RESET 信号が送られたときと同様に、USB ペリフェラルを強制的にリセットします。ソフトウェアがこのビットをクリアするまで、USB ペリフェラルは RESET 状態に保たれます。有効な場合は、「USB-RESET」割込みが生成されます。

USB 割込みステータスレジスタ（USB_ISTR）

アドレスオフセット：0x44

リセット値：0x0000 0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTR	PMA OVR	ERR	WKUP	SUSP	RESET	SOF	ESOF	L1REQ	Res.	Res.	DIR	EP_ID[3:0]			
r	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0			r	r	r	r	r

このレジスタはすべての割込みソースのステータスを含むため、アプリケーションソフトウェアは割込みリクエストの原因になったイベントを判断できます。

このレジスタの上位部分はシングルビットを含み、そのそれぞれが特定のイベントを表します。これらのビットは、関連するイベントが発生したときにハードウェアによってセットされます。USB_CNTR レジスタの対応するビットがセットされている場合、汎用割込みリクエストが生成されます。各ビットを調べる割込みルーチンは、必要なすべてのアクションを実行し、最後に処理済みビットをクリアします。そのいずれかがクリアされていない場合、割込みはまだ保留中であるとみなされ、割込みラインは再びハイに保たれます。複数のビットが同時にセットされた場合、1 つの割込みだけが生成されます。

エンドポイントトランザクションの完了は、割込み応答遅延を軽減するために、さまざまな方法で処理できます。CTR ビットは、エンドポイントがトランザクションを正常に完了するとすぐにハードウェアによってセットされ、USB_CNTR の対応するビットがセットされていた場合は汎用割込みリクエストが生成されます。エンドポイント専用割込み条件は、USB_CNTR レジスタの CTRM ビットとは無関係に有効化されます。どちらの割込み条件も、ソフトウェアが対応する USB_EPnR レジスタの保留中のビットをクリアするまでアクティブなままです（CTR ビットは実際には読出し専用ビットです）。エンドポイント関連の割込みの場合、ソフトウェアは、トランザクション方向（DIR）および EP_ID 読出し専用ビットを使用して、最後の割込みリクエストを発行して、対応する割込みサービスルーチンを呼び出したエンドポイントを特定できます。

ユーザは、ソフトウェアが割込みサービスルーチンで USB_ISTR ビットをチェックする順序を指定することによって、同時に保留中の USB_ISTR ビットの相対的優先順位を選ぶことができます。イベントに関連し、処理済みのビットだけがクリアされます。サービスルーチンの終了時、残りの条件を処理するために、別の割込みがリクエストされます。

ビットの誤ったクリアを避けるために、ロード命令付きでクリアすることが推奨され、その場合、変更してはならないすべてのビットに 1 が書き込まれ、クリアすべきすべてのビットに 0 が書き込まれます（これらのビットはソフトウェアによってのみクリアできます）。読出し-変更-書込みのサイクルは避けるべきです。読出し操作と書込み操作の間に、別のビットがハードウェアによってセットされる可能性があり、マイクロプロセッサにイベントを処理する時間ができる前に、次の書込みによってクリアされる可能性があるためです。

以下に、各ビットについて詳しく説明します。

ビット 15 **CTR**：正しい転送

このビットは、エンドポイントがトランザクションを正常に完了したことを示すために、ハードウェアによってセットされます。DIR および EP_ID ビットを使用して、ソフトウェアは割込みをリクエストしたエンドポイントを判断できます。このビットは読出し専用です。

ビット 14 **PMAOVR**：パケット･メモリ領域のオーバーラン／アンダーラン

このビットは、マイクロコントローラに USB メモリリクエストに対応する時間がなくなった場合にセットされます。USB ペリフェラルは、このイベントを次のように処理します。受信時、ACK ハンドシェイクパケットは送信されず、送信時、送信ストリームのビットスタッフエラーが強制的に発行されます。いずれの場合も、ホストはトランザクションを再試行します。通常操作では、PMAOVR 割込みは発生しません。失敗したトランザクションはホストによって再試行されるので、アプリケーションソフトウェアには、この割込み処理中にデバイス操作をスピードアップして、次のトランザクションの差異施行に備える機会があります。ただし、アイソクロナス転送では（アイソクロナストランザクションは再試行されません）、データの喪失につながるため、これは行われません。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 13 **ERR**：エラー

このフラグは、下記にリストされているエラーのいずれかが発生したときにセットされます。
NANS：応答なし（No ANSwer）。ホスト応答がタイムアウトになりました。
CRC：巡回冗長検査（Cyclic Redundancy Check）エラー。トークンまたはデータで受信された CRC の 1 つが正しくありませんでした。
BST：ビット･スタッフィング･エラー。ビットスタッフィングエラーが、PID、データ、または CRC で検出されました。
FVIO：フレーミング･フォーマット違反。非標準のフレームが受信されました（正しい位置にない EOP、正しくないトークンシーケンスなど）。
USB ペリフェラルと PC ホストは、エラーの場合、完全にトランスペアレントな方法で再送信を管理するため、USB ソフトウェアは、通常、エラーを無視できます。この割込みは、ソフトウェア開発フェーズで、またはUSB バスでの送信品質の監視に役立ち、問題の可能性をユーザに知らせることができます（コネクタの緩み、ノイズが多い環境、USB ケーブルの導体の破損など）。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 12 **WKUP**：ウェイクアップ

このビットは、サスペンドモード中に、USB ペリフェラルをウェイクアップするアクティビティが検出されたときにハードウェアによって 1 にセットされます。このイベントは CTLR レジスタの LP_MODE ビットを非同期でクリアし、USB_WAKEUP ラインを有効にし、レジュームプロセスの開始に関して残りのデバイス（ウェイクアップユニットなど）に通知するために使用されます。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 11 **SUSP**：サスペンド･モード･リクエスト

このビットは、3 ms にわたってトラフィックが受信されず、USB バスからのサスペンドモードリクエストを示したときに、ハードウェアによってセットされます。サスペンド条件チェックは、USB リセットの直後に有効化され、サスペンドモードがアクティブなとき（FSUSP=1）、レジュームシーケンスの終了まで、ハードウェアによって無効にされます。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 10 **RESET**：USB リセット･リクエスト

USB ペリフェラルが入力でアクティブな USB RESET 信号を検出したときにセットされます。USB ペリフェラルは、RESET に対応して、内部プロトコル状態マシンをリセットし、USB_CNTR レジスタの RESETM 有効ビットがセットされている場合、割込みが生成されます。RESET ビットがクリアされるまで、受信と送信は無効です。すべての設定レジスタがリセットされるわけではありません。マイクロコントローラはこれらのレジスタを明示的にクリアする必要があります（これは、RESET 割込みの安全な配信を確保し、RESET の直前のトランザクションを完了できるようにするためです）。機能アドレスおよびエンドポイントレジスタは、USB リセットイベントによってリセットされます。
このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 9 **SOF**：フレーム開始

このビットは、新しい USB フレームの開始を知らせ、SOF パケットが USB バス経由で着信したときにセットされます。割込みサービスルーチンによって SOF イベントを監視して、USB ホストへの 1 ms の同期イベントを実行し、SOF パケット受信時に更新される USB_FNR レジスタを安全に読み出すことができます（これはアイソクロナスアプリケーションに便利です）。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 8 **ESOF**：予期されたフレーム開始

このビットは、予期された SOF パケットが受信されなかったときにハードウェアによってセットされます。ホストは 1 ms ごとに SOF パケットを送信しますが、デバイスが正しく受信しなかった場合、サスペンドタイマがこの割込みを発行します。3 回連続して ESOF 割込みが生成され（すなわち、3 つの SOF パケットが失われ）、その間にトラフィックが発生しなかった場合、SUSP 割込みが生成されます。このビットは、サスペンドタイマがまだロックされていないときに SOF パケットの喪失が発生したときにセットされます。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 7 **L1REQ**：LPM L1 状態リクエスト

このビットは、L1 状態を入力する LPM コマンドが正常に受信され、確認応答された場合に、ハードウェアによってセットされます。このビットは読出し／書込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 6:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **DIR**：トランザクションの方向

このビットは、割込みリクエストを生成した成功したトランザクションの方向に従って、ハードウェアによって書き込まれます。

DIR ビット =0 の場合、割込みをかけたエンドポイントに関連する USB_EPnR レジスタの CTR_TX ビットがセットされます。割込みの原因となったトランザクションは、IN タイプです（USB ペリフェラルからホスト PC へのデータ送信）。

DIR ビット =1 の場合、割込みをかけたエンドポイントに関連する USB_EPnR レジスタの CTR_RX ビットまたは CTR_TX と CTR_RX の両方がセットされます。割込みの原因となったトランザクションは OUT タイプであるか（ホスト PC からのデータを USB ペリフェラルが受信）、または 2 つの保留中のトランザクションが処理を待っています。

これは割込みを保留中にした方向を表すので、アプリケーションソフトウェアはこの情報を使用して、トリガしたトランザクションに関連する USB_EPnR ビットにアクセスできます。このビットは読出し専用です。

ビット 3:0 **EP_ID[3:0]**：エンドポイント識別子

これらのビットは、割込みリクエストを生成したエンドポイント番号に従って、ハードウェアによって書き込まれます。複数のエンドポイントトランザクションが保留中の場合、ハードウェアは、次のように定義される最も高い優先順位を持つエンドポイントに関連するエンドポイント識別子を書き込みます。優先順位の順に、2 つのエンドポイントセットが定義されます。アイソクロナスおよびダブルバッファバルクエンドポイントが最初に考慮され、次に、その他のエンドポイントが調べられます。同じセットの複数のエンドポイントが割込みをリクエストしている場合、最下位リクエストエンドポイントレジスタに従って、USB_ISTR レジスタの EP_ID ビットが割り当てられ、EP0R が最も高い優先順位を持ち、その後に EP1R が続き、以下同様です。アプリケーションソフトウェアは、この優先順位方式に従って、レジスタを各エンドポイントに割り当てて、同時発生したエンドポイントリクエストを適切に順序づけることができます。これらのビットは読出し専用です。

USB フレーム番号レジスタ（USB_FNR）

アドレスオフセット：0x48

リセット値：0x0XXX（X は未定義）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDP	RXDM	LCK	LSOF[1:0]		FN[10:0]										
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15 **RXDP**：受信データ + ライン･ステータス

このビットを使用して、受信データプラスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。

ビット 14 **RXDM**：受信データ - ライン･ステータス

このビットを使用して、受信データマイナスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。

ビット 13 **LCK**：ロック

このビットは、USB リセット条件の終了後、または USB レジュームシーケンスの終了後に、少なくとも 2 つ連続した SOF パケットが受信されたときに、ハードウェアによってセットされます。ロックされると、USB リセットまたは USB サスペンドイベントが発生するまで、フレームタイマはこの状態に保たれます。

ビット 12:11 **LSOF[1:0]**：SOF の喪失

これらのビットは、ESOF 割込みが生成されたときに、ハードウェアによって書き込まれ、失われた連続 SOF パケットの数をカウントします。SOF パケットの受信時、これらのビットはクリアされます。

ビット 10:0 **FN[10:0]**：フレーム番号

このビットフィールドは、最後に受信された SOF パケットに含まれる 11 ビットのフレーム番号を含みます。フレーム番号は、ホストによって送信されたフレームごとにインクリメントされ、アイソクロナス転送の場合に便利です。このビットフィールドは、SOF 割込みの生成時に更新されます。

USB デバイスアドレス（USB_DADDR）

アドレスオフセット：0x4C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EF	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済みです。

ビット 7 **EF**：機能有効

このビットは、USB デバイスを有効にするために、ソフトウェアによってセットされます。このデバイスのアドレスは、次の ADD[6:0] ビットに含まれます。このビットが 0 の場合、USB_EPnR レジスタの設定にかかわらず、トランザクションは処理されません。

ビット 6:0 **ADD[6:0]**：デバイスアドレス

これらのビットは、エニュメレーションプロセス時にホスト PC によって割り当てられた USB 機能を含みます。必要なエンドポイントへのトランザクションを処理するためには、関連する USB_EPnR レジスタのこのフィールドとエンドポイントアドレス（EA）フィールドの両方とも、USB トークンに含まれる情報に一致する必要があります。

バッファテーブルアドレス（USB_BTABLE）

アドレスオフセット：0x50

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BTABLE[15:3]													Res.	Res.	Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			

ビット 15:3 **BTABLE[15:3]**：バッファ･テーブル

これらのビットは、専用パケットメモリ内のバッファ割り当てテーブルの開始アドレスを含みます。このテーブルは、各エンドポイントバッファ位置とサイズを記述し、8 バイトの境界に揃えられなければなりません（3 つの最下位ビットが常に 0）。このデバイスにアドレス指定されたトランザクションの開始ごとに、USB ペリフェラルはアドレス指定されたエンドポイントに関連するこのテーブルの要素を読み出して、バッファ開始位置とバッファサイズを取得します（1977 ページのパケットバッファの構造と用途を参照）。

ビット 2:0 予約済み、ハードウェアによって 0 に固定されています。

LPM 制御およびステータスレジスタ（USB_LPMCSR）

アドレスオフセット：0x54

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BESL[3:0]				REM WAKE	Res.	LPM ACK	LPM EN
								r	r	r	r	r		rw	rw

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:4 **BESL[3:0]**：BESL 値

これらのビットは、最後に ACK された LPM トークンとともに受信した BESL 値を含みます。

ビット 3 **REMWAKE**：bRemoteWake 値

このビットは、最後に ACK された LPM トークンとともに受信した bRemoteWake 値を含みます。

ビット 2 予約済みです。

ビット 1 **LPMACK**：LPM トークン確認応答イネーブル

0：有効な LPM トークンは NYET です。
1：有効な LPM トークンは ACK です。
NYET/ACK は、LPM トランザクションが成功した場合にのみ返されます。
EXT トークンと LPM トークンの両方にエラーがない場合（それ以外の場合は ERROR）
有効な bLinkState = 0001B（L1）を受信した場合（それ以外の場合は STALL）

ビット 0 **LPMEN**：LPM サポート･イネーブル

このビットは、USB デバイス内の LPM サポートを有効にするために、ソフトウェアによってセットされます。このビットが 0 の場合は、LPM トランザクションは処理されません。

バッテリ充電検出回路(USB_BCDR)

アドレスオフセット：0x58

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DPPU	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2 DET	SDET	PDET	DC DET	SDEN	PDEN	DCD EN	BCD EN
rw								r	r	r	r	rw	rw	rw	rw

ビット 15 **DPPU**：DP プルアップ制御

このビットは、DP ライン上の組み込みプルアップを有効にするために、ソフトウェアによってセットされます。0 にクリアすると、ユーザのソフトウェアが必要とするときに、ホストとの接続が切断されたことを通知します。

ビット 14:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **PS2DET**：DM プルアップ検出ステータス

このビットは､PD 中にのみアクティブになって DM 電圧レベルと V_{LGC} 閾値の比較の結果を示します。通常の状況では、DM レベルは閾値を下回る必要があります。閾値を上回る場合は、DM が外部的にハイレベルにプルアップされることを意味します。これは、PS2 ポート（DP および DM ラインの両方をプルアップする）への接続、または BCD 仕様に準拠しない一部の独自の充電器への接続によって発生します。

0：通常のポートが検出されました（SDP、ACA、CDP、または DCP に接続されたポート）。

1：PS2 ポートまたは独自の充電器が検出されました。

ビット 6 **SDET**：2 次検出（SD）ステータス

このビットは、SD の結果を示します。

0：CDP が検出されました。

1：DCP が検出されました。

ビット 5 **PDET**：1 次検出（PD）ステータス

このビットは、PD の結果を示します。

0：BCD サポートは検出されませんでした（SDP または独自デバイスに接続）。

1：BCD のサポートが検出されました（ACA、CDP、または DCP に接続）。

ビット 4 **DCDET**：データ接触検出（DCD）ステータス

このビットは、DCD の結果を示します。

0：データラインの接触は検出されませんでした。

1：データラインの接触が検出されました。

ビット 3 **SDEN**：2 次検出（SD）モード･イネーブル

このビットは、BCD を SD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 2 **PDEN**：1 次検出（PD）モード･イネーブル

このビットは、BCD を PD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 1 **DCDEN**：データ接触検出（DCD）モード･イネーブル

このビットは、BCD を DCD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード（DCD、PD、SD、または OFF）を 1 つだけ選択する必要があります。

ビット 0 **BCDEN**：バッテリ充電検出（BCD）イネーブル

このビットは、USB デバイス内の BCD サポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHY は BCD によって完全に制御され、通常の通信では使用できなくなります。BCD の検出完了後は、通常の USB 動作を可能にするために、このビットを 0 にクリアして BCD を OFF モードにします。

エンドポイント固有レジスタ

これらのレジスタの数は､USB ペリフェラルが処理する設計になっているエンドポイント数に応じて変化します。USB ペリフェラルは、最大 8 つの双方向エンドポイントをサポートします。各 USB デバイスは、コントロールエンドポイントをサポートしなければならず、そのアドレス（EA ビット）は 0 にセットされなければなりません。複数のエンドポイントが有効であり、同じエンドポイント番号値を持つ場合、USB ペリフェラルの動作は未定義です。各エンドポイントについて、USB_EPnR レジスタはエンドポイント固有の情報を格納できます。

USB エンドポイント n レジスタ（USB_EPnR）、n=[0..7]

アドレスオフセット：0x00 から 0x1C

リセット値：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTR_RX	DTOG_RX	STAT_RX[1:0]		SETUP	EP TYPE[1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX[1:0]		EA[3:0]			
rc_w0	t	t	t	r	rw	rw	rw	rc_w0	t	t	t	rw	rw	rw	rw

USB リセットが USB バスから受信されたとき､または CTLR レジスタのビット FRES を通じて強制されたときにもリセットされますが､USB リセットイベントの直前の正しいパケット通知の喪失を回避するために、CTR_RX および CTR_TX ビットは変更されません。各エンドポイントに USB_EPnR レジスタがあり、*n* はエンドポイント識別子です。

これらのレジスタでは読出し-変更-書込みのサイクルは避けるべきです。読出し操作と書込み操作の間に、いくつかのビットがハードウェアによってセットされる可能性があり、CPU に変更を検出する時間ができる前に、次の書込みによって変更される可能性があるためです。この目的のために、この問題の影響を受けるすべてのビットが「不変」の値を持ち、変更が不要なときに使用される必要があります。これらのレジスタはロード命令付きで変更することが推奨され、この場合、ハードウェアによって変更できるすべてのビットには「不変」値が書き込まれます。

ビット 15 **CTR_RX**：受信の場合の正しい転送

このビットは、このエンドポイントで OUT/SETUP トランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアでは、このビットのクリアのみができます。USB_CNTR レジスタの CTRM ビットがセットされた場合、エンドポイント関連の割込み条件（常に有効化されます）とともに汎用割込み条件が生成されます。発生したトランザクションのタイプ（OUT または SETUP）は、下記の SETUP ビットから決めることができます。

NAK または STALL ハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。

このビットは読出し／書込みですが､0 のみを書き込むことができ､1 を書き込んでも効果はありません。

ビット 14 **DTOG_RX**：受信転送の場合のデータ･トグル

エンドポイントがアイソクロナスでない場合、このビットには、次の受信データ･パケットのデータ･トグル･ビット（0 = DATA0、1 = DATA1）に想定される値が設定されます。データ PID 値が一致するデータパケット受信後、ACK ハンドシェイクが USB ホストに送信されたときに、ハードウェアはこのビットをトグルします。エンドポイントが制御エンドポイントとして定義された場合、ハードウェアはこのエンドポイントにアドレス指定された SETUP PID の受信時に、このビットをクリアします。
エンドポイントがダブルバッファリングを使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます（セクション 45.5.3 : ダブルバッファエンドポイントを参照）。
エンドポイントがアイソクロナスの場合、この種類のエンドポイントではデータトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケットバッファスワッピングをサポートするためにのみ使用されます（セクション 45.5.4 : アイソクロナス転送を参照）。アイソクロナス転送ではハンドシェイクは使用されないので、ハードウェアはデータパケット受信の終了直後に、このビットをトグルします。
このビットは、値を初期化するため（エンドポイントが制御エンドポイントでないときは必須）、または特定のデータトグル／パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが 0 を書き込むと、DTOG_RX の値は変更されませんが、1 を書き込むと、ビット値がトグルされます。このビットは読出し／書込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 13:12 **STAT_RX[1:0]**：受信転送の場合のステータスビット

これらのビットは、表 416 : 1998 ページの受信ステータスエンコードにリストされているエンドポイントステータスに関する情報を含みます。これらのビットは、ソフトウェアによってトグルして、値を初期化することができます。アプリケーションソフトウェアが 0 を書き込むと、値は変更されませんが、1 を書き込むと、ビット値がトグルされます。ハードウェアは、このエンドポイントにアドレス指定された OUT または SETUP（制御のみ）トランザクションに対応する正しい転送が発生したとき（CTR_RX=1）、STAT_RX ビットを NAK にセットするので、ソフトウェアは新しいトランザクションに確認応答する前に受信データを処理する時間があります。
ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します（セクション 45.5.3 : ダブルバッファエンドポイントを参照）。
エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみなので、ハードウェアはトランザクションの成功後にエンドポイントのステータスを変更できません。ソフトウェアがアイソクロナスエンドポイントの STAT_RX ビットを STALL または NAK にセットした場合、USB ペリフェラルの動作は未定義です。これらのビットは読出し／書込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 11 **SETUP**：セットアップ･トランザクション完了

このビットは読出し専用であり、最後に完了したトランザクションが SETUP のときにハードウェアによってセットされます。このビットはコントロールエンドポイントについてのみ値を変更します。成功した受信トランザクションの場合（CTR_RX イベント）、これを調べて、発生したトランザクションのタイプを判断する必要があります。割込みサービスルーチンを次の着信トークンによる SETUP ビットの変更から保護するために、このビットは CTR_RX ビットが 1 の間は停止され、CTR_RX が 0 のときに状態が変更されます。このビットは読出し専用です。

ビット 10:9 **EP_TYPE[1:0]**：エンドポイントタイプ

これらのビットは、表 417：1998 ページのエンドポイントタイプエンコードに記載されているように、このエンドポイントの動作を設定します。エンドポイント 0 は常にコントロールエンドポイントでなければならず、各 USB 機能にはアドレス 0 を持つコントロールエンドポイントが少なくとも 1 つ必要ですが、必要な場合は、他のコントロールエンドポイントを使用することもできます。制御エンドポイントだけが SETUP トランザクションを処理し、他の種類のエンドポイントによって無視されます。SETUP トランザクションは NAK または STALL で応答することはできません。コントロールエンドポイントが NAK として定義された場合、SETUP トランザクションが受信されたときには、受信方向では USB ペリフェラルは応答せず、受信エラーをシミュレートします。制御エンドポイントが受信方向で STALL として定義された場合、SETUP パケットは受け入れられ、データを送信し、CTR 割込みを発行します。エンドポイントが制御エンドポイントの場合でも、OUT トランザクションの受信は通常通りに処理されます。

バルクおよび割込みエンドポイントの動作はほぼ同様ですが、EP_KIND 設定ビットを使用して使用可能な特殊な機能のみが異なります。

アイソクロナスエンドポイントの使用については、セクション 45.5.4：アイソクロナス転送で説明します。

ビット 8 **EP_KIND**：エンドポイントの種類

このビットの意味は、EP_TYPE ビットによって設定されたエンドポイントのタイプに依存します。表 418 はさまざまな意味を要約します。

DBL_BUF：このビットは、このバルクエンドポイントのダブルバッファリング機能を有効にするために、ソフトウェアによってセットされます。ダブルバッファバルクエンドポイントの使用については、セクション 45.5.3：ダブルバッファエンドポイントで説明します。

STATUS_OUT：このビットは、ステータスアウトトランザクションが予期されることを示すためにソフトウェアによってセットされます。この場合、1 個以上のデータバイトを含んでいるすべての OUT トランザクションは、ACK ではなく STALL で応答されます。このビットを使用して、制御転送時のプロトコルエラーに対するアプリケーションの堅牢性を向上させることができ、その使用はコントロールエンドポイントのみを対象としています。STATUS_OUT がリセットされると、OUT トランザクションは必要に応じて任意の数のバイトを持つことができます。

ビット 7 **CTR_TX**：送信の場合の正しい転送

このビットは、このエンドポイントで IN トランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアのみがこのビットをクリアできます。USB_CNTR レジスタの CTRM ビットがセットされた場合、エンドポイント関連の割込み条件（常に有効化されます）とともに汎用割込み条件が生成されます。

NAK または STALL ハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。

このビットは読出し／書込みですが、0 のみを書き込むことができます。

ビット 6 **DTOG_TX**：送信転送の場合のデータトグル

エンドポイントがアイソクロナスでない場合、このビットは次の送信データパケットのデータトグルビット（0=DATA0、1=DATA1）の必要な値を含みます。データパケットの送信後、USB ホストから ACK ハンドシェイクが受信されると、ハードウェアはこのビットをトグルします。エンドポイントが制御エンドポイントとして定義された場合、このエンドポイントにアドレス指定された SETUP PID の受信時に、ハードウェアはこのビットを 1 にセットします。

エンドポイントがダブルバッファ機能を使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます（セクション 45.5.3：ダブルバッファエンドポイントを参照）。

エンドポイントがアイソクロナスの場合、この種類のエンドポイントではデータトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケットバッファスワッピングをサポートするためにのみ使用されます（セクション 45.5.4：アイソクロナス転送を参照）。アイソクロナス転送ではハンドシェイクは使用されないので、ハードウェアはデータパケット送信の終了直後に、このビットをトグルします。

このビットは、値を初期化するため（エンドポイントが制御エンドポイントでないときは必須）、または特定のデータトグル／パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが 0 を書き込むと、DTOG_TX の値は変更されませんが、1 を書き込むと、ビット値がトグルされます。このビットは読出し／書込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 5:4 **STAT_TX[1:0]**：送信転送の場合のステータスビット

これらのビットは、表 419 にリストされているエンドポイントのステータスに関する情報を含みます。これらのビットは、ソフトウェアによってトグルして、値を初期化できます。アプリケーションソフトウェアが 0 を書き込むと、値は変更されませんが、1 を書き込むと、ビット値がトグルされます。ハードウェアは、このエンドポイントにアドレス指定された IN または SETUP（制御のみ）トランザクションに対応する正しい転送が発生したとき（CTR_TX=1）、STAT_TX ビットを NAK にセットします。次に、ソフトウェアが次の送信データセットを準備するのを待ちます。

ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します（セクション 45.5.3: ダブルバッファエンドポイントを参照）。エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみです。したがって、ハードウェアは、トランザクションの成功後にエンドポイントのステータスを変更することはできません。ソフトウェアがアイソクロナスエンドポイントの STAT_TX ビットを STALL または NAK にセットした場合、USB ペリフェラルの動作は未定義です。これらのビットは読出し／書込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 3:0 **EA[3:0]**：エンドポイント･アドレス

ソフトウェアは、このエンドポイント宛てのトランザクションを識別するために使用される 4 ビットのアドレスをこのフィールドに書き込む必要があります。値は、対応するエンドポイントを有効にする前に書き込まれる必要があります。

表 416. 受信ステータスエンコード

STAT_RX[1:0]	意味
00	**DISABLED**：このエンドポイントにアドレス指定されたすべての受信リクエストは無視されます。
01	**STALL**：エンドポイントは停止していて、すべての受信リクエストは STALL ハンドシェイクで終了します。
10	**NAK**：エンドポイントは NAK され、すべての受信リクエストは NAK ハンドシェイクで終了します。
11	**VALID**：このエンドポイントは受信に有効です。

表 417. エンドポイントタイプエンコード

EP_TYPE[1:0]	意味
00	BULK
01	CONTROL
10	ISO
11	INTERRUPT

表 418. エンドポイントの種類の意味

EP_TYPE[1:0]		EP_KIND の意味
00	BULK	DBL_BUF
01	CONTROL	STATUS_OUT
10	ISO	未使用
11	INTERRUPT	未使用

表 419. 送信ステータス･エンコード

STAT_TX[1:0]	意味
00	**DISABLED**：このエンドポイントにアドレス指定されたすべての送信リクエストは無視されます。
01	**STALL**：エンドポイントは停止していて、すべての送信リクエストは STALL ハンドシェイクで終了します。
10	**NAK**：エンドポイントは NAK され、すべての送信リクエストは NAK ハンドシェイクで終了します。
11	**VALID**：このエンドポイントは送信に有効です。

45.6.2 バッファディスクリプタテーブル

注： **バッファディスクリプタテーブルは、別の「USB SRAM」アドレス空間のパケットバッファメモリ内に位置しています。**

バッファディスクリプタテーブルはパケットバッファメモリ（「USB SRAM」領域）内にありますが、そのエントリは､USB マクロセルとデバイスの間でデータを交換するためのパケットバッファの位置とサイズを設定する追加レジスタとみなすことができます。

最初のパケットメモリ位置は、USB SRAM ベースアドレスにあります。USB_EPnR レジスタに関連するバッファディスクリプタテーブルのエントリについて、以下に説明します。パケットメモリへのアクセスは、バイト（8 ビット）またはハーフワード（16 ビット）アクセスによってのみ行う必要があります。ワード（32 ビット）アクセスは許されません。

パケットバッファとバッファディスクリプタテーブルの使用に関する詳細については、1977 ページのパケットバッファの構造と用途を参照してください。

送信バッファアドレス n（USB_ADDRn_TX）

アドレスオフセット：[USB_BTABLE] + n*8

注： **IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_TX_0 と呼ばれます。**

OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_RX_0 と呼ばれます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRn_TX[15:1]															-
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	-

ビット 15:1 **ADDRn_TX[15:1]**：送信バッファアドレス

これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の IN トークンで送信されるデータを含んでいるパケットバッファの開始アドレスを指します。

ビット 0 パケットメモリはハーフワード長であり、すべてのパケットバッファがハーフワードで揃えられている必要があるため、常に 0 が書き込まれる必要があります。

送信バイトカウント n（USB_COUNTn_TX）

アドレスオフセット：[USB_BTABLE] + n*8 + 2

注： **IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_TX_0 と呼ばれます。**

OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_RX_0 と呼ばれます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	COUNTn_TX[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:10 USB 仕様によりパケットサイズは 1023 バイトに制限されるので、これらのビットは使用されません。これらの値は、USB ペリフェラルによって考慮されません。

ビット 9:0 **COUNTn_TX[9:0]**：送信バイトカウント

これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の IN トークンで送信されるバイト数を含みます。

受信バッファアドレス n（USB_ADDRn_RX）

アドレスオフセット：[USB_BTABLE] + n*8 + 4

注： **OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_RX_1 と呼ばれます。**

IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_TX_1 と呼ばれます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRn_RX[15:1]															-
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	-

ビット 15:1 **ADDRn_RX[15:1]**：送信バッファアドレス

これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の OUT/SETUP トークンで受信されるデータを含んでいるパケットバッファの開始アドレスを指します。

ビット 0 パケットメモリはハーフワード長であり、すべてのパケットバッファがハーフワードで揃えられている必要があるため、常に 0 が書き込まれる必要があります。

受信バイトカウント n（USB_COUNTn_RX）

アドレスオフセット：[USB_BTABLE] + n*8 + 6

注： **OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_RX_1 と呼ばれます。**

IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_TX_1 と呼ばれます。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLSIZE	NUM_BLOCK[4:0]					COUNTn_RX[9:0]									
rw	rw	rw	rw	rw	rw	r	r	r	r	r	r	r	r	r	r

このテーブル位置は、両方ともパケット受信時に必要とされる 2 つの値を格納するために使用されます。最上位ビットは割り当てられるバッファサイズの定義を含み、バッファオーバーフロー検出を可能にし、この位置の最下位部分には、受信終了時に USB ペリフェラルによって実際の受信バイト数が書き込まれます。使用可能なビット数が制限されているため、バッファサイズは割り当てられたメモリブロック数を使用して表され、細かい粒度／小さいバッファと粗い粒度／大きなバッファの間のトレードオフにより、ブロックサイズを選択できます。割り当てられたバッファのサイズは、エンドポイントディスクリプタの一部であり、通常、エニュメレーションプロセス時に maxPacketSize パラメータ値にしたって定義されます（「ユニバーサルシリアルバス仕様」を参照）。

ビット 15 **BL_SIZE**：ブロックサイズ

このビットは、割り当てられたバッファ領域を定義するために使用されるメモリブロックのサイズを選択します。

– BL_SIZE=0 の場合、メモリブロックは 2 バイト長であり、これは、ハーフワード長のメモリで可能な最小ブロックです。このブロックサイズでは、割り当てられるバッファサイズは 2 ～ 62 バイトです。

– BL_SIZE=1 の場合、メモリブロックは 32バイト以上であり、USB 仕様によって定義された最大パケット長に達することができます。このブロックサイズでは、割り当てられるバッファサイズは、理論上、32 ～ 1024 バイトであり、USB 標準仕様で可能な最長のパケットサイズです。ただし、適用可能なサイズは、使用可能なバッファメモリによって制限されます。

ビット 14:10 **NUM_BLOCK[4:0]**：ブロック数

これらのビットは、このパケットバッファに割り当てられるメモリブロック数を定義します。実際の割り当てメモリの量は、表 420 に示されているように、BL_SIZE の値に依存します。

ビット 9:0 **COUNTn_RX[9:0]**：受信バイトカウント

これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、最後の OUT/SETUP トランザクションで受信されたバイト数を含みます。

表 420. 割り当てられるバッファメモリの定義

値 NUM_BLOCK[4:0]	メモリ割り当て （BL_SIZE=0 の場合）	メモリ割り当て （BL_SIZE=1 の場合）
0 (‘00000)	設定禁止	32 バイト
1 (‘00001)	2 バイト	64 バイト
2 (‘00010)	4 バイト	96 バイト
3 (‘00011)	6 バイト	128 バイト
.....	.....	.....
14 (‘01110)	28 バイト	480 バイト
15 (‘01111)	30 バイト	
16 (‘10000)	32 バイト	
.....	.....	.....
29 (‘11101)	58 バイト	
30 (‘11110)	60 バイト	
31 (‘11111)	62 バイト	N/A

45.6.3 USB レジスタマップ

次の表に、USB レジスタ･マップとリセット値を示します。

表 421. USB レジスタのマップとリセット値

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x00	**USB_EP0R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x04	**USB_EP1R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x08	**USB_EP2R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0C	**USB_EP3R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	**USB_EP4R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x14	**USB_EP5R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	**USB_EP6R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x1C	**USB_EP7R**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR_RX	DTOG_RX	STAT_RX [1:0]		SETUP	EP TYPE [1:0]		EP_KIND	CTR_TX	DTOG_TX	STAT_TX [1:0]		EA[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20-0x3F	予約済みです。																																
0x40	**USB_CNTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTRM	PMAOVRM	ERRM	WKUPM	SUSPM	RESETM	SOFM	ESOFM	L1REQM	Res.	L1RESUME	RESUME	FSUSP	LP_MODE	PDWN	FRES
	リセット値																	0	0	0	0	0	0	0	0	0		0	0	0	0	1	1
0x44	**USB_ISTR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CTR	PMAOVR	ERR	WKUP	SUSP	RESET	SOF	ESOF	L1REQ	Res.	Res.	DIR	EP_ID[3:0]			
	リセット値																	0	0	0	0	0	0	0	0	0			0	0	0	0	0
0x48	**USB_FNR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDP	RXDM	LCK	LSOF [1:0]		FN[10:0]										
	リセット値																	0	0	0	0	0	x	x	x	x	x	x	x	x	x	x	x
0x4C	**USB_DADDR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	EF	ADD[6:0]						
	リセット値																									0	0	0	0	0	0	0	0
0x50	**USB_BTABLE**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BTABLE[15:3]													Res.	Res.	Res.
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0			

表 421. USB レジスタのマップとリセット値 （続き）

オフセット	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x54	USB_LPMCSR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	BESL[3:0]				REMWAKE	Res.	LPMACK	LPMEN
	リセット値																									0	0	0	0	0		0	0
0x58	USB_BCDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DPPU	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PS2DET	SDET	PDET	DCDET	SDEN	PDEN	DCDEN	BCDEN
	リセット値																	0								0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

46 USB Type-C®/USB Power Delivery インタフェース（UCPD）

46.1 概要

USB Type-C/USB Power Delivery インタフェースは以下の仕様に準拠しています。

- Universal Serial Bus Type-C Cable and Connector Specification Release 2.0（2019年8月）
- Universal Serial Bus Power Delivery Specifications
  - Revision 2.0, Version 1.3（2017 年 1 月 12 日）
  - Revision 3.0, Version 2.0（2019 年 8 月 29 日）

このインタフェースは、Power Delivery（PD）仕様の物理層に CC ラインによる通知方法（VBUS なし）を統合し、Type-C ケーブルで動作できるようにしています。

46.2 UCPD の主な特徴

- USB Type-C 仕様リリース 2.0 準拠
- USB Power Delivery 仕様リビジョン 2.0 および 3.0 準拠
  - PPS（プログラム可能電源）などの高度なアプリケーションの実現
- STOP モードでの低電力動作のサポート
- 内蔵アナログ PHY
  - USB Type-C のプルアップ抵抗（すべての値の Rp）およびプルダウン抵抗（Rd）
  - 「デッド･バッテリ」抵抗 Rd のサポート
  - USB Power Delivery のメッセージの送受信
  - FRS（高速ロール･スワップ）Rx のサポート
- デジタル･コントローラ
  - BMC（Bi-phase Mark Coding）のエンコードとデコード
  - 4B5B 符号のエンコードとデコード
  - デバウンス有 USB Type-C レベル検出で割込みを生成
  - 割込みを生成する FRS 検出
  - USB Power Delivery ペイロード用のバイトレベルの DMA 互換インタフェース（割込みを生成）
  - USB Power Delivery クロックのプリスケーラ／分周回路
  - CRC の生成とチェック
  - オーダード・セットのサポート（プログラム可能な受信時のオーダード・セット・マスクを含む）
  - 受信 Rx ストリームからのクロック･リカバリ

46.3 UCPD の実装

このデバイスは USB Type-C ポートを 1 つサポートする UCPD コントローラを 1 つ搭載しています。

表 422. UCPD の実装(1)

UCPD の機能	UCPD1
外部信号 UCPDx_DBCC1 および UCPDx_DBCC2 による デッド･バッテリのサポート	X
オルタネート機能ピンとして使用できる UCPDx_FRSTX	X
全自動トリミング（ソフトウェアによるオーバーライド不要）	X
ディスクリート･コンポーネント PHY のサポート	-

1. “X” はサポートされています。“-” はサポートされていません。

46.4 UCPD の機能詳細

UCPD ペリフェラルは、USB Power Delivery 制御インタフェース仕様に対応するために専用設計された I/O を使用して、このインタフェースをサポートするハードウェアを提供します。

内蔵 PHY で Type-C の電圧レベルを直接検出し、Power Delivery BIST キャリア･モード 2（Tx のみ）、BIST テスト･データ（Tx と Rx）、Power Delivery の Rx FRS 信号処理をサポートします。

Power Delivery の FRS Tx 信号では、UCPD_FRSTX ピン（オルタネート機能）の設定によって外付けの NMOS トランジスタを制御するようにデバイスを設定できるので、CC ラインを低抵抗でプルダウンできます。

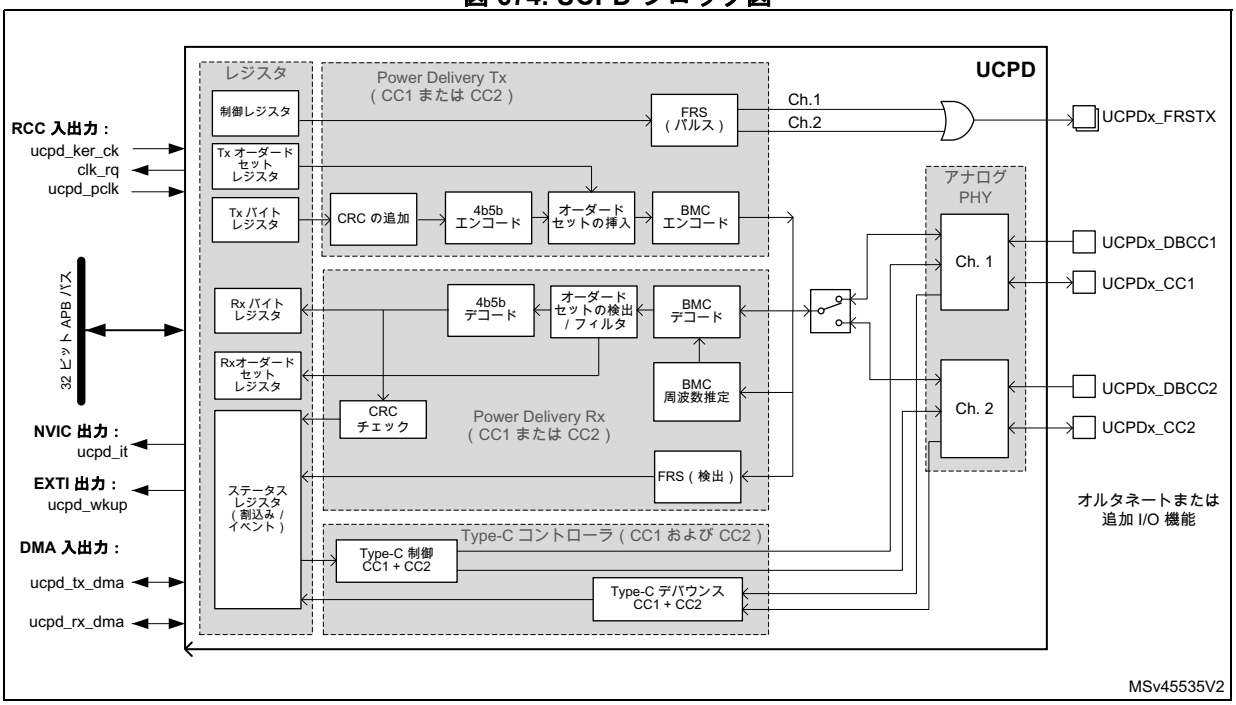
UCPD トランスミッタは、データを BMC（Bi-phase Mark Coding）エンコードして、Type-C コネクタの CC ラインに送信します。このデータとして、プリアンブル、SOP、プロトコル層からの 4B5B エンコード済みペイロード、CRC、EOP があります。自動的にフレーム間ギャップを挿入して「Hard Reset」を実行します。

UCPD レシーバは、SOP の検出、受信ストリームの BMC デコード、プリアンブルのリカバリ、ペイロード･データの 4B5B デコード、EOP 検出、CRC チェックを実行します。5 つの K コード SOP と 2 つのリセットオーダード・セットのほか、2 つのソフトウェア定義パターンを自動検出します（規格に従い、4 つの K コードのうち 3 つのみを適切に受信できます）。

受信した USB Power Delivery メッセージ と FRS 通知を検出する機能が STOP モードのペリフェラルでも維持されているので、低電力動作が可能です。

46.4.1 UCPD ブロック図

図 674. UCPD ブロック図



次の表に、外部信号の一覧を示します（オルタネート機能または追加の I/O 機能）

表 423. ピンの UCPD 信号

ピン名	信号タイプ	説明
UCPDx_FRSTX	出力	USB Type-C の高速ロール･スワップ（FRS）信号の制御（DRP のみに適用）。この信号（アクティブ･ハイ）は、アクティブな CC ラインをプルダウンする外付けの NMOS トランジスタを駆動します。代表的な適用例では、各 CC ラインに 1 つずつ合計 2 つの NMOS トランジスタを使用し、これらを駆動する独立した I/O を確保します。当初、これらの I/O はロー駆動の GPIO として設定されています。接続されたケーブルの向きが Type-C のステート･マシンを介して検出されると、アクティブな CC ラインがどちらであるかが決まるので、アクティブな CC ラインの I/O を UCPDx_FRSTX オルタネート機能に、非アクティブな CC ラインの I/O をロー駆動の GPIO に設定する必要があります。
UCPDx_CC1	入力／出力	USB Type-C 設定制御ライン 1。USB Type-C コネクタの CC1 端子に接続します。
UCPDx_CC2	入力／出力	USB Type-C 設定制御ライン 2。USB Type-C コネクタの CC2 端子に接続します。
UCPDx_DBCC1	入力	USB Type-C 設定制御ライン 1 のデッド･バッテリ信号。デッド･バッテリのサポートが必要な場合は USB Type-C コネクタの CC1 端子に接続します。
UCPDx_DBCC2	入力	USB Type-C 設定制御ライン 2 のデッド･バッテリ信号。デッド･バッテリのサポートが必要な場合は USB Type-C コネクタの CC2 端子に接続します。

次の表に、主要な内部信号を示します。

表 424. UCPD 内部信号

内部信号名	信号タイプ	説明
ucpd_pclk	入力	レジスタの APB クロック
ucpd_ker_ck	入力	カーネル･クロック
ucpd_tx_dma	入力／出力	Rx DMA 確認応答／リクエスト
ucpd_rx_dma	入力／出力	Tx DMA 確認応答／リクエスト
ucpd_it	出力	NVIC 接続用の割込みリクエスト（すべての割込みの OR）
ucpd_wkup	出力	EXTI に接続するウェイクアップ･リクエスト
clk_rq	出力	RCC に接続するクロック･リクエスト

46.4.2 UCPD のリセットおよびクロック

ペリフェラルにはリセット信号が 1 つあります（APB バス･リセット）。

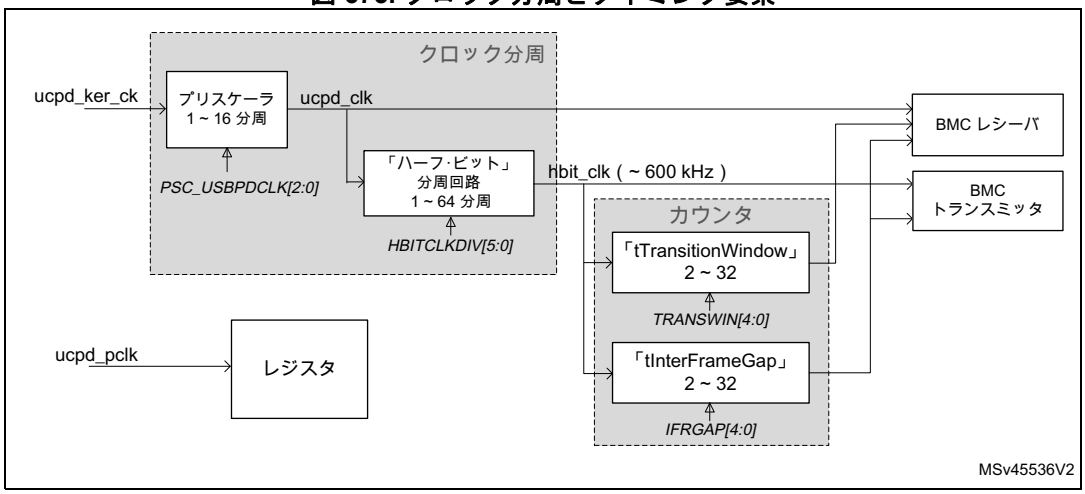
レジスタの領域は APB クロック（ucpd_pclk）で駆動されます。

トランスミッタの主な機能部分は、ucpd_ker_ck（HSI16）クロックをプリスケーラに通して得られる ucpd_clk クロックで動作します。プリスケーラの設定は、UCPD_CFGR1 レジスタの PSC_USBPDCLK[2:0] ビットフィールドに従います。レシーバの主な機能部分は、受信ビット･ストリームから抽出（リカバリ）される ucpd_rx_clk クロックで駆動されます。

レシーバは、クロック周波数 6 ~ 18 MHz で動作するように設計されています。ただし、最適な性能を得るには 9 ~ 18 MHz の範囲での動作とします。

次の図は、UCPD ペリフェラルのクロック供給とタイミング要素について示したものです。

図 675. クロック分周とタイミング要素



適切な遅延を設定するには、USB PD の仕様を参照してください。*tTransitionWindow* と、特に *tInterFrameGap* では、どのような場合でも指定のタイミングが得られるように、クロック周波数の不確実性を考慮する必要があります。

46.4.3 物理層のプロトコル

物理層は、USB Power Delivery 仕様の基礎となる信号送信をカバーします。

トランスミッタ側の主な機能は、定義されたパケット･フォーマットに従ってパケットを形成することです。このパケットを構成するデータは一般的に次のとおりです。

- プリアンブル
- パケット開始（SOP、オーダード・セット）
- ペイロード･ヘッダ
- ペイロード･データ
- 巡回冗長検査（CRC）情報
- パケット終了（EOP）

CC ラインに送出される前のデータ･ストリームは、指定されたタイミング制約を満足しながら、BMC エンコードされます。

受信側の主な処理は、以下のとおりです。

- パケット開始（SOP、オーダード・セット）情報の抽出
- ペイロード･ヘッダの抽出
- ペイロード･データの抽出
- CRC の受信とチェック
- パケット終了（EOP）の受信

基本的に受信は送信と逆の処理になるので、まずデータストリームを BMC デコードします。

シンボルのエンコード

プリアンブルを除き、すべてのシンボルは次表に示す仕様に従い 4B5B 方式でエンコードされます。

表 425. 4B5B シンボル･エンコード表

名前	4B	5B	シンボルの内容
0	0000	11110	16 進数データ 0
1	0001	01001	16 進数データ 1
2	0010	10100	16 進数データ 2
3	0011	10101	16 進数データ 3
4	0100	01010	16 進数データ 4
5	0101	01011	16 進数データ 5
6	0110	01110	16 進数データ 6
7	0111	01111	16 進数データ 7
8	1000	10010	16 進数データ 8
9	1001	10011	16 進数データ 9
A	1010	10110	16 進数データ A
B	1011	10111	16 進数データ B
C	1100	11010	16 進数データ C
D	1101	11011	16 進数データ D
E	1110	11100	16 進数データ E

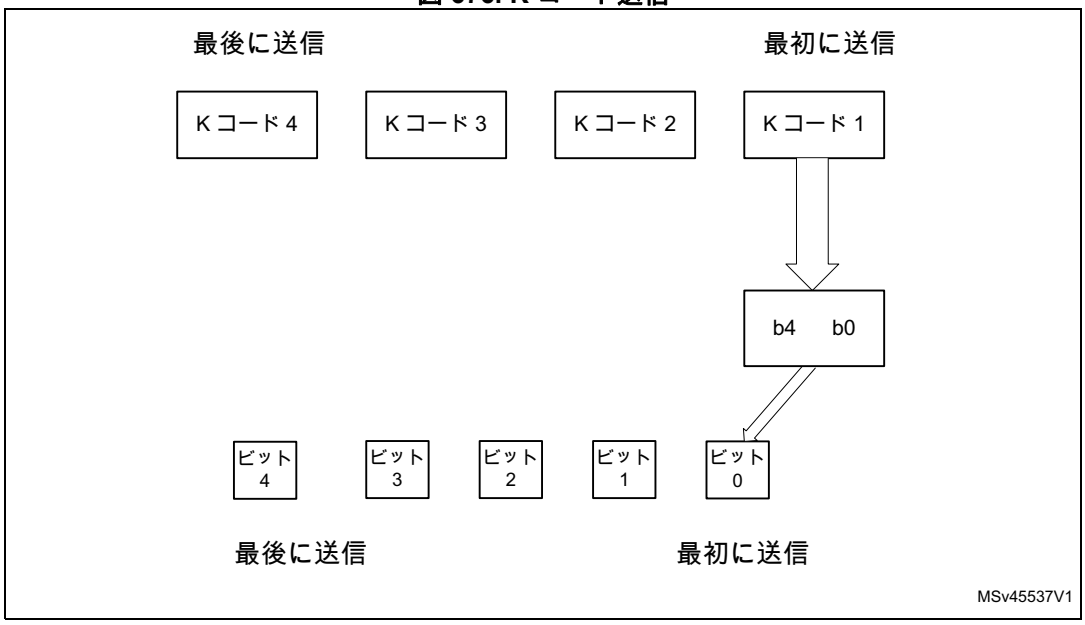
表 425. 4B5B シンボル･エンコード表 （続き）

名前	4B	5B	シンボルの内容
F	1111	11101	16 進数データ F
Sync-1	K コード	11000	Startsynch #1
Sync-2	K コード	10001	Startsynch #2
RST-1	K コード	00111	Hard Reset #1
RST-2	K コード	11001	Hard Reset #2
EOP	K コード	01101	EOP
予約済みです。	エラー	00000	使用不可
予約済みです。	エラー	00001	使用不可
予約済みです。	エラー	00010	使用不可
予約済みです。	エラー	00011	使用不可
予約済みです。	エラー	00100	使用不可
予約済みです。	エラー	00101	使用不可
Sync-3	K コード	00110	Startsynch #3
予約済みです。	エラー	01000	使用不可
予約済みです。	エラー	01100	使用不可
予約済みです。	エラー	10000	使用不可
予約済みです。	エラー	11111	使用不可

オーダード・セット

オーダード・セットは、次の図に示すように 4 つの K コードによって構成されます。

図 676. K コード送信



次の表に、可能なすべての SOP* シーケンスなど、定義されているオーダード・セットの一覧を示します。

物理層での Hard Reset は他のオーダード・セットよりも優先されるため、進行中の Tx メッセージが中断されることがあります。

表 426. オーダード・セット

オーダード・セット名	K コード #1	K コード #2	K コード #3	K コード #4
SOP	Sync-1	Sync-1	Sync-1	Sync-2
SOP'	Sync-1	Sync-1	Sync-3	Sync-3
SOP"	Sync-1	Sync-3	Sync-1	Sync-3
Hard Reset	RST-1	RST-1	RST-1	RST-2
Cable Reset	RST-1	Sync-1	RST-1	Sync-3
SOP'_Debug	Sync-1	RST-2	RST-2	Sync-3
SOP"_Debug	Sync-1	RST-2	Sync-3	Sync-2

受信時の物理層は、4 つの K コードのうち 3 つを任意に組み合わせたオーダード・セットを受け入れることができる必要があります。

表 427. オーダード・セットの検証

ステータス	1 番目のコード	2 番目のコード	3 番目のコード	4 番目のコード
有効	破損コード	K コード	K コード	K コード
有効	K コード	破損コード	K コード	K コード
有効	K コード	K コード	破損コード	K コード
有効	K コード	K コード	K コード	破損コード
有効（完全）	K コード	K コード	K コード	K コード
無効（例）	K コード	破損コード	K コード	破損コード

送信時のビット順序

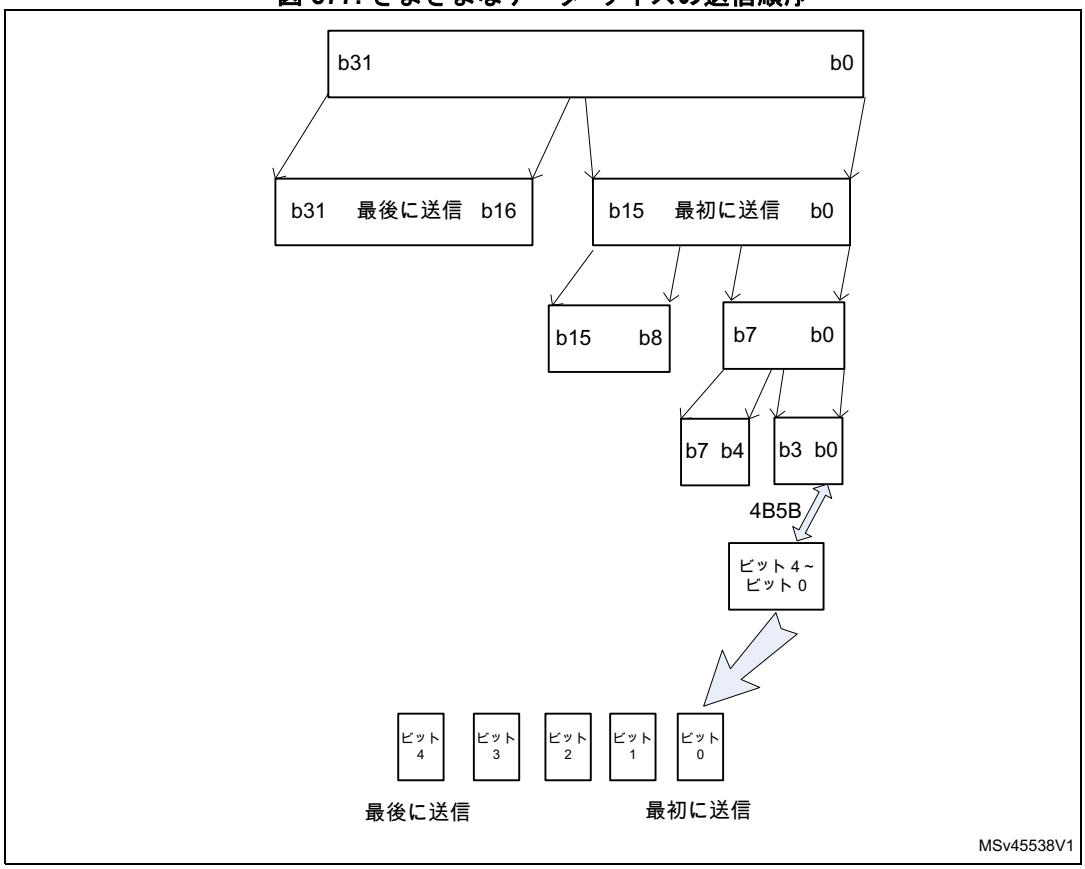
送信できるデータのユニットとサイズを次の表に示します。

表 428. データ･サイズ

データユニット	エンコード前	エンコード後
バイト	8 ビット	10 ビット
ワード	16 ビット	20 ビット
ダブルワード	32 ビット	40 ビット

次の図に送信順序を示します。

図 677. さまざまなデータ･サイズの送信順序

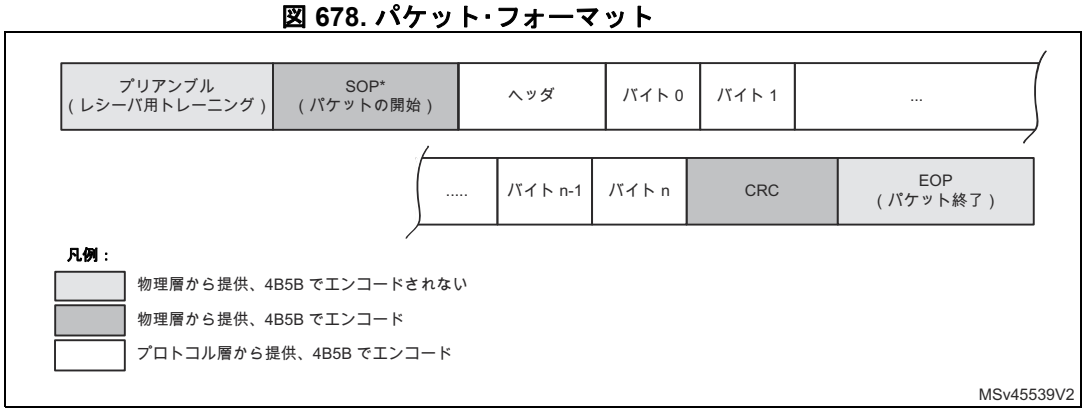


パケットフォーマット

Hard Reset と Cable Reset 以外のメッセージ

次の図に、パケットのフォーマットおよび 4B5B エンコードとデータ･ソースに関する情報を示します。

図 678. パケット･フォーマット



Hard Reset

物理層は Hard Reset 通知を他のタイプのメッセージとは異なる方法で処理します。この信号が、進行中の転送を中断できる高い優先度を持つからです。

物理層の仕様から、Tx メッセージの進行中には次のようなシーケンスが考えられます。

1. K コードの EOP を送信することでメッセージを終了し、残りのメッセージを破棄します。
2. *tInterFrameGap* の時間の経過を待ちます。
3. CC ラインがアイドル状態でない場合は、アイドル状態になるまで待ちます。
4. プリアンブルにつづいて Hard Reset 通知である 4 つの K コードを送信します。
5. CC チャネルを無効にし（送受信を停止）、物理層をリセットして、物理層がリセットされたことをプロトコル層に通知します。
6. プロトコル層からリクエストされた場合は、チャネルを再度有効にします。

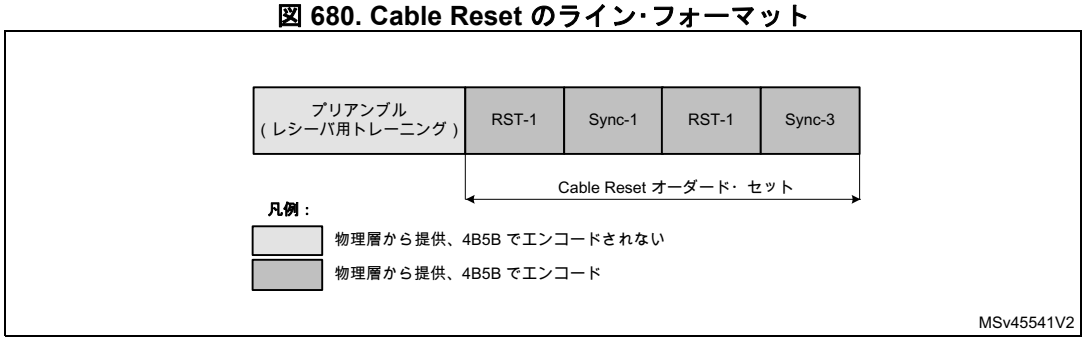
図 679. Hard Reset のライン･フォーマット



Cable Reset

次の図に示す Cable Reset は、フォーマットが Hard Reset に似ているものの、Hard Reset とは違い、特に優先して処理される必要はありません。

図 680. Cable Reset のライン･フォーマット



衝突の回避

物理層では、Tx メッセージの最後に送信されたビットの終了から次のメッセージの最初のビットの間に *tInterFrameGap* の遅延が置かれます。

また、送信開始前に CC ラインがアイドル状態であるかどうかが確認されます。CC ラインがアイドル状態と見なされるのは、*tTransitionWindow* の期間（12 ~ 20 µs）に発生する遷移が 2 回（*nTransitionCount*）以下の場合です。Power Delivery 仕様リビジョン 3.0 では、Rp の値（ソース側）を管理し、Rp の変更による Type-C 電圧レベル（シンク時）の変化を監視することも要件としています。

物理層の信号方式

ビットは BMC（Bi-phase Mark Coding）で送受信されます。

BIST

プロトコル層が要求する BIST アクションに応じて、以下のいずれかを実行できます。

- Tx BIST パターン･テスト。TXMODE および TXSEND を書き込むことで実現します。
- Rx BIST パターンテスト。RXBIST の正しい値に RXMODE を書き込むことで達成されます。

UCPD では、（「BMC」モードに対応する）次の 2 つのパターンをサポートできます。

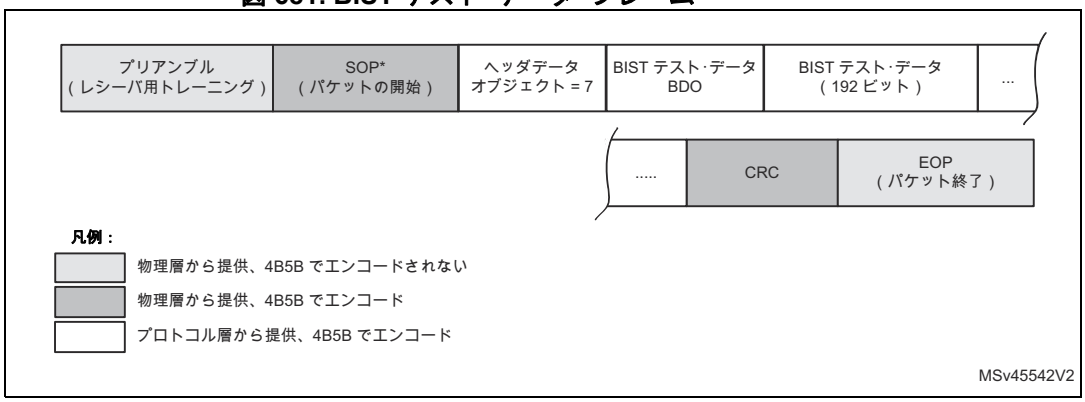
- BIST テストデータ（192 ビットパターン）。Tx および Rx に適用されます。Rx のテストではメッセージを受信します（ただし、プロトコル層には渡されず破棄されます。その場合でも、確認応答として GoodCRC Tx メッセージを生成する必要があります）。
- BIST キャリア･モード 2（単一パターン、無限長メッセージ）。Tx にのみ適用されます。Tx とは異なり、このモードのレシーバは、この状態にいる間、単に CC ラインを無視します。

BIST テスト･データ･パターン

テスト･データ･パターンは、UCPD では特殊なケースとみなされません。

BIST テスト･データのパケット･フレーム･フォーマットを次の図に示します。

図 681. BIST テスト･データ･フレーム



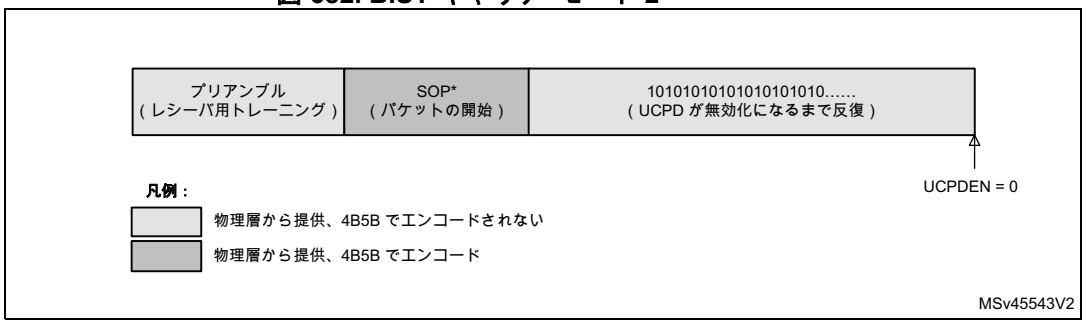
固定長のテスト･データ･パターンです。実際、図 678：パケット･フォーマットに示されている通常のパケットフォーマットとの違いを表す点はヘッダの内容のみです。UCPD は Tx ヘッダの内容をプログラミングによって受信するため（単にペイロードの一部とみなされている）、BIST テストデータパケットから通常のパケットを区別するのはこのプログラミングのみ（ブロックの動作ではない）です。

BIST キャリア･モード 2

必要に応じて、この BIST テスト･モードは、継続的に反復される 1010 の交番パターンを送信します。このモードは信号分析用であるため、一定の長さを持たない安定した状態です（USB PD 仕様の V1.0）。USB PD 仕様の V1.1 以降では、このモードを終了するタイミングを決めるカウンタをプロトコル層で定義します。

1010 の無限シーケンスを終了するには（USB PD 仕様の要件に従い）、UCPDEN ビットにより UCPD ペリフェラルを無効化します。

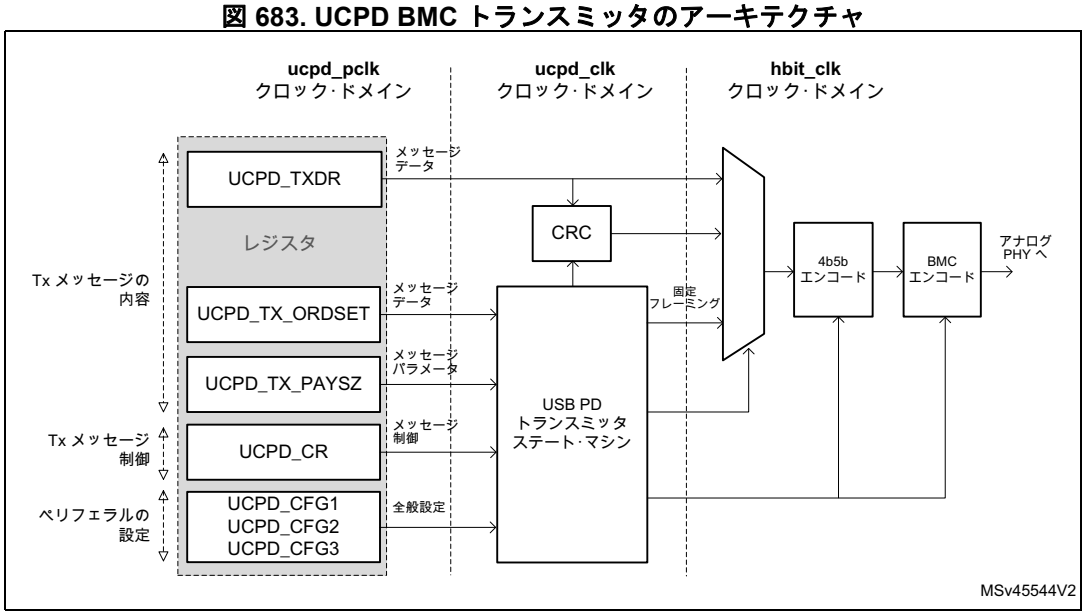
図 682. BIST キャリア･モード 2



46.4.4 UCPD BMC トランスミッタ

BMC トランスミッタには、次の図に示すように 4B5B エンコード、CRC 生成、BMC エンコードの機能があります。出力はチャネル･スイッチを介してアナログ PHY に送信されます。

図 683. UCPD BMC トランスミッタのアーキテクチャ



BMC エンコーダ

BMC（Bi-phase Mark Coding）の方法は、IEC 60958-1 Digital Audio Interface Part 1: General Edition 3.0 2008-09（www.iec.ch）に定義されています。

簡単な分周回路を使用して ucpd_clk からハーフビットのクロック hbit_clk が得られます。この分周回路は UCPD_CFGR1 レジスタの HBITCLKDIV[5:0] ビットフィールドで制御します。これによってハイとローの期間が等しいハーフビット周期と（立ち上がりエッジと立ち下がりエッジの期間のわずかな差異とジッタによる影響を無視した場合）、同じビット期間（ジッタを無視した場合）が確実に得られます。

トランスミッタのタイミングと衝突の回避

衝突の回避に対するハードウェアのサポートは、トランスミッタのハーフビット時間の機能の 1 つです。2 つのカウンタが実装されています。

- *tInterFrameGap* : IFRGAP で設定（事前定義値、変更可能）
- *tTransitionWindow* : TRANSWIN で設定（事前定義値、変更可能）

これら 2 つのカウンタを適切に設定すれば、フレーム間ギャップが生成されます。

トランスミッタにおける Hard Reset

Hard Reset の生成を容易にするために、TXMODE フィールドの特殊コードが使用されます。他のフィールドに書き込む必要はありません。

正しいコードを書き込むと、ハードウェアは実行中の Tx メッセージを基準とした適切な（最適な）タイミングで強制的に Hard Reset を実行します。現行のシーケンスを切り詰め、末尾に EOP K コード･シーケンスを直接付加することで、送信中のメッセージ（まだ進行中である場合）をクリーンに終了します。この切り詰めイベントに関連して特定の割込みが発生することはありません。

エラー時のトランスミッタの動作

アンダーラン状態（TXUND 割込み）が偶発的に発生する可能性があり、その場合、UCPD では（正しい）Tx ペイロードが不足して Tx メッセージを正しく完結できなくなります。これは重大なエラーです（これが生じるということは、ソフトウェアが適時応答できないということ）。このため、メッセージの最後で CRC が正しくないことを確認して、レシーバ側でメッセージが確実に破棄されるようにします。

46.4.5 UCPD BMC レシーバ

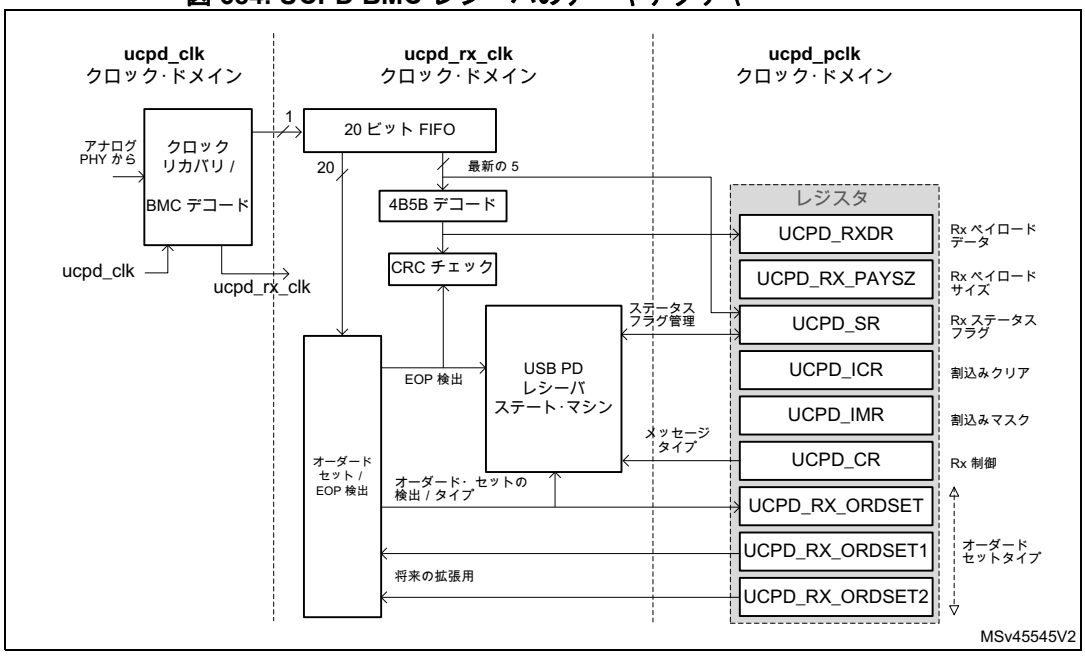
UCPD BMC レシーバは次の機能を実行します。

- クロック･リカバリ
- プリアンブル検出／タイミング導出
- BMC デコード
- 4b5b でのデコード
- K コードオーダード・セットの検出
- CRC チェック
- SOP 検出
- EOP 検出

レシーバは UCPD ペリフェラルが有効化され次第アクティブになりますが、メッセージの受信を試みる前に CC ラインがアイドル状態になるのを待ちます。

次の図に、UCPD BMC レシーバのアーキテクチャ概要を示します。

図 684. UCPD BMC レシーバのアーキテクチャ



CRC チェッカー

受信したビットは CRC チェッカーに供給され、受信したペイロードビットストリーム中に 32 ビットの状態が進展します。最後には、CRC の 32 ビットもロジックに供給されます。

EOP 検出（5 ビット）はプロセスを停止し、固定残差状態のチェックを行ってペイロードを正しく受信したことを確認します（実際には残差は 0xC704DD78 です）。

この時点で UCPD によって割込み RXMSGEND が発生します。CRC が不正だった場合は RXERR が真に設定され、受信データは破棄する必要があります。

CRC が正しく、正常動作であれば、RXMSGEND 割込みは確認応答済みであり、クリアされています。そうでない場合は別の割込み RXOVR が RXMSGEND の代わりに生成されます。

オーダード・セットの検出

この機能は、それぞれ 4 つの 5 ビット K コードから成るさまざまなオーダード·セットを検出します。

プリアンブルに入ったら、オーダード・セット（4 つの 5 ビット·ワード）のスライディング·ウィンドウ検出を開始します。

検出されたオーダード・セットは、すべての SOP*（SOP、SOP'、SOP''）コードのほかに、Hard Reset、Cable Reset、SOP'_Debug、SOP''_Debug、およびレジスタ USBPD1_RX_ORDEXT1 と USBPD1_RX_ORDEXT2 で定義される 2 つの拡張コードで構成されます。

EOP 検出および Hard Reset の例外処理

EOP はメッセージの終了を表す固定の 5 ビット K コードです。

トランスミッタに求められる Hard Reset の送信方法は、前のメッセージ送信がまだ進行中であれば、それを早い段階の EOP によって切り捨てることです。

Hard Resetが無視された場合は、EOP検知は所定の時間に発生するはずです。しかし、この Hard Rest の発行により、Rx メッセージの受信中は EOP 検出をアクティブな状態に維持する必要があります。「早い段階の EOP」が検出されると、切り捨てられた Rx メッセージは直ちに破棄されます。

切り捨てられたメッセージまたは破損したメッセージによる例外

オーダード・セットが検出されると、メッセージによっては CRC と EOP で終了するデータ･バイトを受信する場合があります。これらのフェーズのどこかでエラー条件が発生すると、次のような状況になります。

- 1 "UI" 周期を著しく上回る期間中はラインが静的になります（この条件が発生するための厳密な閾値は重要ではありませんが、3 UI が経過するより前に例外を生成する必要があります）。
- メッセージが終端に達しても認識されません（例：EOP の破損）。

どちらの場合も、レシーバは現在のメッセージを終了して、RXMSGEND フラグと RXERR フラグをセットします。

短いプリアンブルまたは不完全なオーダード・セットによる例外

レシーバで確認できるプリアンブル長が想定の半分未満である例外事象が発生すると、正しい BMC デコードに必要な周波数を推定できなくなります。プリアンブル全体を確認でき、周波数を推定できたとしても、ラインが静的になる前にすべてのオーダード・セットを受信できないと、レシーバのステート･マシンは開始されません。

どちらの場合もクロック･リカバリ／BMC デコーダを再起動し、最初に IDLE 条件、次にプリアンブルを確認し、続いて周波数を推定します。

46.4.6 UCPD Type-C のプルアップ抵抗（Rp）およびプルダウン抵抗（Rd）

UCPD では、これらの抵抗を ANAMODE および ANASUBMODE[1:0] によって簡単に制御できます。使用している CC ラインが 1 本のみの場合、CCENABLE[1:0] ビットフィールドの設定によって他方の CC ラインの抵抗に対する制御を無効化すると、消費電力を最適化できます。

マイクロコントローラに給電していない場合でも､UCPDx_DBCC1 ピンと UCPDx_DBCC2 ピンがそれぞれ UCPDx_CC1 ピンと UCPDx_CC2 ピンに接続されていれば､「デッド･バッテリ」Rd は有効な状態を維持しています。

デッド･バッテリ動作が必要ない場合は（ソース専用製品など）､UCPDx_DBCC1 ピンと UCPDx_DBCC2 ピンの両方をグランドに接続する必要があります。

電源が投入されてマイクロコントローラがブートした後､まず目的とする動作（たとえばソース）を､ANAMODE と ANASUBMODE[1:0] にプログラムしますデッド･バッテリのプルダウン抵抗を切り離し、プログラムした値が機能できるようにする、PWR_CR3 レジスタの UCPD1_DBDIS ビット。つづいて､PWR_CR3 レジスタの UCPD1_DBDIS ビットをセットしてデッド･バッテリ対応プルダウン抵抗を切り離し、プログラムした値を有効にします。

切断されている状態のシンクの場合は、STANDBY 低電力モードを使用できます。

46.4.7 UCPD Type-C 電圧監視とデバウンス

Type-C ステート･マシンが適切に動作して、ケーブルの方向を正しく検出できるようにするには、CC1 と CC2 ラインの電圧レベルを、ピークのような高速のイベントは無視して監視する必要があります。

CC1 と CC2 ラインの電圧レベル間の閾値は、PHY の閾値検出回路の設定で決まります。

TYPEC_VSTATE_CC1/2[1:0] ビットフィールドには、ピークなどの高速なライン･イベントを抑制するハードウェア･デバウンス･フィルタで処理された CC1 と CC2 ラインのレベルが反映されます。Power Delivery 信号処理に CC1 または CC2 ラインのどちらを使用するかは、PHYCCSEL ビットで選択します。

消費電力を最小限にするには、ポーリングによる方法を推奨します。この方法では、Type-C 検出回路を常時オンに維持するのではなく、ポーリング実行のときにのみオンにして、CC1 または CC2 ラインのイベントでデバイスを STOP モードからウェイクアップします。

46.4.8 UCPD 高速ロール･スワップ（FRS）信号および検出

FRS 信号

FRSTX ビットをセットした時点で FRS 条件（特定の長さのパルス）が生成されます。

FRS 条件が継続する間、UCPD_FRSTX（オルタネート機能）に設定された I/O が、アクティブな CC ラインをプルダウンする外付け NMOS トランジスタのゲートをハイ･レベルで制御します。

FRS 検出

検出されたケーブルの方向に応じてアクティブな CC ラインを選択する PHYCCSEL への書込みが終了した後、FRSRXEN ビットをセットすることで FRS 監視が有効になります。

46.4.9 UCPD DMA インタフェース

DMA は UCPD に実装され、有効にすると、USBPD1_TXDR および USBPD1_RXDR レジスタ（Tx と Rx のデータレジスタ、それぞれ 1 バイト）を処理するバイトレベルの割込みは不要になります。

TXDMAEN や RXDMAEN ビットを有効にすることで、Tx や Rx の機能に対して個別に DMA を有効にできます。

46.4.10 STOP モードからのウェイクアップ

消費電力を最適化するには、STOP モードを使用し、CC ラインでのイベントを待ってマイクロコントローラをウェイクアップする方法が有効です。

これが機能するためには、まず WUPEN に 1 を書き込んでウェイクアップを有効にする必要があります。

ウェイクアップを引き起こすイベントは次のとおりです。

- BMC レシーバでのイベント（RXORDDET、RXHRSTDET）、ハードウェアによる PHYRXEN ビットの有効化
- FRS 検出器でのイベント（FRSEVT）、ハードウェアによる FRSRXEN ビットの有効化
- Type-C 検出器でのイベント（TYPECEVT1、TYPECEVT2）、ハードウェアによる CC1TCDIS、CC2TCDIS ビットの有効化

46.4.11 UCPD プログラミング･シーケンス

UCPD ユニットを使用する通常のシーケンスは、次のとおりです。

1. UCPD を設定します。
2. UCPD を有効化します。
3. 以下を同時に実行します。
   - プロトコル層からの要求に応じて Tx メッセージを送信します。
   - 関連する Rx メッセージを捕捉し（ポーリングまたは割込み待ち）、プロトコル層に渡す詳細情報を復元します。

期限を指定せずに最後のステップを繰り返します。

初期化フェーズ

次のシーケンスを使用してクリーンに起動します。

1. UCPD_CFG レジスタに書き込むことで初期のクロック分周比をすべて設定します。
2. UCPDEN ビットをセットしてユニットを有効にします。

Type-C ステート･マシンの処理

ソース･ポート、シンク･ポート、またはデュアルロール･ポート（ソースとシンクとの最後の切り替わり）の一般的アプリケーションでは、対応する USB Type-C ステート･マシンをソフトウェアで実装する必要があります。基本的なコーディングを次の表に示します。

表 429. ANAMODE と ANASUBMODE のコーディングおよび TYPEC_VSTATE_CCx とのリンク

ANAMODE	ANASUBMODE[1:0]	注	TYPEC_VSTATE_CCx[1:0]			
			00	01	10	11
0：ソース	00：無効	無効	N/A			
	01：デフォルト USB Rp	-	vRa[Def]	vRd[Def]	vOPEN[Def]	N/A
	10：1.5A Rp	-	vRa[1.5]	vRd[1.5]	vOPEN[1.5]	
	11：3.0A Rp	-	vRa[3.0]	vRd[3.0]	vOPEN[3.0]	
1：シンク	xx	-	vRa	vRd-USB	vRd-1.5	vRd-3.0

CCENABLE[1:0] ビットフィールドにより、いずれかの CC ラインのプルアップ抵抗またはプルダウン抵抗を無効化できます。

注： **Type-C ステート･マシンは CC ラインのレベルだけでなく、VBUS の存在の検出にも依存します（シンク･モードの場合）。ソース･モードの場合は、VCONN の生成と VBUS の状態（オン、オフ、+ 電圧レベル、放電）が Type-C ステート･マシンで決まります。UCPD は VBUS の生成回路や VCONN 負荷スイッチ（CC ラインに接続する VCONN 電源ジェネレータを有効化）を直接制御しませんが、アプリケーションは正常に機能するためにこれらの入力や制御を必要とします。**

一般的なプログラミング･シーケンス（UCPD を設定後、有効化）

1. USB Type-C ステート･マシンでの現在の位置に基づいて（ソースの場合は電流レベルの通知にも基づいて）、ANAMODE、ANASUBMODE[1:0] を設定します。これにより、CC ライン上の適切なプルアップ抵抗またはプルダウン抵抗がオンになり、TYPEC_VSTATE フィールドが表す電圧レベルが定義されます。プログラミングの前には、PHY が実質的にオフである点に注意してください。
2. TYPEC_VSTATE_CC1/2 を読み込んで、Type-C の初期状態を決定します（ローカル･ソースがリモート･シンクに接続されているかどうかなど）。
3. 接続されていない場合は、接続イベントを待ちます。
4. 接続が検出され、ローカルの Power Delivery 機能が実装されたと仮定して、Power Delivery のメッセージの送受信を開始します。
5. 安定した電圧の変化を示す新しい割込み／イベントが PHYEVT1/2 に発生した場合は、その影響を再評価して、この入力を Type-C ステート･マシンに提供します。

3種類の Rp 値（Default-USB、1.5A、3.0A）を切り替える必要があるソースと、それに接続したシンクの場合：

- [ソース] 単純に ANASUBMODE[1:0] を再プログラムします。
- [その時点からのシンクの動作] PHYEVT1/2 が発生し、TYPEC_VSTATE1/2 がこれに合わせて変化します。

ソースからシンクに切り替わるデュアルロール･ポート（DRP）のプログラミング：

- 新しい動作を開始するように ANAMODE、ANASUBMODE[1:0] を再プログラムします。

詳細なプログラミング･シーケンス（例）：

表 430. Type-C シーケンス（ソース：3A）、ケーブル／シンク接続済み（CC1 に Rd、CC2 に Ra）

<table>
<tr><th>Type-C ステート</th><th>ANAMODE; ANASUBMODE [1:0]</th><th>CCENABLE</th><th>PHYCCSEL</th><th>RDCH</th><th>CC[x] VCONNEN(1)</th><th>イベント => 次の行に移動</th><th>コメント</th></tr>
<tr><td>Unattached. SRC</td><td rowspan="2">0：ソース、
11：Rp3A0</td><td rowspan="2">11：
両方有効</td><td>0（無視）</td><td rowspan="5">0：
[通常]</td><td rowspan="2">00：
[両方
非アクティブ]</td><td>PHYEVT1：
[VRd-3A0]</td><td>シンク接続検出待ち、CC1 [EVT1] に出現</td></tr>
<tr><td>Attachwait. SRC</td><td rowspan="4">0
[CC1 は Rd]</td><td>PHYEVT2：
[VRa]</td><td>接続待ち開始（100～200ms）、Ra => VCONN のリクエストも参照</td></tr>
<tr><td rowspan="4">Attached. SRC
[VCONN => CC2]</td><td>0：転送元、
11：Rp3A0
[SinkTxOK]</td><td rowspan="3">01：
CC2 無効
（外部 VCONN スイッチにより可能／推奨）</td><td rowspan="3">10：
[CC2
アクティブ]</td><td>タイマ（100 ms）、PHYEVTx なし</td><td>PHY から切断されたローカルの CC2（VCONN スイッチは VCONN ソースを外部的に CC2 に接続）
PHYEVT1 の監視を継続</td></tr>
<tr><td>0：転送元、
10：Rp1A5
[SinkTxNG]</td><td rowspan="2">SW タイマ（SinkTxNG）</td><td>ソースがメッセージシーケンスの初期化を希望（SinkTxNG 条件を最初にセット）</td></tr>
<tr><td rowspan="3">0：転送元、
11：Rp3A0
[SinkTxOK]</td><td>ソースがメッセージシーケンスを終了（その後 SinkTxOK 条件を設定）</td></tr>
<tr><td>Unattached wait. SRC</td><td>11：
両方有効</td><td>0：（無視）</td><td>1：
[放電]</td><td>00：
[両方非
アクティブ]</td><td>PHYEVT1：
[VOpen-3A0]</td><td>シンクの切断待ち（CC1 は Vopen）</td></tr>
<tr><td>Unattached. SRC</td><td>0：ソース、
11：Rp3A0</td><td>0：
[通常]</td><td>> 0.8V 検出（またはタイマ?）</td><td>VCONN 状態の特殊なソース（ECR 2016 年 4 月）：
能動的に VCONN [CC2] を放電 [Rdch]、< 0.8V まで</td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>[詳細は表の最初の行を参照]</td></tr>
</table>

1. 外付けの負荷切り替え部品を使用して 2 つの GPIO で VCONN を有効化します。

USB PD トランスミッタ

プロトコル層からメッセージ（送信するメッセージ）を受信したら、UCPD_TX_ORDSET レジスタと UCPD_TX_PAYSZ レジスタに書き込むことで Tx メッセージの内容を用意します。

TXMODE ビットフィールドに適切な値を設定したうえで、TXSEND ビットをセットするとメッセージの送信がトリガされます。

データ･バイトが送信されると、TXIS フラグがセットされて UCPD_TXDR レジスタへの新規データの書込みが要求されます。

この動作が、データのペイロード全体が送信されるまで繰り返されます。

CRC パケットが送信されると、メッセージ送信の完了を示す TXMSGSENT フラグがセットされます。

Hard Reset の送信

Hard Reset を送信する必要があることが判明した場合は、直ちに UCPD_CR レジスタの TXHRST ビットをセットして、内部ステート･マシンで強制的に適切なシーケンスを生成します。この特定のケースでは、UCPD_TX_ORDSET の値を更新する必要はありません（Hard Reset の適切なコードが UCPD によって送信されます）。

USB Power Delivery 仕様では、メッセージ送信が進行していても Hard Reset を優先することが求められます。この場合、たとえば UCPD で現在のメッセージのペイロードを切り捨てて、その末尾に EOP を付加します。レジスタからは何の通知（TXMSGSEND フラグなど）も得られません。この制約は、Hard Reset が、それより前のあらゆるアクティビティより優先されるという事実によって正当化されます（そのアクティビティが完了したかどうかを知ることは重要ではなくなるからです）。

送信用の DMA の使用

送信について DMA（Direct Memory Access）を有効にするには、UCPD_CR レジスタの TXDMAEN ビットをセットします。

各メッセージに対して：

- メッセージ全体をメモリ内で準備します（2 つのヘッダバイトから始まります）。
- データ･ワード数に 4 を乗じた値（バイト数）をヘッダの 2 バイトに加算した値に相当する長さで DMA 動作の長さをプログラムします。
- メッセージ転送を開始するには、TXSEND を書き込みます。
- TXMSGDISC の場合は、前のラインに戻ります（TXSEND）。
- DMA 転送完了割込みを待ちます（UCPD への最後の Tx バイトの書込みの完了）。
- 後続の TXMSGSENT 割込みが発生していないかダブルチェックします。

USB PD 受信

受信メッセージ･シーケンス開始の通知は、UCPD_SR（ビット RXORDDET）の割込みによってトリガされます。

この情報は、以下の読出しによって復元されます。

- UCPD_RX_SOP（RXORDDET 割込み時）
- UCPD_RXDR（割込み RXNE、バイトごとに繰り返される）
- UCPD_RXPAYSZ（割込み RXMSGEND）

以前に上記の UCPD_RXDR から読み出したデータは、RXERR フラグがセットされている場合、この時点で破棄する必要があります。

CRC が有効であれば、受信データがプロトコル層に転送されます。

デバッグ目的で、受信した不正な K コード数の統計情報を追跡することが望ましい場合があります（仕様で定義されているとおり、K コードの 3/4 が有効である場合にのみ実行します）。これは、以下のビットを使用すると容易に実現できます。

- 無効な K コードが少なくとも 1 つは存在することを示す RXSOP3OF4 ビット
- オーダード・セットに存在する無効な K コードの順序位置を特定する RXSOPKINVALID ビットフィールド

受信用の DMA の使用

受信について DMA（Direct Memory Access）を有効にするには、UCPD_CR レジスタの RXDMAEN ビットをセットします。

Rx メッセージの受信が想定される場合は以下を実行します。

- DMA 受信動作（および予備バッファ）を、受信する可能性がある最長のメッセージよりも若干長くプログラムします（長さは拡張メッセージ･サポートの有無に依存します）。
- RXORDDET を受信すると、DMA がバックグラウンドで動作を開始します。
- RXMSGEND 割込みの受信時に RXPAYSZ を読み出します。
- DMA Rx のバイト数との比較で RXPAYSZ をダブルチェックします（両方の値が一致する必要があります。ただし、RXDR で最後のバイトを取得してから DMA がRXDR を読み出すまでわずかな遅延があり、その間は一致しないことがあります）。
- DMA Rx バッファを処理します。
- 次の受信に備えて、できるだけ速やかに Rx DMA バッファの準備を整えます。

46.5 UCPD 低電力モード

低電力モードの概要を以下の表 431：低電力モードが UCPD に与える影響に示します。

表 431. 低電力モードが UCPD に与える影響

モード	説明
SLEEP	影響なし。
STOP	イベントの検出（Type-C、BMC Rx、FRS 検出）動作は継続するのでマイクロコントローラをウェイクアップできます。
STANDBY	UCPD は動作しておらず、MCU をウェイクアップすることはできません。 プルダウンは、設定されている場合はアクティブなままです。
電源オフ	デッド･バッテリのプルダウンはアクティブなままです。

UCPD は、次のいずれかの関連するイベントを認識したときに、マイクロコントローラを STOP モードからウェイクアップできます。

- いずれかの CC ラインの電圧範囲の変化に関連する Type-C イベント。TYPEC_VSTATE_CCx で確認できます。
- オーダード・セットが RXORDSETEN[8:0] のフィルタリング結果に一致する Power Delivery メッセージの受信。RXORDSET を読み出すことで確認できます。

STOP モードからのウェイクアップを有効にするには、UCPD_CFG2 レジスタの WUPEN ビットをセットします。

UCPD レベルでは、カーネル･クロックのアクティビティを必要とする次の 3 種類のイベントが STOP モード中に発生する可能性があります。

- アナログ PHY 電圧閾値検出器でのアクティビティ。Type-C 仕様で定義された電圧範囲に収まる安定した変化であるかどうかが後で確認されることがあります。
- Power Delivery BMC レシーバでのアクティビティ（選択した CC ラインからの入力によるアクティビティ）。Rx メッセージ･イベント（RXORDSET）が後で生成されることがあります。
- Power Delivery FRS 検出器でのアクティビティ。FRS 信号検出イベント（FRSEVT）が後で生成されることがあります。

以下の回路や信号に対する非同期アクティビティが発生する場合、RCC で適切に機能するように、クロック･リクエスト信号が有効になります（WUPEN の条件によります）。

- Type-C 電圧閾値検出器（いずれかの CC ラインを入力とします）
- Power Delivery レシーバ信号（選択した CC ラインを入力とします）
- FRS 検出信号（選択した CC ラインを入力とします）

46.6 UCPD 割込み

下表は、UCPD のイベント･フラグの一覧です。関連するフラグ･クリア･ビットと割込みイネーブル･ビットも示しています。

表 432. UCPD 割込みリクエスト

割込みイベント	イベントフラグ	イベントフラグ／割込みのクリア方法	割込み有効制御ビット
FRS 検出	FRSEVT	FRSEVTCF をセットする	FRSEVTIE
Type-C 電圧レベル変化イベント（CC2）	TYPECEVT2	TYPECEVT2CF をセットする	TYPECEVT2IE
Type-C 電圧レベル変化イベント（CC1）	TYPECEVT1	TYPECEVT1CF をセットする	TYPECEVT1IE
Rx メッセージ受信	RXMSGEND	RXMSGENDCF をセットする	RXMSGENDIE
Rx データ･オーバーフロー	RXOVR	RXOVRCF をセットする	RXOVR
Rx Hard Reset 検出	RXHRSTDET	RXHRSTDETCF をセットする	RXHRSTDETIE
Rx オーダード・セット（4 つの K コード）検出	RXORDDET	RXORDDETCF をセットする	RXORDDETIE
受信データ･レジスタ･ノット･エンプティ	RXNE	UCPD_RXDR のデータを読み出す	RXNEIE
Tx データ･アンダーラン	TXUND	TXUNDCF をセットする	TXUNDIE
Hard Reset 送信	HRSTSENT	HRSTSENTCF をセットする	HRSTSENTIE
Hard Reset の破棄	HRSTDISC	HRSTDISCCF をセットする	HRSTDISCIE
送信メッセージ･アボート	TXMSGABT	TXMSGABTCF をセットする	TXMSGABTIE
送信メッセージ送信	TXMSGSENT	TXMSGSENTCF をセットする	TXMSGSENTIE

表 432. UCPD 割込みリクエスト （続き）

割込みイベント	イベントフラグ	イベントフラグ／割込みのクリア方法	割込み有効制御ビット
送信メッセージ破棄	TXMSGDISC	TXMSGDISCCF をセットする	TXMSGDISCIE
送信データ要求	TXIS	UCPD_TXDR レジスタにデータを書き込む	TXISIE

UCPD からの割込みが受信された場合、ソフトウェアで UCPD_SR レジスタを読み出すことで割込みの発行元を確認する必要があります。

どのビットが 1 であるかによって、ISR はこの条件を処理し、UCPD_ICR レジスタの該当するビットに書き込んで、この 1 であるビットをクリアする必要があります。

46.7 UCPD レジスタ

46.7.1 UCPD 設定レジスタ 1（UCPD_CFGR1）

アドレス･オフセット：0x000

リセット値：0x0000 0000

ペリフェラルの全般設定です。このレジスタへの書込みは UCPD が無効（UCPDEN = 0）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UCPDEN	RXDMAEN	TXDMAEN	RXORDSETEN[8:0]									PSC_USBPDCLK[2:0]			Res.
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRANSWIN[4:0]					IFRGAP[4:0]					HBITCLKDIV[5:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **UCPDEN**：UCPD ペリフェラル･イネーブル

UCPD ペリフェラルの全般的な有効化。

0：無効化

1：イネーブル

無効化したペリフェラルは、実行中のアクティビティをただちに終了し、すべての制御ビットとビットフィールドはデフォルトのリセット値に設定されます。これらの各ビットは、ペリフェラルが無効から有効な状態に遷移するたびに、必要な値に設定する必要があります。

ビット 30 **RXDMAEN**：受信 DMA モード･イネーブル

このビットをセットすると、受信用の DMA モードが有効になります。

0：無効化

1：イネーブル

ビット 29 **TXDMAEN**：送信 DMA モード･イネーブル

このビットをセットすると、送信用の DMA モードが有効になります。

0：無効化

1：イネーブル

ビット 28:20 **RXORDSETEN[8:0]**：レシーバのオーダード・セット検出イネーブル

このビットフィールドは、レシーバで検出する必要があるオーダード・セットのタイプを決定します。各ビットをセットまたはクリアすると、そのビットに対応する機能がそれぞれ有効または無効になります。

0bxxxxxxxx1：SOP 検出有効
0bxxxxxxx1x：SOP' 検出有効
0bxxxxxx1xx：SOP'' 検出有効
0bxxxxx1xxx：Hard Reset 検出有効
0bxxxx1xxxx：Cable Detect リセット有効
0bxxx1xxxxx：SOP'_Debug 有効
0bxx1xxxxxx：SOP''_Debug 有効
0bx1xxxxxxx：SOP 拡張 #1 有効
0b1xxxxxxxx：SOP 拡張 #2 有効

ビット 19:17 **PSC_USBPDCLK[2:0]**：ucpd_clk を生成するためのプリスケーラの分周比

このビットフィールドは、UCPD ペリフェラル･クロック（ucpd_clk）を生成するための、カーネル･クロックに対するプリスケーラの分周比を決定します。

0x0：1（バイパス）
0x1：2
0x2：4
0x3：8
0x4：16

ucpd_clk の周波数が 6 ~ 9 MHz の範囲になるようにプリスケーラを設定することを推奨します。

ビット 16 予約済みであり、リセット値に保持する必要があります。

ビット 15:11 **TRANSWIN[4:0]**：遷移ウィンドウの期間

このビットフィールドは、*tTransitionWindow* の時間間隔を生成する hbit_clk の分周回路の分周比（ビットフィールドの値から 1 を減算した値）を決定します。

0x00：サポートされません
0x01：2
0x09：10 （推奨）
0x1F：32

ucpd_clk 周波数と HBITCLKDIV[5:0] ビットフィールドの設定を考慮して、12 ~ 20 µs の時間間隔が生成される値を設定します。

ビット 10:6 **IFRGAP[4:0]**：フレーム間ギャップ･タイマ･クロックを生成するための分周比

このビットフィールドは、フレーム間ギャップ･タイマ･クロック（*tInterFrameGap*）を生成する ucpd_clk の分周回路の分周比（ビットフィールドの値から 1 を減算した値）を決定します。

0x00：サポートされません
0x01：2
0x0D：14
0x0E：15
0x0F：16
0x1F：32

分周比 15 は、Tx クロックを USB PD 2.0 仕様の公称値とする場合に適用する値です。Tx クロックを公称値よりも遅くする場合は 15 未満の分周比、公称値よりも速くする場合は 15 より大きい分周比をそれぞれ適用します。

ビット 5:0 **HBITCLKDIV[5:0]**：ハーフビット･クロック生成のための分周比

このビットフィールドは、ハーフビット･クロック（hbit_clk）を生成する ucpd_clk の分周回路の分周比（ビットフィールドの値に 1 を加算した値）を決定します。

0x00：1（バイパス）
0x1A：27
0x3F：64

46.7.2 UCPD 設定レジスタ 2（UCPD_CFGR2）

アドレス･オフセット：0x004

リセット値：0x0000 0000

UCPD Rx 信号フィルタの設定このレジスタへの書込みは UCPD が無効（UCPDEN = 0）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPEN	FORCECLK	RXFILT2N3	RXFILTDIS
												rw	rw	rw	rw

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **WUPEN**：STOP モードからのウェイクアップ有効

このビットをセットすると、UCPD_ASYNC_INT 信号が有効になります。

0：無効化

1：イネーブル

ビット 2 **FORCECLK**：ClkReq クロック･リクエストの強制

0：クロック･リクエストを強制しません。

1：クロック･リクエストを強制します。

ビット 1 **RXFILT2N3**：BMC デコーダの Rx プレフィルタでのサンプリング方法

新しい値の確認前に取得する、一貫性のある連続サンプルの数。

0：3 サンプル

1：2 サンプル

ビット 0 **RXFILTDIS**：BMC デコーダの Rx プレフィルタ･イネーブル

0：イネーブル

1：無効化

レシーバのサンプリング･クロックを使用します（プリスケーラ適用後のクロック）。

46.7.3 UCPD 制御レジスタ（UCPD_CR）

アドレス･オフセット：0x00C

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが有効（UCPDEN = 1）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2TCDIS	CC1TCDIS	Res.	RDCH	FRSTX	FRSRXEN
										rw	rw		rw	rs	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	CCENABLE[1:0]		ANAMODE	ANASUBMODE[1:0]		PHYCCSEL	PHYRXEN	RXMODE	TXHRST	TXSEND	TXMODE[1:0]	
				rw	rw	rw	rw	rw	rw	rw	rw	rs	rs	rw	rw

ビット 31:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 **CC2TCDIS**：CC2 の Type-C 検出器ディスエーブル

このビットは、CC2 ラインの Type-C 検出器を無効化します。

0：イネーブル

1：無効化

有効化した CC2 の Type-C 検出器は、ANAMODE と ANASUBMODE[1:0] で設定します。

ビット 20 **CC1TCDIS**：CC1 の Type-C 検出器ディスエーブル

このビットは、CC1 ラインの Type-C 検出器を無効化します。

0：イネーブル

1：無効化

有効化した CC1 の Type-C 検出器は、ANAMODE と ANASUBMODE[1:0] で設定します。

ビット 19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **RDCH**：Rdch 条件の駆動

このビットをセットすると、PHYCCSEL ビットで選択したCC ライン（したがって VCONN に関連付けられた CC ライン）に対して Rdch 条件が適用されます。Type-C ステート･マシンの状態に準拠するために、ソースのみの *UnattachedWait.SRC* 状態の間はこの設定を維持することで、この適用が実現します。「ソース VCONN 放電のための USB Type-C ECN」を参照してください。CCENABLE[1:0] ビットフィールドも適宜設定する必要があります。

0：影響なし。

1：Rdch 条件を駆動します。

ビット 17 **FRSTX**：FRS Tx 通知イネーブル

このビットをセットすると FRS Tx 通知が有効になります。

0：影響なし。

1：イネーブル

このビットは、USB Power Delivery 仕様リビジョン 3.0 に従った遅延の後にハードウェアによってクリアされます。

ビット 16 **FRSRXEN**：FRS イベント検出イネーブル

このビットをセットすると、PHYCCSEL ビットで選択した CC ラインでの FRS Rx イベント（FRSEVT）検出が有効になります。0：無効化

1：イネーブル

FRS に対応していないソースまたはシンクにデバイスを接続する場合は、このビットをクリアします。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 予約済みであり、リセット値に保持する必要があります。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11:10 **CCENABLE[1:0]**：CC ライン･イネーブル

このビットフィールドは、ANAMODE と ANASUBMODE[1:0] の設定に従って CC1 ラインと CC2 ラインのアナログ PHY（プルアップとプルダウン）を有効化します。

0x0：両方の PHY を無効化
0x1：CC1 の PHY を有効化
0x2：CC2 の PHY を有効化
0x3：CC1 と CC2 の PHY を有効化

たとえば、一方のラインを外付けの VCONN スイッチを介して VCONN で駆動する場合に、もう一方のラインの PHY のみを有効化できます。両方の PHY を有効化するのは、シンクとソースの通常の使用法です。

ビット 9 **ANAMODE**：アナログ PHY の動作モード

0：ソース
1：シンク

CC1 と CC2 の使用方法は CCENABLE で決まります。このビットフィールドと ANASUBMODE[1:0] の組み合わせの効果については、表 429：ANAMODE と ANASUBMODE のコーディングおよび TYPEC_VSTATE_CCx とのリンクを参照してください。

ビット 8:7 **ANASUBMODE[1:0]**：アナログ PHY のサブモード

このビットフィールドの効果については、表 429：ANAMODE と ANASUBMODE のコーディングおよび TYPEC_VSTATE_CCx とのリンクを参照してください。

ビット 6 **PHYCCSEL**：USB Power Delivery 信号に使用する CC1/CC2 ラインの選択

0：Power Delivery 通信で CC1 IO を使用します。
1：Power Delivery 通信で CC2 IO を使用します。

この選択は、接続時に検出されるケーブルの方向によって決まります。

ビット 5 **PHYRXEN**：USB Power Delivery レシーバ･イネーブル

0：無効化
1：イネーブル

このビットをクリアすると、CC1 と CC2 両方のレシーバが無効になります。このビットをセットすると、PHYCCSEL ビットで選択した CC ラインのレシーバのみが有効になります。

ビット 4 **RXMODE**：レシーバ･モード

レシーバのモードを決定します。

0：通常受信モード
1：BIST 受信モード（BIST テスト･データ･モード）

このビットをセットすると、RXORDSET は通常どおりに動作します。RXDR はバイトを受信しなくなりますが、通常のメッセージに対しては CRC チェックが継続します。このモードでは、メッセージ ID が記載されているヘッダが受信されません。したがって、このテストであっても送信する必要がある GoodCRC 確認応答にメッセージ ID を記述するために、受け取ったメッセージ ID のカウンタをソフトウェア側で自動インクリメントする必要があります。

ビット 3 **TXHRST**：Tx Hard Reset を送信するコマンド

0：影響なし。
1：Tx Hard Reset メッセージを開始します。

このビットは、メッセージの送信が開始されるか破棄されると、ハードウェアによってただちにクリアされます。

ビット 2 **TXSEND**：Tx パケットを送信するコマンド

0：影響なし。

1：Tx パケット送信を開始します。

このビットは、パケットの送信が開始されるか破棄されると、ハードウェアによってただちにクリアされます。

ビット 1:0 **TXMODE[1:0]**：Tx パケットのタイプ

このビットフィールドへの書込みにより、その値に応じて次のような動作がトリガされます。

0x0：他のレジスタであらかじめ定義された Tx パケットの送信

0x1：Cable Reset シーケンス

0x2：BIST テスト･シーケンス（BIST キャリア･モード 2）

その他：無効

USB PD 仕様の V1.1 以降では、BIST キャリア･モード 2 の期間をカウントするカウンタが定義されています。"tBISTContMode" の遅延後にこのモードを適切に終了するには、ペリフェラルを無効化します（UCPDEN = 0）。

46.7.4 UCPD 割込みマスク･レジスタ（UCPD_IMR）

アドレス･オフセット：0x010

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが有効（UCPDEN = 1）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTIE	Res.	Res.	Res.	Res.
											r				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2IE	TYPECEVT1IE	Res.	RXMSGENDIE	RXOVRIE	RXHRSTDETIE	RXORDDETIE	RXNEIE	Res.	TXUNDIE	HRSTSENTIE	HRSTDISCIE	TXMSGABTIE	TXMSGSENTIE	TXMSGDISCIE	TXISIE
rw	rw		rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **FRSEVTIE**：FRSEVT 割込みイネーブル

0：無効化

1：イネーブル

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TYPECEVT2IE**：TYPECEVT2 割込み有効

0：無効化

1：イネーブル

ビット 14 **TYPECEVT1IE**：TYPECEVT1 割込み有効

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **RXMSGENDIE**：RXMSGEND 割込みイネーブル

0：無効化

1：イネーブル

ビット 11 **RXOVRIE**：RXOVR 割込みイネーブル
0：無効化
1：イネーブル

ビット 10 **RXHRSTDETIE**：RXHRSTDET 割込みイネーブル
0：無効化
1：イネーブル

ビット 9 **RXORDDETIE**：RXORDDET 割込みイネーブル
0：無効化
1：イネーブル

ビット 8 **RXNEIE**：RXNE 割込みイネーブル
0：無効化
1：イネーブル

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TXUNDIE**：TXUND 割込みイネーブル
0：無効化
1：イネーブル

ビット 5 **HRSTSENTIE**：HRSTSENT 割込みイネーブル
0：無効化
1：イネーブル

ビット 4 **HRSTDISCIE**：HRSTDISC 割込みイネーブル
0：無効化
1：イネーブル

ビット 3 **TXMSGABTIE**：TXMSGABT 割込みイネーブル
0：無効化
1：イネーブル

ビット 2 **TXMSGSENTIE**：TXMSGSENT 割込みイネーブル
0：無効化
1：イネーブル

ビット 1 **TXMSGDISCIE**：TXMSGDISC 割込みイネーブル
0：無効化
1：イネーブル

ビット 0 **TXISIE**：TXIS 割込みイネーブル
0：無効化
1：イネーブル

46.7.5 UCPD ステータス･レジスタ（UCPD_SR）

アドレス･オフセット：0x014

リセット値：0x0000 0000

これらのフラグ（シングルビットのステータス･ビットフィールド）は、割込みリクエストに関連付けられています。UCPD_IMR レジスタの該当ビットで割込みが有効になっていると、割込みが生成されます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVT	TYPEC_VSTATE_CC2[1:0]		TYPEC_VSTATE_CC1 [1:0]	
											r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2	TYPECEVT1	RXERR	RXMSGEND	RXOVR	RXHRSTDET	RXORDDET	RXNE	Res.	TXUND	HRSTSENT	HRSTDISC	TXMSGABT	TXMSGSENT	TXMSGDISC	TXIS
r	r	r	r	r	r	r	r		r	r	r	r	r	r	r

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **FRSEVT**：FRS 検出イベント

このフラグは FRSEVTCF ビットをセットすることでクリアされます。

0：新規イベントはありません。

1：新規の FRS 受信イベントが発生しました。

ビット 19:18 **TYPEC_VSTATE_CC2[1:0]**：CC2 ラインの電圧レベル

このステータス･ビットフィールドは、定常状態にある CC2 ラインの電圧レベルを示します。

0x0：最低

0x1：低

0x2：高

0x3：最高

PHY での BMC 変調に起因して USB PD メッセージ送信中の CC2 ラインに発生する電圧変動は、このビットフィールドの値には影響を及ぼしません。

ビット 17:16 **TYPEC_VSTATE_CC1[1:0]**：

このステータス･ビットフィールドは、定常状態にある CC1 ラインの電圧レベルを示します。

0x0：最低

0x1：低

0x2：高

0x3：最高

PHY での BMC 変調に起因して USB PD メッセージ送信中の CC1 ラインに発生する電圧変動は、このビットフィールドの値には影響を及ぼしません。

ビット 15 **TYPECEVT2**：CC2 ラインでの Type-C 電圧レベル･イベント

このフラグは、TYPEC_VSTATE_CC2[1:0] ビットフィールドの値の変化を示し、これは新しい Type-C イベントに対応しています。TYPECEVT2CF ビットをセットすることによってクリアされます。

0：新規イベントはありません。

1：新規の Type-C イベントが発生しました。

ビット 14 **TYPECEVT1**：CC1 ラインでの Type-C 電圧レベル･イベント

このフラグは、TYPEC_VSTATE_CC1[1:0] ビットフィールドの値の変化を示し、これは新しい Type-C イベントに対応しています。TYPECEVT2CF ビットをセットすることによってクリアされます。

0：新規イベントはありません。

1：新規の Type-C イベントが発生しました。

ビット 13 **RXERR**：受信メッセージ･エラー

このフラグは、CRC の誤りや切り捨てられたメッセージなど、RXMSGEND によって宣言された最後の Rx メッセージに発生したエラーを示します。（ラインは EOP を受信するまで静的な状態になります）。RXMSGEND フラグがセットされるたびにアサートされます。

0：0：エラーは検出されていません。

1：エラー（複数可）が検出されました。

ビット 12 **RXMSGEND**：Rx メッセージ受信

このフラグは、CRC の値に関わらず、メッセージ（Hard Reset を除く）が受信されたかどうかを示します。RXMSGENDCF ビットをセットすることでクリアされます。

0：新規 Rx メッセージは受信されていません。

1：新規 Rx メッセージを受信しました。

RXMSGEND フラグがハイに遷移したときに RXERR フラグがセットされた場合、最後に受信したメッセージにエラーがあることを示しています。

ビット 11 **RXOVR**：Rx データ･オーバーフローの検出

このフラグは、Rx データ･バッファがオーバーフローしたことを示します。RXOVRCF ビットをセットすることによってクリアされます。

0：オーバーフローなし

1：オーバーフロー

受信データを十分な速度で読み出すことができなかった場合に、バッファ･オーバーフローが発生することがあります。

ビット 10 **RXHRSTDET**：Rx Hard Reset 受信の検出

このフラグは、有効な Hard Reset メッセージを受信したことを示します。RXHRSTDETCF ビットをセットすることによってクリアされます。

0：Hard Reset は受信されていません。

1：Hard Reset を受信しました。

ビット 9 **RXORDDET**：Rx オーダード・セット（4 つの K コード）検出

このフラグは、オーダード・セットを検出したことを示します。関連する情報は、UCPD_RX_ORDSET レジスタの RXORDSET[2:0] ビットフィールドに格納されます。RXORDDETCF ビットをセットすることによってクリアされます。

0：オーダード・セットは検出されていません。

1：新規のオーダード・セットを検出しました。

ビット 8 **RXNE**：受信データ･レジスタ･ノット･エンプティの検出

このフラグは、UCPD_RXDR レジスタが空ではないことを示します。UCPD_RXDR を読み出すと自動的にクリアされます。

0：Rx データ･レジスタは空です。

1：Rx データ･レジスタは空ではありません。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TXUND**：Tx データ･アンダーランの検出

このフラグは、メッセージ送信を正常に実行するために必要な時間内に Tx データ･レジスタ（UCPD_TXDR）への書き込みが完了しなかったことを示します。TXUNDCF ビットをセットすることによってクリアされます。

0：Tx データ･アンダーランは検出されていません。

1：Tx データ･アンダーランが検出されました。

ビット 5 **HRSTSENT**：Hard Reset メッセージの送信

このフラグは、Hard Reset メッセージを送信したことを示します。HRSTSENTCF ビットをセットすることでクリアされます。

0：Hard Reset メッセージを送信していません。

1：Hard Reset メッセージを送信しました。

ビット 4 **HRSTDISC**：Hard Reset の破棄

このフラグは、Hard Reset メッセージが破棄されたことを示します。HRSTDISCCF ビットをセットすることでクリアされます。

0：Hard Reset は破棄されていません。

1：Hard Reset を破棄しました。

ビット 3 **TXMSGABT**：送信メッセージのアボート

このフラグは、Tx メッセージの送信中に発生した Hard Reset メッセージ送信リクエストが優先されたために、Tx メッセージがアボートされたことを示します。TXMSGABTCF ビットをセットすることによってクリアされます。

0：アボートされた送信メッセージはありません。

1：送信メッセージがアボートされました。

ビット 2 **TXMSGSENT**：メッセージの送信完了

このフラグは、パケット送信が完了したことを示します。TXMSGSENTCF ビットをセットすることによってクリアされます。

0：送信が完了した Tx メッセージはありません。

1：Tx メッセージの送信が完了しました。

メッセージ送信が Hard Reset によって中断された場合、このフラグはセットされません。

ビット 1 **TXMSGDISC**：メッセージ送信の破棄

このフラグは、メッセージ送信が破棄されたことを示します。TXMSGDISCCF ビットをセットすることでクリアされます。

0：破棄された Tx メッセージはありません。

1：Tx メッセージが破棄されました。

メッセージ送信は、同時に進行中の受診がある場合、またはラインに過剰なノイズが存在する場合に破棄されることがあります。Tx メッセージの破棄後、CC ラインがアイドル状態になると、このフラグがセットされます。

ビット 0 **TXIS**：送信割込みステータス

このフラグは、UCPD_TXDR レジスタが空で、新たなデータの書込みが必要であることを示します（送信したデータの量が、TXPAYSZ ビットフィールドで定義されたペイロード･サイズに達していないからです）。UCPD_TXDR レジスタにデータを書き込むとクリアされます。

0：新規の Tx データ書込みは不要です。

1：新規の Tx データ書込みが必要です。

46.7.6 UCPD 割込みクリア･レジスタ（UCPD_ICR）

アドレス･オフセット：0x018

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが有効（UCPDEN = 1）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTCF	Res.	Res.	Res.	Res.
											w				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TYPECEVT2CF	TYPECEVT1CF	Res.	RXMSGENDCF	RXOVRCF	RXHRSTDETCF	RXORDDETCF	Res.	Res.	TXUNDCF	HRSTSENTCF	HRSTDISCCF	TXMSGABTCF	TXMSGSENTCF	TXMSGDISCCF	Res.
w	w		w	w	w	w			w	w	w	w	w	w	

ビット 31:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **FRSEVTCF**：FRS イベントフラグ（FRSEVT）クリア
このビットをセットすると、UCPD_SR レジスタの FRSEVT フラグがクリアされます。

ビット 19:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **TYPECEVT2CF**：Type C CC2 ライン･イベント･フラグ（TYPECEVT2）クリア
このビットをセットすると、UCPD_SR レジスタの TYPECEVT2 フラグがクリアされます。

ビット 14 **TYPECEVT1CF**：Type C CC1 イベント･フラグ（TYPECEVT1）クリア
このビットをセットすると、UCPD_SR レジスタの TYPECEVT1 フラグがクリアされます。

ビット 13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **RXMSGENDCF**：Rx メッセージ受信フラグ（RXMSGEND）クリア
このビットをセットすると、UCPD_SR レジスタの RXMSGEND フラグがクリアされます。

ビット 11 **RXOVRCF**：Rx オーバーフローフラグ（RXOVR）クリア
このビットをセットすると、UCPD_SR レジスタの RXOVR フラグがクリアされます。

ビット 10 **RXHRSTDETCF**：Rx Hard Reset 検出フラグ（RXHRSTDET）クリア
このビットをセットすると、UCPD_SR レジスタの RXHRSTDET フラグがクリアされます。

ビット 9 **RXORDDETCF**：Rx オーダード・セット検出フラグ（RXORDDET）クリア
このビットをセットすると、UCPD_SR レジスタの RXORDDET フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TXUNDCF**：Tx アンダーフローフラグ（TXUND）クリア
このビットをセットすると、UCPD_SR レジスタの TXUND フラグがクリアされます。

ビット 5 **HRSTSENTCF**：Hard Reset 送信フラグ（HRSTSENT）クリア
このビットをセットすると、UCPD_SR レジスタの HRSTSENT フラグがクリアされます。

ビット 4 **HRSTDISCCF**：Hard Reset 破棄フラグ（HRSTDISC）クリア
このビットをセットすると、UCPD_SR レジスタの HRSTDISC フラグがクリアされます。

ビット 3 **TXMSGABTCF**：Tx メッセージアボートフラグ（TXMSGABT）クリア
このビットをセットすると、UCPD_SR レジスタの TXMSGABT フラグがクリアされます。

ビット 2 **TXMSGSENTCF**：Tx メッセージ送信フラグ（TXMSGSENT）クリア

このビットをセットすると、UCPD_SR レジスタの TXMSGSENT フラグがクリアされます。

ビット 1 **TXMSGDISCCF**：Tx メッセージ破棄フラグ（TXMSGDISC）クリア

このビットをセットすると、UCPD_SR レジスタの TXMSGDISC フラグがクリアされます。

ビット 0 予約済みであり、リセット値に保持する必要があります。

46.7.7 UCPD Tx オーダード・セット・タイプ･レジスタ（UCPD_TX_ORDSETR）

アドレス･オフセット：0x01C

リセット値：0x0000 0000

このレジスタへの書込みは、ペリフェラルが有効（UCPDEN = 1）で、パケット送信が進行中でない（TXSEND と TXHRST の両ビットがロー）場合にのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXORDSET[19:16]			
												rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXORDSET[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **TXORDSET[19:0]**：送信するオーダード・セット

このビットフィールドは､送信する 20 ビットの全シーケンスを決定します｡それぞれ 5 ビットの K コード 4 つで構成され、送信するパケットを定義します。ビット 0（K コード 1 のビット 0）を先頭、ビット 19（K コード 4 のビット 4）を末尾として順番に送信されます。

46.7.8 UCPD Tx ペイロード･サイズ･レジスタ（UCPD_TX_PAYSZR）

アドレス･オフセット：0x020

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが有効（UCPDEN = 1）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	TXPAYSZ[9:0]									
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **TXPAYSZ[9:0]**：これから送信するペイロードのサイズ

このビットフィールドは、ソフトウェアとハードウェアで変更されます。これから送信するペイロードのバイト数（ヘッダを含み、CRC を除いた値）が格納されます。データ･バイトが UCPD_TXDR レジスタに書き込まれるたびに、この値がデクリメントされ、TXIS ビットがセットされる処理が、この値がゼロなるまで続きます。以下に列挙する値は、送信開始前の標準的なペイロード･サイズです

0x2：2 バイト - プロトコル層からの制御メッセージのサイズ
0x6：6 バイト - プロトコル層からのデータ･メッセージの最小許容サイズ
0x1E：30 バイト - プロトコル層からのデータ･メッセージの最大許容サイズ
0x106：262 バイト - 拡張メッセージに可能な最大サイズ
0x3FF：1024 バイト - ペイロードに可能な最大サイズ（今後の拡張のために用意）

46.7.9 UCPD Tx データ･レジスタ（UCPD_TXDR）

アドレス･オフセット：0x024

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが有効（UCPDEN = 1）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **TXDATA[7:0]**：送信するデータ･バイト

46.7.10 UCPD Rx オーダード・セット・レジスタ（UCPD_RX_ORDSETR）

アドレス･オフセット：0x028

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPKINVALID[2:0]			RXSOP3OF4	RXORDSET[2:0]		
									r	r	r	r	r	r	r

ビット 31:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **RXSOPKINVALID[2:0]**：

このビットフィールドはデバッグ専用です。
0x0：破損した K コードはありません。
0x1：1 番目の K コードが破損しています。
0x2：2 番目の K コードが破損しています。
0x3：3 番目の K コードが破損しています。
0x4：4 番目の K コードが破損しています。
その他：無効

ビット 3 **RXSOP3OF4**：

このビットは、正常な K コードの数を示します。デバッグ専用です。
0：4 つの K コードのうち、4 つが正常です。
1：4 つの K コードのうち、3 つが正常です。

ビット 2:0 **RXORDSET[2:0]**：Rx オーダード・セット・コード検出

0x0：レシーバで SOP コードを検出しました。
0x1：レシーバで SOP' コードを検出しました。
0x2：レシーバで SOP" コードを検出しました。
0x3：レシーバで SOP'_Debug コードを検出しました。
0x4：レシーバで SOP"_Debug コードを検出しました。
0x5：レシーバで Cable Reset を検出しました。
0x6：レシーバで SOP 拡張 #1 を検出しました。
0x7：レシーバで SOP 拡張 #2 を検出しました。

46.7.11 UCPD Rx ペイロード･サイズ･レジスタ（UCPD_RX_PAYSZR）

アドレス･オフセット：0x02C

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	RXPAYSZ[9:0]									
						r	r	r	r	r	r	r	r	r	r

ビット 31:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:0 **RXPAYSZ[9:0]**：受信した Rx ペイロードのサイズ

このビットフィールドには、受信したペイロードのバイト数（ヘッダを含み、CRC を除いた値）が格納されます。新しいデータ･バイトを UCPD_RXDR レジスタに受信するたびに、この値がインクリメントされ、RXMSGEND フラグがセットされます（割込みが有効な場合は割込みを生成）。

0x2：2 バイト - プロトコル層からの制御メッセージのサイズ

0x6：6 バイト - プロトコル層からのデータ･メッセージの最小許容サイズ

0x1E：30 バイト - プロトコル層からのデータ･メッセージの最大許容サイズ

0x106：262 バイト - 拡張メッセージに可能な最大サイズ

0x3FF：1024 バイト - ペイロードに可能な最大サイズ（今後の拡張のために用意）

バイト受信が進行中（RXMSGEND フラグがロー）の場合は、このビットフィールドに誤った値が返されることがあります。

46.7.12 UCPD 受信データ･レジスタ（UCPD_RXDR）

アドレス･オフセット：0x030

リセット値：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
								r	r	r	r	r	r	r	r

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **RXDATA[7:0]**：受信されたデータ･バイト

46.7.13 UCPD Rx オーダード・セット拡張レジスタ 1（UCPD_RX_ORDEXTR1）

アドレス･オフセット：0x034

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが無効（UCPDEN = 0）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX1[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXSOPX1 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **RXSOPX1[19:0]**：受信されたオーダード・セット 1

このビットフィールドには、受信された 20 ビットの全シーケンスが格納されます。それぞれ 5 ビットの K コード 4 つで構成されます。ビット 0（K コード 1 のビット 0）を先頭、ビット 19（K コード 4 のビット 4）を末尾として順番に受信します。

46.7.14 UCPD Rx オーダード・セット拡張レジスタ 2（UCPD_RX_ORDEXTR2）

アドレス･オフセット：0x038

リセット値：0x0000 0000

このレジスタへの書込みはペリフェラルが無効（UCPDEN = 0）であるときにのみ有効です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX2[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXSOPX2 [15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:20 予約済みであり、リセット値に保持する必要があります。

ビット 19:0 **RXSOPX2[19:0]**：受信されたオーダード・セット2

このビットフィールドには、受信された 20 ビットの全シーケンスが格納されます。それぞれ 5 ビットの K コード 4 つで構成されます。ビット 0（K コード 1 のビット 0）を先頭、ビット 19（K コード 4 のビット 4）を末尾として順番に受信します。

46.7.15 UCPD レジスタ･マップ

表 433. UCPD レジスタのマップとリセット値

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	UCPD_CFGR1	UCPDEN	RXDMAEN	TXDMAEN	RXORDSETEN[8:0]									PSC_USBPDCLK[2:0]			Res.	TRANSWIN[4:0]					IFRGAP[4:0]					HBITCLKDIV[5:0]						
	リセット値	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x004	UCPD_CFGR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	WUPEN	FORCECLK	RXFILT2N3	RXFILTDIS
	リセット値																													0	0	0	0
0x008	予約済みです。	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
0x00C	UCPD_CR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	CC2TCDIS	CC1TCDIS	Res.	RDCH	FRSTX	FRSRXEN	Res.	Res.	Res.	Res.	CCENABLE[1:0]		ANAMODE	ANASUBMODE[1:0]		PHYCCSEL	PHYRXEN	RXMODE	TXHRST	TXSEND	TXMODE[1:0]	
	リセット値											0	0		0	0	0					0	0	0	0	0	0	0	0	0	0	0	0
0x010	UCPD_IMR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTIE	Res.	Res.	Res.	Res.	TYPECEVT2IE	TYPECEVT1IE	Res.	RXMSGENDIE	RXOVRIE	RXHRSTDETIE	RXORDDETIE	RXNEIE	Res.	TXUNDIE	HRSTSENTIE	HRSTDISCIE	TXMSGABTIE	TXMSGSENTIE	TXMSGDISCIE	TXISIE
	リセット値												0					0	0		0	0	0	0	0		0	0	0	0	0	0	0
0x014	UCPD_SR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVT	TYPEC_VSTATE_CC2[1:0]		TYPEC_VSTATE_CC1 [1:0]		TYPECEVT2	TYPECEVT1	RXERR	RXMSGEND	RXOVR	RXHRSTDET	RXORDDET	RXNE	Res.	TXUND	HRSTSENT	HRSTDISC	TXMSGABT	TXMSGSENT	TXMSGDISC	TXIS
	リセット値												0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	0	0	0	0
0x018	UCPD_ICR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	FRSEVTCF	Res.	Res.	Res.	Res.	TYPECEVT2CF	TYPECEVT1CF	Res.	RXMSGENDCF	RXOVRCF	RXHRSTDETCF	RXORDDETCF	Res.	Res.	TXUNDCF	HRSTSENTCF	HRSTDISCCF	TXMSGABTCF	TXMSGSENTCF	TXMSGDISCCF	Res.
	リセット値												0					0	0		0	0	0	0			0	0	0	0	0	0	
0x01C	UCPD_TX_ORDSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXORDSET[19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x020	UCPD_TX_PAYSZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXPAYSZ[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x024	UCPD_TXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0

表 433. UCPD レジスタのマップとリセット値 （続き）

オフセット	レジスタ名	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x028	UCPD_RX_ORDSETR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPKINVALID[2:0]			RXSOP3OF4	RXORDSET[2:0]		
	リセット値																										0	0	0	0	0	0	0
0x02C	UCPD_RX_PAYSZR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXPAYSZ[9:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x030	UCPD_RXDR	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXDATA[7:0]							
	リセット値																									0	0	0	0	0	0	0	0
0x034	UCPD_RX_ORDEXTR1	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX1 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x038	UCPD_RX_ORDEXTR2	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	RXSOPX2 [19:0]																			
	リセット値													0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

レジスタ境界アドレスについては、80ページのセクション 2.2 を参照してください。

47 デバッグ･サポート（DBG）

47.1 概要

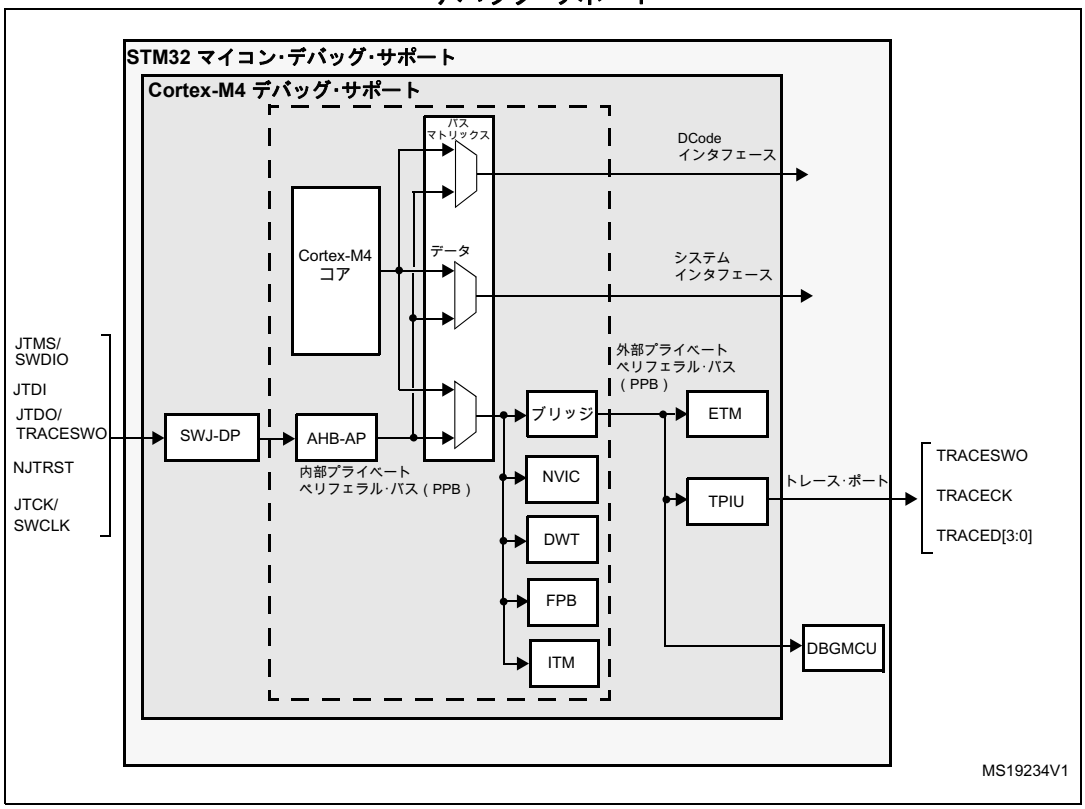
STM32G4 シリーズ は、FPU 搭載 Cortex®-M4 コアを中心にして構築されており、高度なデバッグ機能のためのハードウェア拡張機能を含んでいます。デバッグ拡張機能によって、コアは、特定の命令フェッチ時（ブレークポイント）またはデータアクセス時（ウォッチポイント）に停止することができます。停止したとき、コアの内部状態とシステムの外部状態を調査することができます。調査が終わったら、コアとシステムを復元して、プログラム実行を再開することができます。

デバッグ機能は、STM32G4 シリーズ MCU への接続時とデバッグ時にデバッガによって使用されます。

デバッグ用に、次の 2 つのインタフェースを使用できます。

- シリアルワイヤ
- JTAG デバッグポート

図 685. ブロック図 - STM32 マイクロコントローラおよび FPU 搭載 Cortex®-M4 レベルのデバッグ･サポート



注： **FPU 搭載 Cortex®-M4 コアに内蔵されているデバッグ機能は、Arm® CoreSight Design Kit のサブセットです。**

Arm®FPU 搭載 Cortex®-M4 コアは、統合的なオンチップデバッグサポート機能を提供します。この機能は以下の要素で構成されます。

- SWJ-DP：シリアルワイヤ／JTAG デバッグポート
- AHP-AP：AHB アクセスポート
- ITM：計測トレース･マクロセル（Instrumentation Trace Macrocell）
- FPB：Flash パッチ･ブレークポイント（Flash Patch Breakpoint）
- DWT：データ･ウォッチポイント･トリガ（Data Watchpoint Trigger）
- TPIU：トレース･ポート･インタフェース･ユニット（Trace Port Interface Unit：対応するピンがマッピングされる大型パッケージに搭載）
- ETM：エンベデッド･トレース･マクロセル（Embedded Trace Macrocell：対応するピンがマッピングされる、STM32G4 シリーズ デバイスの大型パッケージのみに搭載）

また、STM32G4 シリーズ専用の以下のデバッグ機能も内蔵されています。

- 柔軟性の高いデバッグピンの割り当て
- MCU デバッグボックス（低電力モードのサポート、ペリフェラルクロックの制御など）

注： **Arm® FPU 搭載 Cortex®-M4 コアでサポートされているデバッグ機能の詳細は、FPU 搭載 Cortex®-M4-r0p1 テクニカルリファレンスマニュアルおよび CoreSight Design Kit-r0p1 TRM を参照してください（セクション 47.2：Arm® リファレンス資料を参照）。**

47.2 Arm® リファレンス資料

- FPU 搭載 Cortex®-M4 r0p1 テクニカルリファレンスマニュアル（TRM）

  http://infocenter.arm.com で「FPU 搭載 Cortex®-M4 テクニカルリファレンスマニュアル」を検索してください。
- Arm® Debug Interface V5
- Arm® CoreSight Design Kit revision r0p1 Technical Reference Manual

47.3 SWJ デバッグポート（シリアルワイヤと JTAG）

STM32G4 シリーズ のコアには、シリアルワイヤ／JTAG デバッグポート（SWJ-DP）が組み込まれています。これは、JTAG-DP（5 ピン）インタフェースと SW-DP（2 ピン）インタフェースを組み合わせた Arm® の標準の CoreSight デバッグ･ポートです。

- JTAG デバッグ･ポート（JTAG-DP）は、AHP-AP ポートに 5 ピンの標準 JTAG インタフェースを提供します。
- シリアルワイヤデバッグポート（SW-DP）は、AHP-AP ポートに 2 ピン（クロック + データ）のインタフェースを提供します。

SWJ-DP では、SW-DP の 2 個の JTAG ピンは、JTAG-DP の 5 個の JTAG ピンの一部と多重化されています。

図 686. SWJ デバッグポート

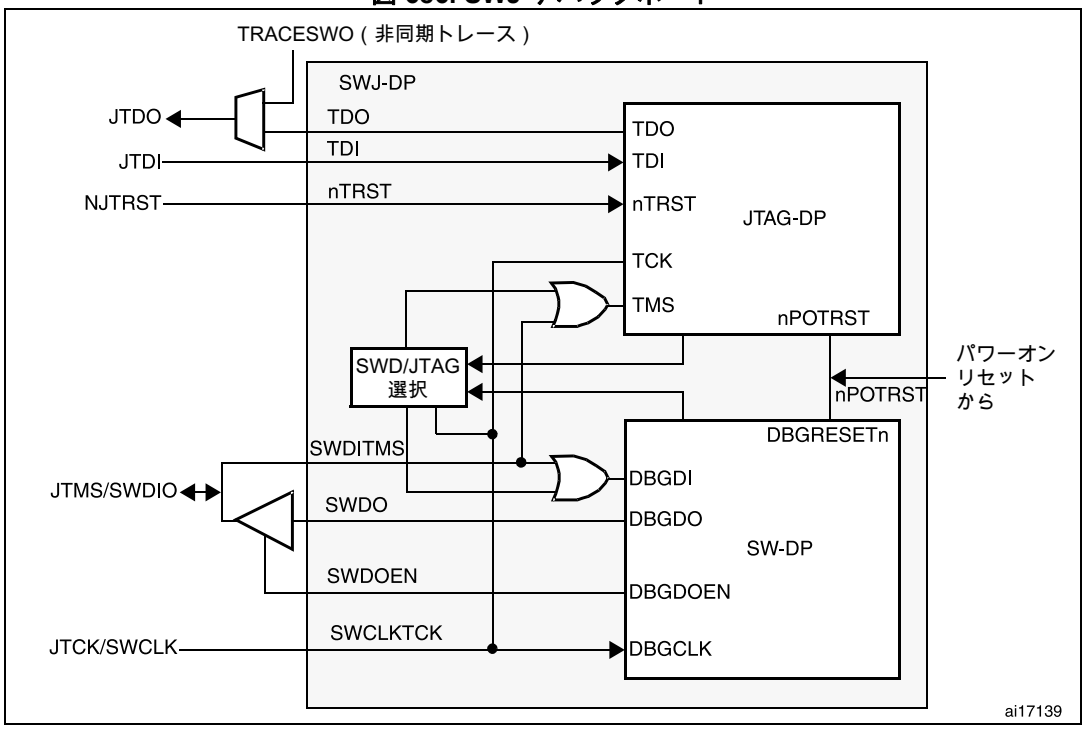


図 686 は、非同期 TRACE 出力（TRACESWO）が TDO と多重化されていることを示します。つまり非同期トレースは、JTAG-DP ではなく、SW-DP にのみ使用できます。

47.3.1 JTAG-DP または SW-DP の選択メカニズム

デフォルトでは、JTAG デバッグポートが選択されています。

SW-DP に切り替えるには、デバッガホストは、TMS／TCK（それぞれ SWDIO／SWCLK に配置）に専用の JTAG シーケンスを提供して、JTAG-DP を無効にし SW-DP を有効にする必要があります。このように、SWCLK ピン と SWDIO ピンのみを使用して、SWDP を有効にすることができます。

このシーケンスは、

1. TMS（SWDIO）=1 で 50 を超える TCK サイクルを送信します。
2. TMS（SWDIO）に 16 ビットシーケンス（0111100111100111）を MSB ファーストで送信します。
3. TMS（SWDIO）=1 で 50 を超える TCK サイクルを送信します。

47.4 ピン名とデバッグポートピン

STM32G4 シリーズ MCUは、使用できるピン数の異なるさまざまなパッケージに組み込まれています。このため、ピンを使用する一部の機能（ETM）は、パッケージによって異なることがあります。

47.4.1 SWJ デバッグポートピン

5 個のピンが SWJ-DP 用の STM32G4 シリーズ からの出力として使用されます（ 汎用入出力のオルタネート機能 ）。これらのピンはすべてのパッケージで使用できます。

表 434. SWJ デバッグポートピン

SWJ-DP ピン名	JTAG デバッグポート		SW デバッグポート		ピン割り当て
	タイプ	説明	タイプ	デバッグ割り当て	
JTMS／SWDIO	I	JTAG テストモード選択	IO	シリアル･ワイヤ･データ入出力	PA13
JTCK／SWCLK	I	JTAG テストクロック	I	シリアルワイヤクロック	PA14
JTDI	I	JTAG テストデータ入力	-	-	PA15
JTDO/TRACESWO	O	JTAG テストデータ出力	-	非同期トレースが有効な場合は TRACESWO	PB3
NJTRST	I	JTAG テスト nReset	-	-	PB4

47.4.2 柔軟性の高い SWJ-DP ピンの割り当て

RESET（SYSRESETn または PORESETn）後、SWJ-DP 用に使用される 5 個のピンはすべて、デバッガホストによってすぐに使用可能な専用ピンとして割り当てられます（なお、デバッガホストによって明示的にプログラミングされた場合を除いて、トレース出力は割り当てられません）。

ただし、STM32G4 シリーズ MCU は SWJ-DP ポートの一部またはすべてを無効にすることができます、従って関連するピンは NJTRST を除いて汎用 IO（GPIO）使用が可能です（下記表にグレー表示）。NJTRST は未接続にすることができますが、デバッガのコネクションを失わずに汎用 GPIO として使うことはできません。SWJ-DP ポートピンを無効にする方法の詳細については、セクション 9.3.2：I/O ピンオルタネート機能マルチプレクサと配置を参照してください。

表 435. 柔軟性の高い SWJ-DP ピンの割り当て

使用可能なデバッグポート	SWJ IO ピンの割当て				
	PA13 / JTMS/ SWDIO	PA14 / JTCK/ SWCLK	PA15 / JTDI	PB3 / JTDO	PB4/ NJTRST
全 SWJ（JTAG-DP + SW-DP）、リセット状態	X	X	X	X	X
全 SWJ（JTAG-DP + SW-DP）、NJTRST なし	X	X	X	X	
JTAG-DP 無効、SW-DP 有効	X	X			
JTAG-DP 無効、SW-DP 無効	解放				

47.4.3 JTAG ピンの内部プルアップ／プルダウン

JTAG 入力ピンは、デバッグ･モード機能を制御するためにフリップフロップに直結されます。したがって、JTAG 入力ピンをフロート状態にしないことが必要です。これらのフリップフロップの一部のクロックに直結される SWCLK／TCK ピンについては、特に注意が必要です。

入出力レベルを正しく制御するため、デバイスには内部プルアップ／プルダウンが JTAG 入力ピンに内蔵されています。

- NJTRST：内部プルアップ
- JTDI：内部プルアップ
- JTMS/SWDIO：内部プルアップ
- TCK/SWCLK：内部プルダウン

JTAG 入出力がユーザソフトウェアによって解放されると、GPIO コントローラが再び制御権を獲得します。リセット状態では、GPIO 制御レジスタは入出力をこれと同等の状態に設定します。

- NJTRST：入力プルアップ
- JTDI：入力プルアップ
- JTMS/SWDIO：入力プルアップ
- JTCK/SWCLK：入力プルダウン
- JTDO：入力フローティング

ソフトウェアはこれらの入出力を標準の GPIO 信号として使用することができます。

注： **JTAG IEEE 規格では、TDI、TMS、および nTRST にプルアップを追加することを推奨していますが、TCK に関しては特別な推奨はありません。ただし、JTCK に関しては、デバイスは内蔵プルダウンを必要とします。**

プルアップとプルダウンを内蔵しているため、外部抵抗を追加する必要がありません。

NJTRST（PB4）ピンには、UCPD_DBCC2 ピン（PA10）の電圧によって制御される内部 UCPD プルダウン抵抗（5.1 KΩ）を実装する UCPD_CC2 機能もあります。JTAG を使用するには、CC ラインに対するプルダウンの効果を除去する必要があります。それには、PWR_CR3 レジスタの UCPD1_DBDIS ビット（USB Type-C と Power Delivery デッド･バッテリの無効化）を使用します（セクション 46.4.6：UCPD Type-C のプルアップ抵抗（Rp）およびプルダウン抵抗（Rd）参照）。

47.4.4 シリアル･ワイヤの使用と、未使用のデバッグ･ピンを GPIO として解放する方法

シリアルワイヤ DP を使って一部の GPIO を解放するには、ユーザソフトウェアが GPIO_MODER レジスタで GPIO（PA15、PB3、および PB4）コンフィギュレーションモードを変更する必要があります。これによって PA15、PB3、および PB4 が解放され、GPIO として使用できるようになります。

デバッグ時には、ホストは次の動作を行います。

- システムリセット中に、すべての SWJ ピンが割り当てられます（JTAG-DP + SW-DP）。
- システムリセット中に、デバッガホストは、JTAG-DP から SW-DP に切り替える JTAG シーケンスを送信します。
- システムリセット中に、さらに、デバッガはベクタリセットにブレークポイントを設定します。
- システムリセットは解除され、コアは停止します。
- これ以降のすべてのデバッグ通信は、SW-DP を使用して行われます。残りの JTAG ピンは、ユーザソフトウェアによって GPIO として再割当てできます。

注： **ユーザソフトウェアの設計に関しては、次の点に注意してください。**

デバッグ･ピンを解放するには、リセット後にユーザ･ソフトウェアがピンを解放するまでの一定の期間、デバッグ･ピンが、まず入力プルアップ（nTRST、TMS、TDI）、入力プルダウン（TCK）、または出力トライステート（TDO）に設定されることに注意してください。

デバッグ･ピン（JTAG、SW、または TRACE）をマッピングすると、IOPORT コントローラでそのピンに対応する入出力ピンの設定を変更してもその効果はありません。

47.5 STM32G4 シリーズJTAG TAP 接続

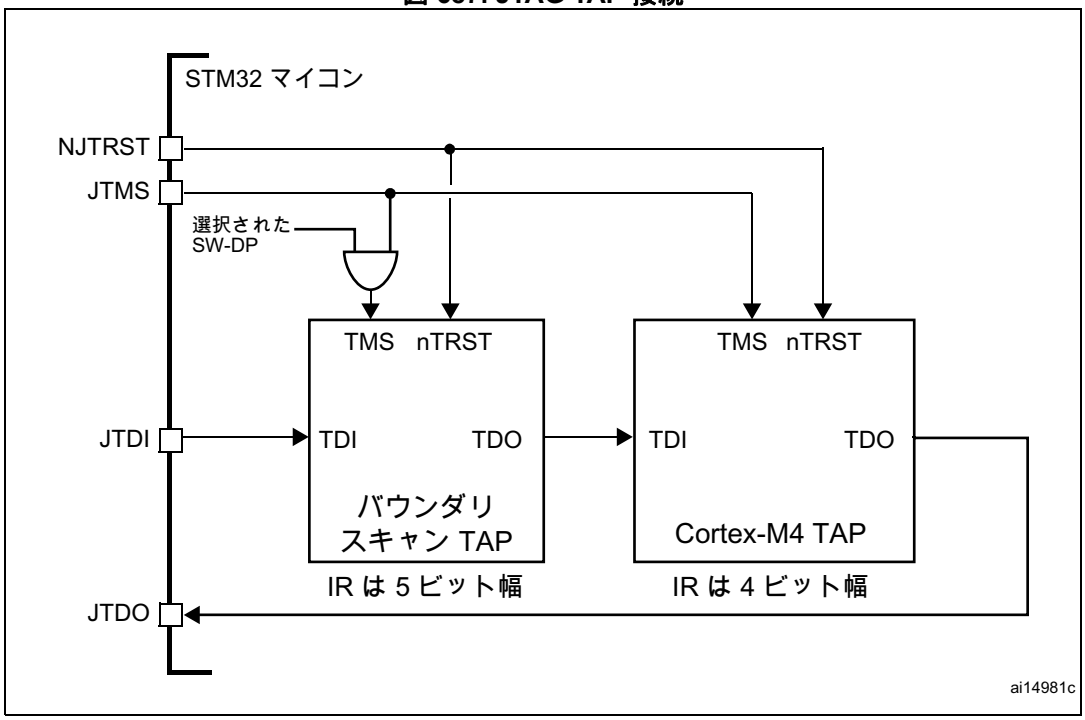
STM32G4 シリーズ MCU は、バウンダリスキャン TAP（IR は 5 ビット幅）と FPU 搭載 Cortex®-M4 TAP （IR は 4 ビット幅）という、シリアル接続された 2 個の JTAG TAP を内蔵しています。

デバッグ目的で FPU 搭載 Cortex®-M4 の TAP にアクセスするには、

1. まず、バウンダリスキャン TAP の BYPASS 命令をシフトする必要があります。
2. 次に、IR シフトごとに、スキャンチェインには 9 (=5+4) ビットが格納され、未使用の TAP 命令を BYPASS 命令によってシフトインする必要があります。
3. データシフトごとに、BYPASS モードにある未使用の TAP は、データスキャンチェインに 1 個のデータビットを追加します。

注： **重要：専用の Arm® JTAG シーケンスを使用してシリアルワイヤが選択されると、バウンダリスキャン TAP は自動的に無効になります（JTMS は強制的にハイレベルになります）。**

図 687. JTAG TAP 接続



47.6 ID コードとロック機構

STM32G4 シリーズ MCU には、内部にいくつかの ID コードがあります。ツール設計者は、外部 PPB メモリマップのアドレス 0xE0042000 に配置されている MCU デバイス ID コードを使用して、デバッガをロックすることを強く推奨します。

47.6.1 MCU デバイス ID コード

STM32G4 シリーズ MCU には MCU ID コードが内蔵されています。この ID は、ST 社製 MCU の部品番号とダイのリビジョンを識別します。これは DBG_MCU 部品の一部であり、外部 PPB バスに配置されます（2063ページのセクション 47.16 を参照）。このコードにアクセスするには、JTAG デバッグポート（4～5 本のピン）、SW デバッグポート（2 本のピン）、またはユーザソフトウェアを使用します。アクセスは MCU がシステムリセット中でも可能です。

デバッガ／プログラマツールでは、DEV_ID(11:0) のみを識別に使用してください。

DBGMCU_IDCODE

アドレス：0xE004 2000

32 ビット･アクセスのみサポートされます。読出し専用。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
REV_ID[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	DEV_ID[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:16 **REV_ID[15:0]** リビジョン識別子

このフィールドは、デバイスのリビジョンを示します。

0x1000：リビジョン A

0x1001：リビジョン Z（カテゴリ 4 デバイスのみ）

0x2000：リビジョン B

0x2001：リビジョン Z

0x2002：リビジョン Y

0x2003：リビジョン X

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DEV_ID[11:0]**：デバイス識別子

デバイス ID は以下のとおりです。

– 0x468：カテゴリ 2 デバイス（表 1：STM32G4 シリーズメモリ容量を参照）

– 0x469：カテゴリ 3 デバイス（表 1：STM32G4 シリーズメモリ容量を参照）

– 0x479：カテゴリ 4 デバイス（表 1：STM32G4 シリーズメモリ容量を参照）

47.6.2 バウンダリ･スキャン TAP

JTAG ID コード

STM32G4 シリーズ BSC （バウンダリ･スキャン）の TAP は、16469041（カテゴリ 3 デバイス）、16468041 （カテゴリ 2 デバイス）、16479041（カテゴリ 4 デバイス）に等しい JTAG ID コードを内蔵しています。

47.6.3 FPU 搭載 Cortex®-M4 TAP

Arm® FPU 搭載 Cortex®-M4 の TAP は JTAG ID コードを内蔵しています。この ID コードは Arm® のデフォルトであり、変更されていません。このコードは、JTAG デバッグポートからのみアクセスできます。
このコードは **0x4BA00477** です（FPU 搭載 Cortex®-M4 r0p1 に対応、セクション 47.2 : Arm® リファレンス資料を参照）。

47.6.4 FPU 搭載 Cortex®-M4 JEDEC-106 ID コード

Arm® FPU 搭載 Cortex®-M4 は JEDEC-106 ID コードを内蔵しています。これは内部 PPB バスのアドレス 0xE00FF000_0xE00FFFFF に配置された 4KB ROM テーブルに置かれています。

このコードは、JTAG デバッグポート（4～5 本のピン）、SW デバッグポート（2 本の ピン）、またはユーザソフトウェアによってアクセスできます。

47.7 JTAG デバッグポート

標準的な JTAG ステートマシンは、4 ビット命令レジスタ（IR）と 5 個のデータレジスタを搭載しています（全詳細については、FPU 搭載 FPU 搭載 Cortex®-M4 r0p1 テクニカルリファレンスマニュアル（TRM）を参照し、参考として セクション 47.2 : Arm® リファレンス資料を参照してください）。

表 436. JTAG デバッグ･ポートのデータ･レジスタ

IR(3:0)	データ･レジスタ	詳細
1111	BYPASS [1 ビット]	-
1110	IDCODE [32 ビット]	ID コード 0x4BA00477（Arm® FPU 搭載 Cortex®-M4 r0p1-01rel0 の ID コード）
1010	DPACC [35 ビット]	デバッグポートアクセスレジスタ デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 – 入力データ転送時 : ビット 34:3 = DATA[31:0] = 書込みリクエスト用に転送する 32 ビットデータ ビット 2:1 = A[3:2] = デバッグポートレジスタの 2 ビットアドレス ビット 0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時 : ビット 34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答 : 010 = OK/FAULT 001 = WAIT その他 = 予約済み A[3:2] ビットの説明については、表 437 を参照してください。

表 436. JTAG デバッグ･ポートのデータ･レジスタ （続き）

IR(3:0)	データ･レジスタ	詳細
1011	APACC [35 ビット]	アクセスポートアクセスレジスタ アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 – 入力データ転送時： ビット 34:3 = DATA[31:0] = 書込みリクエスト用にシフトインする 32 ビットデータ ビット 2:1 = A[3:2] = 2 ビットアドレス（サブアドレス AP レジスタ） ビット 0 = RnW = 読出しリクエスト（1）または書込みリクエスト（0） – 出力データ転送時： ビット 34:3 = DATA[31:0] = 読出しリクエストに続いて読み出される 32 ビットデータ ビット 2:0 = ACK[2:0] = 3 ビット確認応答： 010 = OK/FAULT 001 = WAIT その他 = 予約済み 次の項目の組み合わせとして、多くの AP レジスタ（AHB-AP を参照）をアドレス指定します。 – シフトされた値 A[3:2] – DP SELECT レジスタの現在値
1000	ABORT [35 ビット]	アボートレジスタ – ビット 31:1 = 予約済み – ビット 0 = DAPABORT：DAP アボートを生成するには 1 を書き込みます

表 437. シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ

アドレス	A[3:2] の値	説明
0x0	00	予約済みであり、リセット値に保持する必要があります。
0x4	01	DP CTRL/STAT レジスタ：次の目的で使用されます。 – システム／デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – プッシュ比較／プッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読出し
0x8	10	DP SELECT レジスタ。現在のアクセスポートとアクティブな 4 ワードレジスタウィンドウの選択に使用されます。 – ビット 31:24：（APSEL）：現在の AP を選択します。 – ビット 23:8：予約済み – ビット 7:4：（APBANKSEL）：現在の AP でアクティブな 4 ワードレジスタウィンドウを選択します。 – ビット 3:0：予約済み
0xC	11	DP RDBUFF レジスタ：一連の動作のあとで（新しい JTAG-DP 動作をリクエストせずに）、デバッガが最終結果を得られるようにします。

47.8 SW デバッグポート

47.8.1 SW プロトコルの概要

この同期式シリアルプロトコルでは、次の 2 個のピンを使用します。

- SWCLK：ホストからターゲットへのクロック
- SWDIO：双方向

このプロトコルでは、2 バンクのレジスタ（DPACC レジスタと APACC レジスタ）の読出し／書込みが可能です。

ビットは、ワイヤ上を LSB ファーストで転送されます。

SWDIO を双方向管理するには、ボード上でラインがプルアップされている必要があります（Arm® の推奨値は 100 kΩ）。

プロトコルで SWDIO の方向が変化するたびに、ラインがホストからもターゲットからも駆動されないターンアラウンド時間が挿入されます。このターンアラウンド時間のデフォルトは 1 ビット時間ですが、SWCLK 周波数の設定によって調整できます。

47.8.2 SW プロトコルシーケンス

各シーケンスは 3 つのフェーズで構成されます。

1. ホストによって送信されるパケットリクエスト（8 ビット）
2. ターゲットによって送信される確認応答（3 ビット）
3. ホストまたはターゲットによって送信されるデータ転送フェーズ（33 ビット）

表 438. パケットリクエスト（8 ビット）

ビット	名前	説明
0	スタート	"1"である必要があります。
1	APnDP	0：DP アクセス 1：AP アクセス
2	RnW	0：書込みリクエスト 1：読出しリクエスト
4:3	A[3:2]	DP または AP レジスタのアドレスフィールド（表 437を参照）
5	Parity	先行するビットの 1 ビットパリティ
6	STOP	0
7	Park	ホストによって駆動されません。プルアップの効果で、ターゲットには「1」と読み出される必要があります。

DPACC レジスタと APACC レジスタの詳細については、FPU 搭載 Cortex®-M4 r0p1 *TRM* を参照してください。

パケットリクエストの後には、必ずホストもターゲットもラインを駆動しないターンアラウンド時間（デフォルトでは 1 ビット）が続きます。

表 439. ACK 応答（3 ビット）

ビット	名前	説明
0..2	ACK	001：フォールト 010：WAIT 100：OK

読出しトランザクションの場合や、受信した ACK 応答が WAIT または FAULT の場合にのみ、ACK 応答の後にターンアラウンド時間が必要になります。

表 440. データ転送（33 ビット）

ビット	名前	説明
0..31	WDATA または RDATA	書込み／読出しデータ
32	Parity	32 データビットの 1 ビットパリティ

読出しトランザクションの場合にのみ、データ転送の後にターンアラウンド時間が続く必要があります。

47.8.3 SW-DP ステートマシン（リセット、アイドル状態、ID コード）

SW-DP のステート･マシンには、SW-DP を識別する内部 ID コードがあります。これは JEP-106 規格に準じています。この ID コードは、Arm® のデフォルトコードであり、**0x2BA01477**（FPU 搭載 Cortex®-M4 r0p1 に対応）がセットされています。

注： **SW-DP ステートマシンは、ターゲットがこの ID コードを読み出すまで非アクティブであることに注意してください。**

- パワーオンリセット後、または DP が JTAG から SWD に切り替えられた後、またはラインが 50 サイクルを超えてハイレベルにあった後では、SW-DP ステートマシンは RESET 状態になります。
- リセット状態のあと、ラインが 2 サイクル以上の間ローレベルであれば、SW-DP ステートマシンは アイドル状態になります。
- リセット状態のあとは、まずアイドル状態に入り、次に DP-SW ID CODE レジスタの読出しアクセスを行う必要があります。そうしないと、ターゲットは、他のトランザクションに対して ACK 応答の FAULT を発行します。

SW-DP ステート･マシンの詳細については、FPU 搭載 Cortex®-M4 r0p1 TRM および CoreSight Design Kit r0p1 TRM を参照してください。

47.8.4 DP と AP の読出し／書込みアクセス

- DP への読出しアクセスはポストされません。つまり、ターゲットは、ACK 応答が”OK”の場合はただちに応答し、ACK 応答が”WAIT”の場合は遅れて応答します。
- AP への読出しアクセスはポストされます。つまり、アクセスの結果は次の転送時に返されます。次のアクセスが AP アクセスでない場合、結果を得るには DP-RDBUFF レジスタを読み出す必要があります。
  AP 読出しアクセスが成功したかどうかを判断するため、DP-CTRL/STAT レジスタの READOK フラグは、AP 読出しアクセスまたは RDBUFF 読出しリクエストのたびに更新されます。
- SW-DP は、DP と AP の両方の書込みに使用できる書込みバッファを実装しているため、たとえ他のトランザクションが未処理であっても、書込み動作を受け付けることができます。書込みバッファがフルのとき、ターゲットの ACK 応答は”WAIT”です。例外として、IDCODE 読出し、

CTRL/STAT 読出し、または ABORT 書込みは、書込みバッファがフルであっても受け付けられます。

- 非同期クロックドメイン SWCLK と HCLK によって、書込みを内部的に有効にするには、書込みトランザクション後（パリティビット後）に SWCLK の 2 サイクルが余分に必要となります。これらのサイクルは、ラインをローレベルに駆動している間（アイドル状態）に適用してください。これは、パワーアップリクエストのために CTRL/STAT の書込みを行う際に特に重要です。パワーアップを必要とする次のトランザクションがただちに発生すると､そのトランザクションは失敗します。

47.8.5 SW-DP レジスタ

これらのレジスタへのアクセスは、APnDP = 0 のときに開始されます。

表 441. SW-DP レジスタ

A[3:2]	読出し／書込み	SELECT レジスタの CTRLSEL ビット	レジスタ	注
00	読出し	-	IDCODE	製造者コードは ST のコード **0x2BA01477** ではありません（SW-DP を指定）。
00	書込み	-	ABORT	-
01	読出し／書込み	0	DP CTRL/STAT	用途は以下のとおりです。 – システム／デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – プッシュ比較／プッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読出し
01	読出し／書込み	1	WIRE CONTROL	物理的なシリアルポートプロトコルの設定（ターンアラウンド時間など）を行います。
10	読出し	-	READ RESEND	元の AP 転送を反復しなくても、破壊されたデバッガ転送からの読出しデータの復旧を可能にします。
10	書込み	-	SELECT	現在のアクセス･ポートとアクティブな 4 ワード･レジスタ･ウィンドウを選択します。
11	読出し／書込み	-	読出しバッファ	AP アクセスはポストされるため、この読出しバッファは効果的です（AP 読出しリクエストの結果は、次の AP トランザクションで取得できる）。 この読出しバッファは、新しいトランザクションを開始することなく、前回の読出しの結果として AP から出力されるデータをキャプチャします。

47.8.6 SW-AP レジスタ

これらのレジスタへのアクセスは、APnDP = 1 のときに開始されます。

次の項目の組み合わせとして、多くの AP レジスタ（AHB-AP を参照）をアドレス指定します。

- シフトされた値 A[3:2]
- DP SELECT レジスタの現在値

47.9 AHB-AP（AHB アクセスポート）- JTAG-DP と SW-DP の両方に有効

機能：

- システムアクセスはプロセッサステータスから独立しています。
- SW-DP または JTAG-DP が AHB-AP にアクセスします。
- AHB-AP はバスマトリックスの AHB マスタです。したがって、AHB-AP は ICode バスを除くすべてのデータバス（Dcode バス、システムバス、内部／外部の PPB バス）にアクセスできます。
- ビットバンドトランザクションがサポートされます。
- AHB-AP トランザクションは FPB を迂回します。

32 ビット AHP-AP レジスタは 6 ビット幅（最大 64 ワードまたは 256 バイト）であり、以下の構成となっています。

c) ビット [7:4] = DP_SELECT レジスタのビット [7:4] APBANKSEL

d) ビット [3:2] = SW-DP 用の 35 ビットパケットリクエストの 2 ビットアドレス A(3:2)

FPU 搭載 Cortex®-M4 の AHB-AP は、9 個の 32 ビットレジスタを内蔵しています。

表 442. FPU 搭載 Cortex®-M4AHB-AP レジスタ

アドレスオフセット	レジスタ名	注
0x00	AHB-AP 制御およびステータス･ワード	AHB インタフェースを通じて転送を設定および制御します（サイズ、hprot、現在の転送のステータス、アドレスインクリメントタイプ）。
0x04	AHB-AP 転送アドレス	-
0x0C	AHB-AP データ読出し／書込み	-
0x10	AHB-AP バンク･データ 0	転送アドレスレジスタを書き換えずに、4 個のアラインドデータワードを直接配置します。
0x14	AHB-AP バンク･データ 1	
0x18	AHB-AP バンク･データ 2	
0x1C	AHB-AP バンク･データ 3	
0xF8	AHB-AP デバッグ ROM アドレス	デバッグインタフェースのベースアドレス
0xFC	AHB-AP ID レジスタ	-

詳細については、FPU 搭載 Cortex®-M4 r0p1 TRM を参照してください。

47.10 コアデバッグ

コアデバッグはコアデバッグレジスタを通じてアクセスされます。これらのレジスタへのデバッグアクセスには、Advanced High-performance Bus （AHB-AP）ポートを使用します。プロセッサは、内部のプライベートペリフェラルバス （PPB）を介してこれらのレジスタに直接アクセスできます。

コアデバッグは 4 個のレジスタから構成されています。

表 443. コアデバッグレジスタ

レジスタ	説明
DHCSR	32 ビットのデバッグ停止制御／ステータス･レジスタ： レジスタプロセッサの状態についてのステータス情報を提供し、コアデバッグを有効にし、プロセッサの停止とステップ実行を行います。
DCRSR	17 ビットのデバッグ･コア･レジスタ･セレクタ･レジスタ： データの転送先または転送元となるプロセッサレジスタを選択します。
DCRDR	32 ビットのデバッグ･コア･レジスタ･データ･レジスタ： DCRSR（セレクタ）レジスタによって選択されたプロセッサとの間でレジスタの読出しおよび書込みに使用するデータを保持します。
DEMCR	32 ビットのデバッグ例外／モニタ制御レジスタ： ベクタキャッチとデバッグモニタの制御を行います。このレジスタには、TRACE を使用できるようにする ***TRCENA*** というビットがあります。

注： **重要：これらのレジスタは、システムリセットによってはリセットされません。パワーオンリセットによってのみリセットされます。**

詳細については、FPU 搭載 Cortex®-M4 r0p1 TRM を参照してください。

リセット時に停止させるには、以下の手順が必要です。

- デバッグ例外／モニタ制御レジスタのビット 0（VC_CORRESET）を有効にします。
- デバッグ停止制御／ステータスレジスタのビット 0（C_DEBUGEN）を有効にします。

47.11 システムリセット中のデバッガホスト接続機能

STM32G4 シリーズ MCU のリセットシステムは、次のリセットソースから構成されます。

- POR（パワーオンリセット）：パワーアップのたびに RESET をアサートします。
- 内部ウォッチドッグリセット
- ソフトウェア･リセット
- 外部リセット

FPU 搭載 Cortex®-M4 では、デバッグ部のリセット（一般に PORRESETn）とそれ以外のリセット（SYSRESETn）が区別されます。

こうすることで、リセットベクタをフェッチするときにコアを停止させるようコアデバッグレジスタをプログラミングすることで、デバッガはシステムリセット中に接続が可能になります。これによってホストはシステム･リセットを解除でき、コアはどの命令も実行していない状態でただちに停止します。さらに、システム･リセット中にデバッグ機能をプログラミングできます。

注： **システムリセット中にデバッガホストを接続する（リセットベクタにブレークポイントを設定する）ことを強く推奨します。**

47.12 FPB（Flashパッチブレークポイント）

FPB ユニットは、

- ハードウェアブレークポイントを実装します。
- コード空間からシステム空間に、コードとデータをパッチします。この機能によって、コードメモリ空間にあるソフトウェアバグの訂正が可能になることがあります。

ソフトウェアパッチとハードウェアブレークポイントは同時に使用できません。

FPB は以下から構成されます。

- コード空間からのリテラルロードと照合し、システム空間内の対応する領域に再配置する 2 個のリテラルコンパレータ。
- コード空間からの命令フェッチと照合する 6 個の命令コンパレータ。命令コンパレータを使用すると、システム空間内の対応する領域に再配置したり、コアへのブレークポイント命令を生成したりできます。

47.13 DWT（データウォッチポイントトリガ）

DWT ユニットは 4 個のコンパレータから構成されており、次の用途に設定できます。

- ハードウェアウォッチポイント
- ETM へのトリガ
- PC サンプラ
- データアドレスサンプラ

DWT は、プロファイル情報を提供することもできます。このため、次の値を得るためのカウンタにアクセスできます。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット（LSU）の動作数
- スリープサイクル数
- CPI（命令あたりのクロック数）
- 割込みオーバーヘッドの回数

47.14 ITM（計測トレースマクロセル）

47.14.1 概要

ITM は、アプリケーション駆動のトレースソースであり、printf 方式のデバッグをサポートすることでオペレーティングシステム （OS）やアプリケーションのイベントをトレースし、システムの診断情報を発行します。ITM がパケットとして発行するトレース情報は、次のように生成できます。

- **ソフトウェアトレース**：ソフトウェアは、ITM スティムラスレジスタに直接書込みを行って、パケットを発行できます。
- **ハードウェアトレース**：DWT がパケットを生成し、ITM がそれを発行します。
- **タイムスタンプ**：タイムスタンプはパケットを基準にして発行されます。ITM は、タイムスタンプを生成するための 21 ビットカウンタを内蔵しています。カウンタは、FPU 搭載 Cortex®-M4 クロックまたはシリアルワイヤ ビューア（SWV）出力のビットクロックレートによって駆動されます。

ITM によって発行されたパケットは、TPIU（トレースポートインタフェースユニット）に出力されます。TPIU のフォーマッタは、いくつかのパケットを追加してから（TPIU を参照）、完全なパケットシーケンスをデバッガホストに出力します。

ITM をプログラミングまたは使用するには、その前に、デバッグ例外／モニタ制御レジスタの TRCEN ビットを有効にする必要があります。

47.14.2 タイムスタンプパケット、同期およびオーバーフローパケット

タイムスタンプパケットは、タイムスタンプ情報や一般的な制御および同期情報をコード化します。そのために使用される 21 ビットのタイムスタンプカウンタ（プリスケーラ付きの場合もあります）は、タイムスタンプパケットが発行されるたびにリセットされます。このカウンタは、CPU クロックまたは SWV クロックによって駆動できます。

同期パケットは 6 バイトの値 0x80_00_00_00_00_00 から構成され、TPIU には 00 00 00 00 00 80 として LSB ファーストで発行されます。

同期パケットは、タイムスタンプ制御のためのパケットで、DWT トリガのたびに発行されます。

このため、DWT 制御レジスタの CYCCNTENA（ビット 0）をセットして、ITM をトリガするように DWT を設定する必要があります。さらに、ITM トレース制御レジスタのビット2（SYNCENA）もセットする必要があります。

注： **SYNENA ビットがセットされていない場合、DWT は TPIU への同期トリガを生成し、TPIU 同期パケットのみを送信して、ITM 同期パケットは送信しません。**

オーバーフロー･パケットは、データの書き込んだものの FIFO がフルであったことを示す特殊なタイムスタンプ･パケットで構成されます。

表 444. 主な ITM レジスタ

アドレス	レジスタ	詳細
@E0000FB0	ITM ロック･アクセス	他の ITM レジスタへの書込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
@E0000E80	ITM トレース制御	ビット 31-24 = 常に 0
		ビット 23 = ビジー
		ビット 22-16 = トレースデータのソースを識別する 7 ビットの ATB ID
		ビット 15-10 = 常に 0
		ビット 9:8 = TSPrescale = タイムスタンププリスケーラ
		ビット 7-5 = 予約済み
		ビット 4 = SWOENA = SWV 動作（タイムスタンプカウンタを SWV クロックによって駆動）を有効にします。
		ビット 3 = DWTENA：DWT スティムラスを有効にします。
		ビット 2 = SYNCENA：DWT が同期トリガを生成して TPIU が同期パケットを発行できるようにするには、このビットを 1 にする必要があります。
		ビット 1 = TSENA（タイムスタンプ有効）
		ビット 0 = ITMENA：ITM のグローバル有効ビット
@E0000E40	ITM トレース特権	ビット 3：ポート 31:24 のトレースを有効にするマスク
		ビット 2：ポート 23:16 のトレースを有効にするマスク
		ビット 1：ポート 15:8 のトレースを有効にするマスク
		ビット 0：ポート 7:0 のトレースを有効にするマスク
@E0000E00	ITM トレース有効	各ビットは、トレースを生成するために対応するスティムラスポートを有効にします。
@E0000000～E000007C	スティムラスポートレジスタ 0-31	選択されたスティムラスポート（32 個まで）に 32 ビットデータを書き込んでトレース出力します。

設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、DBGMCU_CR を設定することで TRACE I/O を割り当てます（セクション 47.17.2：TRACE ピンの割当て および セクション 47.16.3：デバッグ MCU 設定レジスタ（DBGMCU_CR）を参照）。
- ITM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ITM レジスタへの書込みアクセスをロック解除します。
- ITM トレース制御レジスタに 0x00010005 を書き込んで、同期を有効にした状態で ITM を有効にし、ATB ID を 0x00 以外の値にします。
- ITM トレース有効レジスタに 0x1 を書き込んで、スティムラスポート 0 を有効にします。
- ITM トレース特権レジスタに 0x1 を書き込んで、スティムラスポート 7:0 をマスク解除します。
- スティムラスポートレジスタ 0 に出力値を書き込みます。これはソフトウェアで行うことができます（printf 機能を使用）。

47.15 ETM（Embedded Trace Macrocell™）

47.15.1 概要

ETM はプログラム実行の再構築を有効にします。データはデータウォッチポイントおよびトレース（DWT）コンポーネントまたは計測トレースマクロセル（ITM）を使用してトレースされます。一方、命令は組み込みトレースマクロセル（ETM）を使用してトレースされます。

ETMは、情報をパケットとして送信し、埋め込みリソースによってトリガされます。これらのリソースは個別にプログラミングする必要があり、トリガソースの選択にはトリガイベントレジスタ（0xE0041008）を使用します。イベントは、単一イベント（アドレスコンパレータからのアドレスマッチ）または 2 つのイベント間の論理式とすることができます。トリガソースは、DWT モジュールの 4 個のコンパレータの 1 つで、次のイベントを監視できます。

- クロックサイクルマッチング
- データアドレスマッチング

トリガリソースの詳細については、セクション 47.13 : DWT（データウォッチポイントトリガ）を参照してください。

ETM によって送信されたパケットは、TPIU（トレースポートインタフェースユニット）に出力されます。TPIU のフォーマッタは、いくつかのパケットを追加してから（ セクション 47.17 : TPIU（トレースポートインタフェースユニット）を参照）、完全なパケットシーケンスをデバッガホストに出力します。

47.15.2 信号プロトコル、パケットタイプ

これについては、Arm® IHI 0014N ドキュメントの第 7 セクション「ETMv3 信号プロトコル」を参照してください。

47.15.3 主な ETM レジスタ

レジスタの詳細については、Arm® IHI 0014N 仕様の第 3 章を参照してください。

表 445. 主な ETM レジスタ

アドレス	レジスタ	詳細
0xE0041FB0	ETM ロック･アクセス	他の ETM レジスタへの書込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
0xE0041000	ETM 制御	このレジスタは、ETM の一般的な動作（たとえば、トレースを有効にする方法）を制御します。
0xE0041010	ETM ステータス	このレジスタは、トレースおよびトリガロジックの現在のステータスに関する情報を提供します。
0xE0041008	ETM トリガ･イベント	このレジスタは、トリガを制御するイベントを定義します。
0xE004101C	ETM トレース有効制御	このレジスタは、どのコンパレータが選択されるかを定義します。
0xE0041020	ETM トレース有効イベント	このレジスタは、トレース有効化イベントを定義します。
0xE0041024	ETM トレース開始／停止	このレジスタは、トレースをそれぞれ開始および停止するためにトリガソースによって使用されるトレースを定義します。

47.15.4 設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、I/IO_TRACEN を有効にして STM32G4 シリーズ デバッグ設定レジスタで TRACE I/O を割り当てます。
- ETM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ITM レジスタへの書込みアクセスをロック解除します。
- 制御レジスタに 0x00001D1E を書き込みます（トレースを設定します）。
- トリガイベントレジスタに 0000406F を書き込みます（トリガイベントを定義します）。
- トレース有効イベントレジスタに 0000006F を書き込みます（開始／停止するイベントを定義します）。
- トレース開始／停止レジスタに 00000001 を書き込みます（トレースを有効にします）。
- ETM 制御レジスタに 0000191E を書き込みます（設定の終了）。

47.16 MCU デバッグコンポーネント（DBGMCU）

MCU デバッグコンポーネントは、デバッガによる以下のサポート機能を支援します。

- 低電力モード
- ブレークポイントにおける、タイマ、ウォッチドッグ、I^2C、および bxCAN のクロック制御
- TRACE ピンの割当て制御

47.16.1 低電力モードのデバッグサポート

低電力モードに入るには、WFI または WFE 命令を実行する必要があります。

MCU はいくつかの低電力モードを実装しており、CPU クロックを無効にしたり、CPU の消費電力を低減したりすることができます。

デバッグセッション中には、コアは FCLK や HCLK をオフにすることはできません。デバッグ時のデバッガ接続に必要なため、これらをアクティブな状態に保つ必要があります。MCU は、ユーザが低電力モードでソフトウェアをデバッグするための特殊な手段を備えています。

このため、デバッガホストは、最初にいくつかのデバッグ設定レジスタをセットして、低電力モード動作を変更する必要があります。

- SLEEP モードでは、DBGMCU_CR レジスタの DBG_SLEEP ビットをデバッガによって事前にセットする必要があります。これによって、HCLK には FCLK と同じクロックが供給されます（システム･クロックはソフトウェアによって事前に設定されています）。
- STOP モードでは、DBG_STOP ビットをデバッガによって事前にセットする必要があります。これによって、内部 RC オシレータ･クロックが、STOP モードで FCLK と HCLK を供給できます。
- STANDBY モードでは、DBG_STANDBY ビットをデバッガによって事前にセットする必要があります。これによって、レギュレータをオンにしたまま、内部 RC オシレータが、STANDBY モードで FCLK と HCLK にクロックを供給できます。システム･リセットが内部的に生成されるため、STANDBY モードから抜けることは、リセットからのフェッチと同じになります。

システムリセット中、DBGMCU_CR レジスタはデバッガによって書込みが可能です。デバッガホストがこれらの機能をサポートしない場合でも、ソフトウェアによってこのレジスタへの書込みが可能です。

47.16.2 タイマ、RTC、ウォッチドッグ、および I²C のデバッグサポート

ブレークポイントにおいては、以下に示すタイマのカウンタ、RTC やウォッチドッグの動作方法を選択する必要があります。

- ブレークポイントの中でもカウントを継続できます。この動作は、たとえば、PWM がモータを制御しているときに一般的に必要です。
- ブレークポイントの中でカウントを停止できます。この動作はウォッチドッグ用に必要です。

I^2C の場合、ブレークポイントにおいては SMBUS タイムアウトをブロックするように選択できます。

システムリセット中、DBGMCU フリーズレジスタはデバッガによって書込みが可能です。デバッガホストがこれらの機能をサポートしない場合でも、ソフトウェアによってこれらのレジスタへの書込みが可能です。

47.16.3 デバッグ MCU 設定レジスタ（DBGMCU_CR）

アドレス：0xE004 2004

パワーオン･リセット：0x0000 0000

システム･リセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE [1:0]		TRACE_IOEN	Res.	Res.	DBG_STANDBY	DBG_STOP	DBG_SLEEP
								rw	rw	rw			rw	rw	rw

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:5 **TRACE_MODE[1:0] および TRACE_IOEN**：TRACE ピン割り当て制御

– TRACE_IOEN=0 の場合：

TRACE_MODE=xx：TRACE ピンは割り当てられません（デフォルト）。

– TRACE_IOEN=1 の場合：

  – TRACE_MODE=00：非同期モードの TRACE ピン割り当てです。
  – TRACE_MODE=01：TRACEDATA サイズが 1 の同期モードの TRACE ピン割り当てです。
  – TRACE_MODE=10：TRACEDATA サイズが 2 の同期モードの TRACE ピン割り当てです。
  – TRACE_MODE=11：TRACEDATA サイズが 4 の同期モードの TRACE ピン割り当てです。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DBG_STANDBY**：デバッグ STANDBY モード

0：（FCLK = オフ、HCLK = オフ）デジタル部全体が電源オフになります。
ソフトウェアから見て、STANDBY モードから抜けることは、リセットベクタのフェッチと同じになります（ただし、いくつかのステータスビットは MCU が STANDBY モードから再開していることを示す）。
1：（FCLK = オン、HCLK = オン）この場合、デジタル部は電源オフ状態ではなく、アクティブ状態を維持する内部 RC オシレータから FCLK と HCLK が供給されます。さらに、MCU は STANDBY モード中にシステムリセットを生成するため、STANDBY モードから抜けることはリセットからのフェッチと同じになります。

ビット 1 **DBG_STOP**：デバッグ STOP モード

0：（FCLK = オフ、HCLK = オフ）STOP モードでは、クロック･コントローラがすべてのクロック（HCLKと FCLK を含む）を無効にします。STOP モードから抜けると、クロック設定はリセット後の場合と同じになります（CPU は 8 MHz の内部 RC オシレータ（HSI16）から供給されます）。したがって、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります。

1：（FCLK = オン、HCLK = オン）この場合、STOP モードに入ると、STOP モードでもアクティブ状態を維持する内部 RC オシレータから FCLK と HCLK が供給されます。STOP モードから抜けるとき、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります（DBG_STOP=0 の場合と同様です）。

ビット 0 **DBG_SLEEP**：デバッグ SLEEP モード

0：（FCLK = オン、HCLK = オフ）SLEEP モードでは、FCLK はソフトウェアによって事前に設定されたシステム･クロックによって駆動され、HCLK は無効にされます。

SLEEP モードでは、クロックコントローラの設定はリセットされず、事前にプログラミングされた状態のままです。したがって、SLEEP モードから抜けるときに、ソフトウェアでクロックコントローラを再設定する必要はありません。

1：（FCLK = オン、HCLK = オン）この場合、SLEEP モードに入ると、HCLK には FCLK と同じクロック（ソフトウェアによって事前に設定されたシステム･クロック）が供給されます。

47.16.4 デバッグ マイクロコントローラ APB1 フリーズ･レジスタ 1（DBGMCU_APB1FZR1）

アドレス：0xE004 2008

パワーオン･リセット（POR）：0x0000 0000

システム･リセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DBG_LPTIM1_STOP	DBG_I2C3_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.
rw	rw								rw	rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_RTC_STOP	Res.	Res.	Res.	Res.	DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
			rw	rw	rw					rw	rw	rw	rw	rw	rw

ビット 31 **DBG_LPTIM1_STOP**：コア停止時に LPTIM1 カウンタは停止

0：コアが停止しても LPTIM1 カウンタのクロックは供給されます。

1：コア停止時に LPTIM1 カウンタのクロックは停止します。

ビット 30 **DBG_I2C3_STOP**：コア停止時に I2C3 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。

1：I2C3 SMBus タイムアウトは停止します。

ビット 29:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **DBG_I2C2_STOP**：コア停止時に I2C2 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。

1：I2C2 SMBus タイムアウトは停止します。

ビット 21 **DBG_I2C1_STOP**：コア停止時に I2C1 SMBUS タイムアウトカウンタは停止します。
0：通常モードと同じ動作です。
1：I2C1 SMBus タイムアウトは停止します。

ビット 20:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DBG_IWDG_STOP**：コア停止時に独立型ウォッチドッグカウンタは停止します。
0：コアが停止しても独立型ウォッチドッグ･カウンタのクロックは継続されます。
1：コア停止時に独立型ウォッチドッグ･カウンタのクロックは停止します。

ビット 11 **DBG_WWDG_STOP**：コア停止時にウィンドウ型ウォッチドッグカウンタは停止します。
0：コアが停止してもウィンドウ型ウォッチドッグ･カウンタのクロックは継続されます。
1：コア停止時にウィンドウ型ウォッチドッグ･カウンタのクロックは停止します。

ビット 10 **DBG_RTC_STOP**：コア停止時に RTC カウンタは停止します。
0：コアが停止しても RTC カウンタのクロックは供給されます。
1：コア停止時に RTC カウンタのクロックは停止します。

ビット 9:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **DBG_TIM7_STOP**：コア停止時に TIM7 カウンタは停止
0：コアが停止しても TIM7 カウンタのクロックは供給されます。
1：コア停止時に TIM7 カウンタのクロックは停止します。

ビット 4 **DBG_TIM6_STOP**：コア停止時に TIM6 カウンタは停止
0：コアが停止しても TIM6 カウンタのクロックは供給されます。
1：コア停止時に TIM6 カウンタのクロックは停止します。

ビット 3 **DBG_TIM5_STOP**：コア停止時に TIM5 カウンタは停止
0：コアが停止しても TIM5 カウンタのクロックは供給されます。
1：コア停止時に TIM5 カウンタのクロックは停止します。

ビット 2 **DBG_TIM4_STOP**：コア停止時に TIM4 カウンタは停止
0：コアが停止しても TIM4 カウンタのクロックは供給されます。
1：コア停止時に TIM4 カウンタのクロックは停止します。

ビット 1 **DBG_TIM3_STOP**：コア停止時に TIM3 カウンタは停止
0：コアが停止しても TIM3 カウンタのクロックは供給されます。
1：コア停止時に TIM3 カウンタのクロックは停止します。

ビット 0 **DBG_TIM2_STOP**：コア停止時に TIM2 カウンタは停止
0：コアが停止しても TIM2 カウンタのクロックは供給されます。
1：コア停止時に TIM2 カウンタのクロックは停止します。

47.16.5 デバッグ MCU APB1 フリーズレジスタ 2（DBGMCU_APB1FZR2）

アドレス：0xE004 200C

パワーオンリセット（POR）：0x0000 0000

システム･リセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
														rw	

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **DBG_I2C4_STOP**：コア停止時に I2C4 SMBUS タイムアウトカウンタは停止します。

0：通常モードと同じ動作です。

1：I2C4 SMBus タイムアウトは停止します。

ビット 0 予約済みであり、リセット値に保持する必要があります。

47.16.6 デバッグ MCU APB2 凍結レジスタ（DBGMCU_APB2FZR）

アドレス：0xE004 2010

パワーオン･リセット（POR）：0x0000 0000

システム･リセット：影響なし

アクセス：32 ビットアクセスのみサポートされます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	Res.	Res.	Res.	Res.	DBG_HRTIM_STOP	Res.	Res.	Res.	Res.	Res.	DBG_TIM20_STOP	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP
					rw						rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
		rw		rw											

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **DBG_HRTIM_STOP**：コア停止時に HRTIM カウンタは停止します。

0：コアが停止しても HRTIM カウンタのクロックは供給されます。

1：コア停止時に HRTIM カウンタのクロックは停止します。

ビット 25:21 予約済みであり、リセット値に保持する必要があります。

ビット 20 **DBG_TIM20_STOP**：コア停止時に TIM20 カウンタは停止

0：コアが停止しても TIM20 カウンタのクロックは供給されます。

1：コア停止時に TIM20 カウンタのクロックは停止します。

Bit19 予約済みであり、リセット値に保持する必要があります。

ビット 18 **DBG_TIM17_STOP**：コア停止時に TIM17 カウンタは停止

0：コアが停止しても TIM17 カウンタのクロックは供給されます。
1：コア停止時に TIM17 カウンタのクロックは停止します。

ビット 17 **DBG_TIM16_STOP**：コア停止時に TIM16 カウンタは停止

0：コアが停止しても TIM16 カウンタのクロックは供給されます。
1：コア停止時に TIM16 カウンタのクロックは停止します。

ビット 16 **DBG_TIM15_STOP**：コア停止時に TIM15 カウンタは停止

0：コアが停止しても TIM15 カウンタのクロックは供給されます。
1：コア停止時に TIM15 カウンタのクロックは停止します。

ビット 15:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **DBG_TIM8_STOP**：コア停止時に TIM8 カウンタは停止

0：コアが停止しても TIM8 カウンタのクロックは供給されます。
1：コア停止時に TIM8 カウンタのクロックは停止します。

ビット 12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **DBG_TIM1_STOP**：コア停止時に TIM1 カウンタは停止

0：コアが停止しても TIM1 カウンタのクロックは供給されます。
1：コア停止時に TIM1 カウンタのクロックは停止します。

ビット 10:0 予約済みであり、リセット値に保持する必要があります。

47.17 TPIU（トレースポートインタフェースユニット）

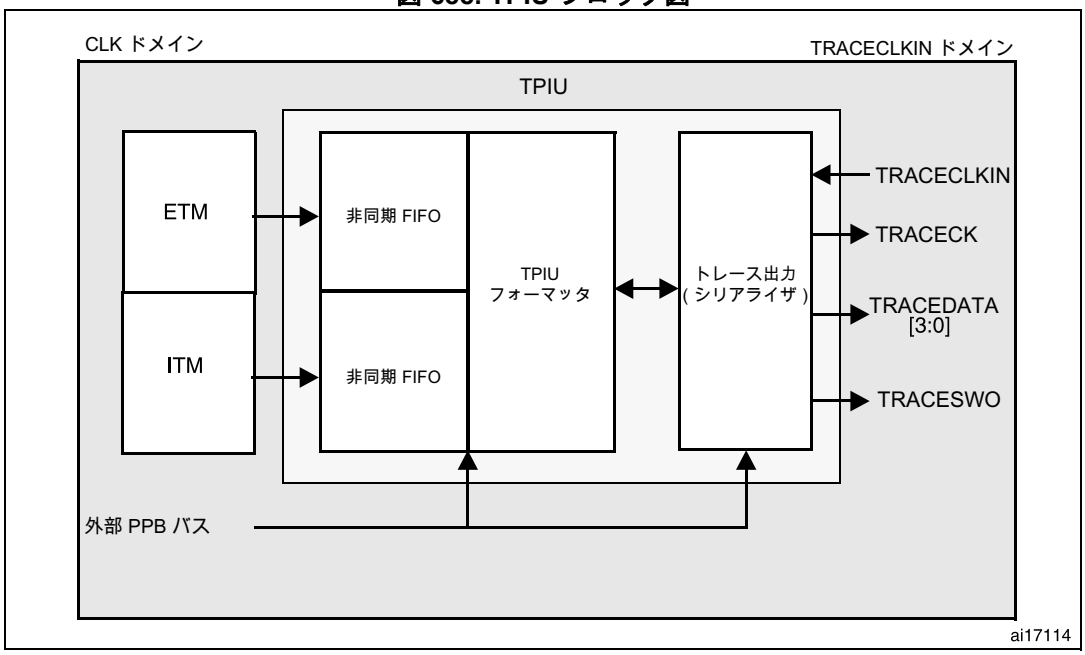
47.17.1 概要

TPIU は、ITM と ETM からのオンチップトレースデータ間のブリッジとしての役割を果たします。

出力データストリームによってカプセル化されたトレースソース ID は、トレースポートアナライザ（TPA）によってキャプチャされます。

コア には、特に低コストのデバッグ向けの単純な TPIU（特別バージョンの CoreSight TPIU から構成）が内蔵されています。

図 688. TPIU ブロック図



47.17.2 TRACE ピンの割当て

- 非同期モード

  非同期モードは、1 個のピンを追加で必要とし、すべてのパッケージで使用できます。非同期モードは、JTAG モードではなくシリアルワイヤモードを使用する場合にのみ使用できます。

表 446. 非同期 TRACE ピンの割当て

TPUI ピン名	TRACE 同期モード		STM32G4 シリーズ ピン割り当て
	タイプ	説明	
TRACESWO	O	TRACE 非同期データ出力	PB3

- 同期モード

  同期モードは、データトレースサイズに応じて 2～6 本のピンを追加で必要とし、大型パッケージでのみ使用できます。さらに、JTAG モードとシリアルワイヤモードでも使用でき、非同期トレースよりも優れた帯域幅出力機能を提供します。

表 447. 同期 TRACE ピンの割当て

TPUI ピン名	TRACE 同期モード		STM32G4 シリーズ ピン割り当て
	タイプ	説明	
TRACECK	O	TRACE クロック	PE2
TRACED[3:0]	O	TRACE 同期データ出力 1、2、または 4 とすることができます。	PE[6:3]

TPUI TRACE ピンの割り当て

デフォルトでは、これらのピンは割り当てられません。これらのピンを割り当てるには、デバッグ MCU 設定レジスタ（DBGMCU_CR）の TRACE_IOEN および TRACE_MODE ビットをセットします。この設定はデバッガホストで行う必要があります。

さらに、割り当てるピン数は、トレースの設定（非同期／同期）によって異なります。

- **非同期モード**：1 本のピンが追加で必要です。
- **同期モード**：データトレースポートレジスタのサイズ（1、2、または 4）に応じて、2～5 本のピンが追加で必要です。
  - TRACECK
  - TRACED(0)、ポートサイズが 1、2、または 4 に設定された場合
  - TRACED(1)、ポートサイズが 2 または 4 に設定された場合
  - TRACED(2)、ポートサイズが 4 に設定された場合
  - TRACED(3)、ポートサイズが 4 に設定された場合

TRACE ピンを割り当てるには、デバッガホストは、デバッグ MCU 設定レジスタ（DBGMCU_CR）の TRACE_IOEN ビットと TRACE_MODE[1:0] ビットをプログラミングする必要があります。デフォルトでは、TRACE ピンは割り当てられません。

このレジスタは、外部 PPB に配置され、PORESET によってリセットされます（システムリセットではありません）。このレジスタは、システムリセット中にデバッガによる書込みができます。

表 448. 柔軟性の高い TRACE ピン割り当て

DBGMCU_CR レジスタ		ピン割当て：	TRACE IO ピンの割当て					
TRACE_IOEN	TRACE_MODE [1:0]		PB3 / JTDO/ TRACESWO	PE2 / TRACECK	PE3 / TRACED[0]	PE4 / TRACED[1]	PE5 / TRACED[2]	PE6 / TRACED[3]
0	XX	トレースなし（デフォルト状態）	解放 (1)	-				
1	00	非同期トレース	TRACESWO	-	-	解放（GPIO として使用可能）		
1	01	同期トレース、1 ビット	解放 (1)	TRACECK	TRACED[0]	-	-	-
1	10	同期トレース、2 ビット		TRACECK	TRACED[0]	TRACED[1]	-	-
1	11	同期トレース、4 ビット		TRACECK	TRACED[0]	TRACED[1]	TRACED[2]	TRACED[3]

1. シリアルワイヤモードが使用されているときには解除されます。しかし、JTAG が使用されると、JTDO に割り当てられます。

注： **デフォルトでは、TPIU の TRACECLKIN 入力クロックは GND に接続されています。このクロックは、TRACE_IOEN ビットがセットされてから 2 クロックサイクル後に HCLK に割り当てられます。**

デバッガは、TPIU の SPP_R（選択ピンプロトコル）レジスタの PROTOCOL[1:0] ビットに書き込むことによって、トレースモードをプログラミングする必要があります。

- PROTOCOL=00：トレースポートモード（同期モード）。
- PROTOCOL = 01 または 10 : シリアルワイヤ（マンチェスタまたは NRZ）モード（非同期モード）。デフォルト状態は 01 です。

次に、TPIU の CPSPS_R（カレント同期化ポートサイズレジスタ）のビット [3:0] に書き込むことによって TRACE ポートサイズも設定します。

- 0x1：1 ピンの場合（デフォルト状態）
- 0x2：2 ピンの場合
- 0x8：4 ピンの場合

47.17.3 TPUI フォーマッタ

フォーマッタプロトコルは、データを 16 バイトフレームで出力します。

- 7 バイトのデータ
- 8 バイトの混合バイトであり、以下の構成になっています。
  - 1 ビット（LSB）：データバイト（”0”）または ID バイト（”1”）であることを示します。
  - 7 ビット（MSB）：データまたはソース ID トレースの変更とすることができます。
- 1 バイトの補助ビット：各ビットは 8 つの混合バイトのいずれかに対応します。
  - 対応するバイトがデータであった場合、このビットはデータのビット 0 を示します。
  - 対応するバイトが ID 変更であった場合、このビットはその ID 変更が有効になるタイミングを示します。

注： **詳細については、Arm® CoreSight Architecture Specification v1.0（Arm® IHI 0029B）を参照してください。**

47.17.4 TPUI フレーム同期パケット

TPUI は 2 種類の同期パケットを生成できます。

- フレーム同期パケット（フルワード同期パケット）

  これはワード0x7F_FF_FF_FF（LSB ファーストで発行）から構成されます。ID ソースコード 0x7F が使用されていない場合、このシーケンスは他のタイミングでは発生できません。

  このパケットはフレーム **間** で定期的に出力されます。

  連続モードでは、同期フレームが検出されると、TPA がこれらすべてのフレームを破棄する必要があります。

- ハーフワード同期パケット

  これはハーフワード0x7F_FF（LSB ファーストで発行）から構成されます。

  このパケットはフレーム **間または内** で定期的に出力されます。

  これらのパケットは連続モードでのみ生成され、TPA は TRACE ポートが IDLE モードである（TRACE はキャプチャされない）ことを検出できるようになります。TPA によって検出されたパケットは、破棄する必要があります。

47.17.5 同期フレームパケットの送信

コアの TPIU には、同期カウンタレジスタは実装されていません。したがって、同期トリガは **DWT**によってのみ生成できます。DWT 制御レジスタ（SYNCTAP[11:10] ビット）と DWT カレント PC サンプラサイクルカウントレジスタの説明を参照してください。

TPUI フレーム同期パケット（0x7F_FF_FF_FF）は次の場合に発行されます。

- 毎回の TPIU リセット解除後。このリセットは、TRACECLKIN クロックの立ち上がりエッジに同期して解除されます。つまり、DBGMCU_CFG レジスタの TRACE_IOEN ビットがセットされると、このパケットが送信されます。この場合、ワード 0x7F_FF_FF_FF の後に、フォーマットされたパケットは続きません。
- 毎回の DWT トリガ時（DWT は事前に設定されているものとします）。次の 2 つの場合があります。
  - ITM の SYNENA ビットがリセットされた場合、ワード 0x7F_FF_FF_FF だけが発行され、フォーマットされたストリームが続くことはありません。
  - ITM の SYNENA ビットがセットされた場合、TPUI によってフォーマット（トレース･ソース ID を追加）された ITM 同期パケットが続きます（0x80_00_00_00_00_00）。

47.17.6 同期モード

トレースデータの出力サイズは 4、2、または 1 ピンから選択できますTRACED[3:0]

出力クロックはデバッガに出力されます（TRACECK）。

なお、TRACECLKIN は内部で駆動され、TRACE が使用される場合にのみ HCLK に接続されます。

注： **同期モードでは、必ずしも安定したクロック周波数を提供する必要はありません。**

TRACE I/O（TRACECK を含む）は TRACLKIN（HCLK と同等）の立ち上がりエッジで駆動されます。したがって、TRACECK の出力周波数は HCLK/2 になります。

47.17.7 非同期モード

これは 1 ピン（非同期出力ピン TRACESWO）のみを使用してトレースを出力する低コストな方法ですが、明らかにその帯域幅は制限されます。

SW-DP ピンを使用する場合、TRACESWO ピンは JTDO ピンと多重化されます。このように、この機能はすべての STM32G4 シリーズ パッケージで使用できます。

この非同期モードでは、TRACECLKIN に一定の周波数が要求されます。標準的な UART（NRZ）のキャプチャ機構では、5 % の精度が必要です。マンチェスタエンコード方式では、10 % までの誤差が許されます。

47.17.8 STM32G4 シリーズ内部の TRACECLKIN 接続

STM32G4 シリーズでは、この TRACECLKIN 入力は内部で HCLK に接続されます。つまり、非同期トレースモードでは、アプリケーションは、CPU 周波数が安定しているタイムフレームの使用のみに制限されています。

注： **重要：非同期トレースを使用する場合は、以下のことに注意してください。**

STM32G4 シリーズ MCU のデフォルトクロックは、内部 RC オシレータです。リセット中の周波数はリセット解除後の周波数とは異なります。これは、RC 較正はシステムリセット中のデフォルト動作であり、システムリセット解除のたびに更新されるからです。

したがって、トレース･ポート･アナライザ（TPA）は、システム･リセット中に（TRACE_IOEN ビットによって）トレースを有効にするべきではありません。なぜなら、同期フレーム･パケットは、リセット解除後に送信されるトレース･パケットとは異なるビット時間で発行されるからです。

47.17.9 TPIU レジスタ

TPIU APB レジスタは、デバッグ例外／モニタ制御レジスタ（DEMCR）の TRCENA ビットがセットされた場合にのみ読出しと書込みが可能です。そうでない場合、これらのレジスタはゼロとして読み出されます（このビットの出力は TPIU の PCLK を有効にします）。

表 449. 重要な TPIU レジスタ

アドレス	レジスタ	説明
0xE0040004	現在のポート･サイズ	トレースポートサイズの選択： ビット 0：ポートサイズ = 1 ビット 1：ポートサイズ = 2 ビット 2：ポートサイズ = 3、サポートされません ビット 3：ポートサイズ = 4 1 ビットのみセットする必要があります。デフォルトでは、ポートサイズは 1 ビットです。（0x00000001）
0xE00400F0	選択ピン･プロトコル	トレースポートプロトコルの選択： ビット 1:0 = 00：同期トレース･ポート･モード 01：シリアル･ワイヤ出力 - マンチェスタ（デフォルト値） 10：シリアル･ワイヤ出力 - NRZ 11：予約済み
0xE0040304	フォーマッタおよび一掃制御	ビット 31-9 = 常に“0” ビット 8 = TrigIn = 常に“1”で、トリガが指定されたことを示します ビット 7-4 = 常に 0 ビット 3-2 = 常に 0 ビット 1 = EnFCont同期トレースモード（Select_Pin_Protocol レジスタのビット 1:0 = 00）では、このビットは強制的に“1”にされ、フォーマッタは自動的に連続モードで有効にされます。非同期モード（Select_Pin_Protocol レジスタのビット 1:0 <> 00）では、このビットの書込みによって、フォーマッタを有効／無効にできます。 ビット 0 = 常に“0” このレジスタのデフォルト値は 0x102 です。 **注：**同期モードでは、TRACECTL ピンがチップ外に配置されないため、フォーマッタは常に連続モードで有効になります。このように、フォーマッタは、トレースパケットの転送元を識別するための制御パケットを挿入します。
0xE0040300	フォーマッタおよび一掃ステータス	FPU 搭載 Cortex®-M4 では使用されず、常に 0x00000008 が読み出されます。

47.17.10 設定例

- デバッグ例外およびモニタ制御レジスタ（DEMCR）の TRCENA ビットをセットします。
- TPIU カレントポートサイズレジスタに希望する値を書き込みます（デフォルトは 0x1 で 1 ビットのポートサイズ）。
- TPIU フォーマッタおよび一掃制御レジスタに 0x102（デフォルト値）を書き込みます。
- TPIU 選択ピンプロトコルレジスタに書き込んで同期／非同期モードを選択します。例：非同期 NRZ モード（UART に類似）の場合は 0x2 を書き込みます。
- DBGMCU 制御レジスタに 0x20（IO_TRACEN ビット）を書き込み、TRACE I/O を非同期モードとします。このとき、TPIU 同期パケット（FF_FF_FF_7F）が発行されます。
- ITM を設定し、ITM スティムラスレジスタに書き込んで値を出力します。

47.18 DBG レジスタマップ

次の表にデバッグレジスタの一覧を示します。

表 450. DBG レジスタのマップとリセット値

アドレス	レジスタ	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0xE0042000	**DBGMCU_IDCODE**	REV_ID																Res.	Res.	Res.	Res.	DEV_ID											
	リセット値(1)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X					X	X	X	X	X	X	X	X	X	X	X	X
0xE0042004	**DBGMCU_CR**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	TRACE_MODE[1:0]		TRACE_IOEN	Res.	Res.	DBG_STANDBY	DBG_STOP	DBG_SLEEP
	リセット値																									0	0	0			0	0	0
0xE004 2008	**DBGMCU_APB1FZR1**	DBG_LPTIM1_STOP	DBG_I2C3_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C2_STOP	DBG_I2C1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_RTC_STOP	Res.	Res.	Res.		DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
	リセット値	0	0								0	0									0	0	0					0	0	0	0	0	0
0xE004 200C	**DBGMCU_APB1FZR2**	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	DBG_I2C4_STOP	Res.
	リセット値																																
0xE004 2010	**DBGMCU_APB2FZR**	Res.	Res.	Res.	Res.	Res.	DBG_HRTIM_STOP	Res.	Res.	Res.	Res.	Res.	DBG_TIM20_STOP	Res.	DBG_TIM17_STOP	DBG_TIM16_STOP	DBG_TIM15_STOP	Res.	Res.	DBG_TIM8_STOP	Res.	DBG_TIM1_STOP	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.
	リセット値						0						0		0	0	0			0		0											

1. リセット値は製品によって異なります。詳細については、セクション 47.6.1 : MCU デバイス ID コードを参照してください。

48 デバイス電子署名

デバイス電子署名は、Flashメモリモジュールのシステムメモリ領域に格納され、デバッグインタフェースまたは CPU を使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データおよび較正データを使用すれば、ユーザファームウェアやその他の外部デバイスは、STM32G4 シリーズ マイクロコントローラの特性に自動的に整合させることができます。

48.1 ユニークデバイス ID レジスタ（96 ビット）

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号（例：USB 文字列シリアル番号やその他のエンドアプリケーション）として使用
- 内部Flashメモリをプログラムする前に、このユニーク ID をソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、Flashメモリ内のコードのセキュリティを高めるためのセキュリティキーの一部として使用
- セキュアなブートプロセスなどの起動に使用

96 ビットのユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

ベースアドレス：0x1FFF 7590

アドレスオフセット：0x00

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[31:16]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **UID[31:0]**：ウェハ上の X および Y 座標

アドレスオフセット：0x04

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[63:48]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[47:32]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:8 **UID[63:40]：**LOT_NUM[23:0]
ロット番号（ASCII で符号化）

ビット 7:0 **UID[39:32]：**WAF_NUM[7:0]
ウェーハ番号（8 ビットの符号なし番号）

アドレスオフセット：0x08

読出し専用 = 0xXXXX XXXX（X は出荷時にプログラムされます）

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UID[95:80]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
UID[79:64]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **UID[95:64]：**LOT_NUM[55:24]
ロット番号（ASCII で符号化）

48.2 Flashサイズデータレジスタ

ベースアドレス：0x1FFF 75E0

アドレス･オフセット：0x00

読出し専用 = 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Flash_SIZE															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 15:0 **Flash_SIZE[15:0]**：Flash メモリ･サイズ
このビットフィールドは、デバイスのFlashメモリサイズを KB 単位で示します。
たとえば、0x040 は 64 KB に対応します。

48.3 パッケージデータレジスタ

ベースアドレス：0x1FFF 7500

アドレスオフセット：0x00

読出し専用 = 0xXXXX（X は出荷時にプログラムされます）

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	Res.	PKG[4:0]				
											r	r	r	r	r

ビット 15:5 予約済みであり、リセット値に保持する必要があります。

ビット 4:0 **PKG[4:0]**：パッケージ･タイプ

00000：LQFP64
00001：WLCSP64
00010：LQFP-100（すべてのデバイス）および LQFP-80（カテゴリ 2 および カテゴリ 3 デバイスの場合）
00101：WLCSP81
00111：LQFP-128 / UFBGA-121
01000：UFQFPN32
01001：LQFP32
01010：UFQFPN48
01011：LQFP48
01100：WLCSP49
01101：UFBGA64
01110：TFBGA100
10001：LQFP-80（カテゴリ 4 デバイスの場合のみ）
その他：予約済み

49 改版履歴

表 451. 文書改版履歴

日付	版	変更内容
2019 年 5 月 06 日	1	初版発行
2019 年 10 月 10 日	2	**このマニュアルにおける表記の規則のセクション** – 表 2 : 製品固有の機能を更新。 **システム·アーキテクチャのセクション** – セクション 2.1 : システムアーキテクチャ を更新（FMC を FSMC に置換）。 – 図 2 : メモリマップ を更新（FMC を FSMC に置換）。 **メモリ構成のセクション** – 表 3 : STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレスを更新。 – 表 2.4 : 内蔵 SRAM SRAM2 を更新（アドレス 0x2000 4000 に配置）。 – 表 7 : Flash モジュール - 512/256/128 KB のデュアル·バンク構成（読出し幅 64 ビット）を更新。 – 表 8 : Flash モジュール - 512/256/128 KB のシングル·バンク構成（読出し幅 128 ビット）を更新。 **電源制御のセクション** – セクション 6.1 : 電源を更新。 **リセットおよびクロック制御（RCC）のセクション** – 図 17 : クロックツリーを更新。 以下の FMC を FSMC に更新 : – セクション 7.4.10 : AHB3 ペリフェラルリセットレジスタ（RCC_AHB3RSTR） – セクション 7.4.16 : AHB3 ペリフェラル·クロック有効化レジスタ（RCC_AHB3ENR）を参照してください。 – セクション 7.4.22 : SLEEP モードおよび STOP モードにおける AHB3 ペリフェラル·クロック有効化レジスタ（RCC_AHB3SMENR）を参照してください。 **システム·コントローラ設定のセクション** – セクション 10.2.1 : SYSCFG メモリ再配置レジスタ（SYSCFG_MEMRMP）の ビット[2:0] の説明を更新。 **ペリフェラル相互接続マトリックスのセクション** – 表 82 : 相互接続 11 のオンチップ·ソース FLTxSRC[1:0] = 01 の列を更新。 **ネスト化されたベクタ割込みコントローラのセクション** – 表 97 : STM32G4 シリーズ ベクタテーブル を更新。 **フレキシブル·スタティック·メモリ·コントローラ（FSMC）のセクション** – セクション 19 : フレキシブル·スタティック·メモリ·コントローラ（FSMC）を更新。 **CORDIC コプロセッサ（CORDIC）のセクション** – セクション 17 : CORDIC コプロセッサ（CORDIC）を更新。 **アナログ·デジタル·コンバータ（ADC）のセクション** – セクション 21.4.33 : 内部電圧基準の監視を更新。 – セクション : サンプリング時間制御トリガモードを更新。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2019 年 10 月 10 日	2 （続き）	**デジタル･アナログ･コンバータ （D/AC）のセクション** – セクション 22 : D/A コンバータ （DAC）を更新。 **コンパレータのセクション** – セクション 24.6.2 : COMP レジスタマップを更新。 **AES ハードウェア･アクセラレータ （AES）のセクション** - セクション 34 : AES ハードウェア･アクセラレータ （AES）を更新。 **USB Power Delivery インタフェース （UCPD）のセクション** – セクション 46 : USB Type-C®/USB Power Delivery インタフェース （UCPD）を更新。 **デバッグのセクション** – セクション 47.4.2 : 柔軟性の高い SWJ-DP ピンの割り当てを更新。 – セクション 47.8.3 : SW-DP ステートマシン（リセット、アイドル状態、ID コード）ID コードを 0x2BA01477 に更新 – 表 448 : 柔軟性の高い TRACE ピン割り当てを更新。 – 表 436 : JTAG デバッグ･ポートのデータ･レジスタを更新。
2020年 3月 26日	3	カテゴリ 4 デバイス（STM32G491、STM32G4A1）を以下に追加。 – 表 1 : STM32G4 シリーズメモリ容量 – 表 2 : 製品固有の機能を参照してください。 – セクション 2.4 : 内蔵 SRAMを参照してください。 – セクション 2.4.1 : パリティチェックを参照してください。 – 表 4 : CCM SRAM の構成を参照してください。 – セクション 2.5 : Flashメモリの概要を参照してください。 – セクション 4 : 内蔵 Flash メモリ （Flash）カテゴリ 4 デバイスの場合を参照してください。 – 表 90 : DMAMUX の構成を参照してください。 **内蔵 Flash メモリ （FLASH）のセクション :** 更新 : – すべてのカテゴリの CPU クロック （HCLK）周波数によるウェイトステート数 の表 – 表 11 : オプション･バイトの構成 – 表 21 : オプションバイトの構成を参照してください。 – セクション 3.4.2 : オプション･バイトのプログラミング 「デュアル･バンク･モードの有効化（DBANK = 0 から DBANK = 1 への切替え）」のパラグラフ – セクション 4.7.6 : Flash制御レジスタ （Flash_CR）PNB[7:0] ビット –「ユーザ･オプション･バイトおよび読出し保護オプション･バイト」および セクション 4.7.8 : Flashオプションレジスタ （Flash_OPTR）レジスタの説明に PB4_PUPEN ビットを追加。 – セクション 5.7.14 : Flash レジスタマップ **電源制御のセクション :** 更新 : – 表 38 : レンジ 1 ブーストモード設定 SYSCLK の下限を削除。 – セクション 6.4.2 : 電源制御レジスタ 2 （PWR_CR2）PLS[2:0] ビットの説明。 **リセットおよびクロック制御 （RCC）のセクション :** 更新 : – 図 17 : クロックツリー – 表 51 : RCC レジスタのマップとリセット値を参照してください。 **ペリフェラル相互接続マトリックスのセクション :** – 表 60 : STM32G4 シリーズペリフェラル相互接続マトリックスを更新。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2020年 3月 26日	3 （続き）	**DMA リクエスト･マルチプレクサ（DMAMUX）のセクション：** – セクション 13.3.2：DMAMUX のマッピング を更新 **アナログ･デジタル･コンバータ（ADC）のセクション：** 更新： – セクション 21.2：ADC の主な機能 – 図 83：A/D コンバータのクロック構成を参照してください。 – 図 86：ADC3 接続性を参照してください。 – セクション 21.4.7：シングルエンド入力および差動入力のチャネルを参照してください。 **コンパレータのセクション：** 図 168：コンパレータブロック図 を更新。 **オペアンプ（OPAMP）のセクション：** 更新： – 表 201：オペアンプの可能な接続 – セクション 25.3.7：較正 の手順 – セクション 25.3.8：タイマによって制御されたマルチプレクサ･モード の手順 – セクション 25.5.1：OPAMP1 制御／ステータス･レジスタ（OPAMP1_CSR） – セクション 25.5.2：OPAMP2 制御／ステータス･レジスタ（OPAMP2_CSR）を参照してください。 – セクション 25.5.3：OPAMP3 制御／ステータス･レジスタ（OPAMP3_CSR）を参照してください。 – セクション 25.5.4：OPAMP4 制御／ステータス･レジスタ（OPAMP4_CSR）を参照してください。 – セクション 25.5.5：OPAMP5 制御／ステータス･レジスタ（OPAMP5_CSR）を参照してください。 – セクション 25.5.6：OPAMP6 制御／ステータス･レジスタ（OPAMP6_CSR）を参照してください。 **FD コントローラ･エリア･ネットワーク（FDCAN）のセクション：** セクション 44.4.7：FDCAN 公称ビットタイミングおよびプリスケーラレジスタ（FDCAN_NBTP）の注を更新。 **デバッグ･サポートのセクション：** 更新： – セクション 47.4.3：JTAG ピンの内部プルアップ／プルダウン – セクション 47.6.1：MCU デバイス ID コードを参照してください。 – セクション 47.6.2：バウンダリ･スキャン TAPを参照してください。 **デバイス電子署名のセクション：** – セクション 48.3：パッケージデータレジスタ を追加。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 4 月 14 日	4	**内蔵 Flash メモリ（FLASH）のセクション：** 更新： – 表 9：CPU クロック（HCLK）周波数によるウェイトステート数 – 表 19：CPU クロック（HCLK）周波数によるウェイトステート数を参照してください。 – 表 29：CPU クロック（HCLK）周波数によるウェイトステート数を参照してください。 – セクション 5.4.1：オプションバイトの説明 の「セキュリティ保護可能メモリ領域オプション･バイト」のパラグラフ – セクション 5.7.13：Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R） **リセットおよびクロック制御（RCC）のセクション：** 更新： – セクション 7.2.4：PLL – セクション 7.4.4：PLL 設定レジスタ（RCC_PLLCFGR）の PLLN[6:0] と PLLM[3:0] の説明 **高分解能タイマ（HRTIM）のセクション：** – セクション 27.3.1：概要 を更新。
2020 年 11 月 20 日	5	**メモリ･マップのセクション：** 表 3：STM32G4 シリーズ メモリマップとペリフェラルのレジスタ境界アドレス の USB SRAM を 1KB に更新。 図 2：メモリマップ を更新。 **内蔵 Flash メモリのセクション：** カテゴリ 3 デバイスの以下を更新： – NRST_MODE における「ユーザ･オプション･バイトおよび読出し保護オプション･バイト」のレジスタ･ビット 29.28 の名前 –「セキュリティ保護可能メモリ領域バンク 1 オプション･バイト」レジスタの BOOT_LOCK の説明の注意を削除。 – セクション 3.7.17：Flash セキュリティ保護可能領域バンク 1 レジスタ（FLASH_SEC1R）BOOT_LOCK の説明 カテゴリ 4 デバイスの以下を更新： – NRST_MODE における「ユーザ･オプション･バイトおよび読出し保護オプション･バイト」のレジスタ･ビット 29.28 の名前 –「セキュリティ保護可能メモリ領域オプション･バイト」レジスタの BOOT_LOCK の説明の注意を削除。 – セクション 4.7.1：Flash アクセス制御レジスタ（FLASH_ACR）のリセット値 – セクション 4.7.13：Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）BOOT_LOCK の説明 – セクション 4.7.14：Flash レジスタマップ カテゴリ 2 デバイスの以下を更新： – NRST_MODE における「ユーザ･オプション･バイトおよび読出し保護オプション･バイト」のレジスタ･ビット 29.28 の名前 –「セキュリティ保護可能メモリ領域オプション･バイト」レジスタの BOOT_LOCK の説明の注意を削除。 – セクション 5.7.13：Flash セキュリティ保護可能領域レジスタ（FLASH_SEC1R）BOOT_LOCK の説明 すべてのオプション･バイトの rw を r に更新。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2020 年 11 月 20 日	5 （続き）	RDP レベル 1 の代わりに RDP レベル 0 について、以下の表と図の「デフォルト」を「出荷時の値」に置換。 – 表 12 : Flashメモリの読出し保護ステータス – 図 4 : 読出し保護（RDP）レベルの変更を参照してください。 – 表 22 : Flashメモリの読出し保護ステータスを参照してください。 – 図 7 : 読出し保護（RDP）レベルの変更を参照してください。 – 表 32 : Flashメモリの読出し保護ステータスを参照してください。 – 図 10 : 読出し保護（RDP）レベルの変更を参照してください。 **電源制御（PWR）のセクション** : セクション 6.4.22 : 電源制御レジスタ （PWR_CR5）を更新。 **リセットおよびクロック制御（RCC）のセクション :** 更新 : – セクション 7.2.11 : ADC クロック – セクション 7.4.19 : APB2 ペリフェラルクロック有効レジスタ（RCC_APB2ENR） SPI4EN ビットの説明 **ペリフェラル相互接続マトリックスのセクション** : – 表 73 : 相互接続 2 を更新。 **DMA リクエスト･マルチプレクサ（DMAMUX）のセクション :** 更新 : – セクション 13.3.2 : DMAMUX のマッピング – セクション 13.6.1 : DMAMUX リクエスト･ライン･マルチプレクサ･チャネル x 設定レジスタ（DMAMUX_CxCR）を参照してください。 **ネスト化されたベクタ割込みコントローラ（NVIC）のセクション :** 表 97 : STM32G4 シリーズ ベクタテーブル を更新。 **アナログ･デジタル･コンバータ（ADC）のセクション :** セクション 21.2 : ADC の主な機能 を更新。 **高分解能タイマ（HRTIM）のセクション :** 更新 : – 表 222 : UDM ビットの設定によって変わる EExFLTR[3:0] コード – 表 223 : 外部イベントの機能を参照してください。 – セクション 27.3.8 : 外部イベントのグローバル調整を参照してください。 – セクション 27.3.9 : タイミングユニットでの外部イベントのフィルタリングを参照してください。 – セクション 27.5.28 : HRTIM タイマ x 出力 1 リセット･レジスタ（HRTIM_RSTx1R）（x = A ~ F）を参照してください。 – セクション 27.5.66 : HRTIM ADC トリガ 1 レジスタ（HRTIM_ADC1R）を参照してください。 – セクション 27.5.67 : HRTIM ADC トリガ 2 レジスタ（HRTIM_ADC2R）を参照してください。 – セクション 27.5.68 : HRTIM ADC トリガ 3 レジスタ （HRTIM_ADC3R）に注を追加。 **汎用タイマ（TIM2/TIM3/TIM4/TIM5）のセクション :** **表 286 : TIM2/TIM3/TIM4/TIM5レジスタマップとリセット値**を更新。 **汎用タイマ（TIM15/TIM16/TIM17）のセクション :** 更新 : – セクション 30.8.19 : TIMx オプション･レジスタ 1（TIMx_OR1）（x = 16～17）のアドレス･オフセットを 0x68 に。 – セクション 30.8.22 : TIM16/TIM17 レジスタマップ **デバッグ･サポート（DBG）のセクション :** – セクション 47.6.1 : MCU デバイス ID コード REV_ID[15:0] ビットの説明を更新。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2021 年 2 月 08 日	6	**内蔵 Flash メモリのセクション：** 更新： – セクション 3.5.3：書込み保護（WRP） – セクション 3.5.4：セキュリティ保護可能な領域を参照してください。 – セクション 3.7.1：Flash アクセス制御レジスタ（FLASH_ACR）のリセット値 – セクション 4.5.3：書込み保護（WRP） – セクション 4.5.4：セキュリティ保護可能な領域を参照してください。 – セクション 5.4.1：オプションバイトの説明 のセキュリティ保護可能メモリ領域オプション・バイトのパラグラフ – セクション 5.5.3：書込み保護（WRP） – セクション 5.5.4：セキュリティ保護可能な領域を参照してください。 – セクション 5.7.1：Flash アクセス制御レジスタ（FLASH_ACR）のリセット値 **フィルタ演算アクセラレータ（FMAC）のセクション：** セクション 18.4：FMAC レジスタを更新。 **アナログ・デジタル・コンバータ（ADC）のセクション：** 更新： – 表 178：A/D コンバータ・レジスタ・マップとリセット値（各 ADC、オフセット = 0x000 （マスタ A/D コンバータ）、0x100（スレーブ A/D コンバータ））ADC_CFGR2 のアドレス – セクション 21.7.6：A/D コンバータ・サンプル時間レジスタ 1（ADC_SMPR1）SMPPLUS ビット **オペアンプ（OPAMP）のセクション：** – 表 201：オペアンプの可能な接続 を更新。 **USB Type-C™ / USB Power Delivery インタフェース（UCPD）のセクション：** 更新： – セクション 46.7.3：UCPD 制御レジスタ（UCPD_CR） – セクション 46.7.15：UCPD レジスタ・マップを参照してください。 UCPD 設定レジスタ 3 （UCPD_CFGR3）を削除。 **デバッグ・サポート（DBG）のセクション：** セクション 47.4.2：柔軟性の高い SWJ-DP ピンの割り当て の注を削除して更新。

表 451. 文書改版履歴 （続き）

日付	版	変更内容
2022 年 2 月 04 日	7	更新： – 関連ドキュメント – セクション 2.2：メモリ構成 – 表 7：Flash モジュール - 512/256/128 KB のデュアル･バンク構成（読出し幅 64 ビット） – 表 8：Flash モジュール - 512/256/128 KB のシングル･バンク構成（読出し幅 128 ビット） – セクション 3：内蔵 Flash メモリ（Flash）：カテゴリ 3 デバイスの場合 – セクション 3.4.1：オプションバイトの説明 – セクション 3.7.5：Flashステータスレジスタ（Flash_SR） – セクション 3.7.8：Flash オプション･レジスタ（FLASH_OPTR） – セクション 4：内蔵 Flash メモリ（Flash）カテゴリ 4 デバイスの場合 – セクション 4.4.1：オプションバイトの説明 – セクション 4.7.5：Flashステータスレジスタ（Flash_SR） – セクション 4.7.8：Flashオプションレジスタ（Flash_OPTR） – 表 18：Flash モジュール - 256/512 KB の構成（読出し幅 64 ビット） – 表 28：Flash モジュール - 64/128 KB の構成（読出し幅 64 ビット） – セクション 5：内蔵 Flash メモリ（Flash）カテゴリ 2 デバイスの場合 – セクション 5.4.1：オプションバイトの説明 – セクション 5.7.5：Flashステータスレジスタ（Flash_SR） – セクション 5.7.8：Flashオプションレジスタ（Flash_OPTR） – セクション 6.2.3：ペリフェラル電圧モニタ（PVM） – セクション 6.4.2：電源制御レジスタ 2（PWR_CR2） – 表 51：RCC レジスタのマップとリセット値 – セクション 9.3.15：PB8 を GPIO として使用 – セクション 9.4.1：GPIO ポートモードレジスタ（GPIOx_MODER）（x = A ~ G） – セクション 10.2.2：SYSCFG 設定レジスタ 1（SYSCFG_CFGR1） – 表 97：STM32G4 シリーズ ベクタテーブル – セクション 18.4.7：FMAC ライト･データ･レジスタ（FMAC_WDATA） – セクション 18.4.8：FMAC 読出しデータ･レジスタ（FMAC_RDATA） – 図 151：レギュラ同時モードにおける DMA リクエスト（MDMA = 10） – セクション 24.6.1：コンパレータ x 制御/ステータスレジスタ（COMP_CxCSR） – セクション 24.6.2：COMP レジスタマップ – セクション 25.3.7：較正 – セクション 27.5.68：HRTIM ADC トリガ 3 レジスタ（HRTIM_ADC3R） – セクション 27.5.81：HRTIM フォールト入力レジスタ 4（HRTIM_FLTINR4） – 表 244：HRTIM レジスタマップとリセット値 - 共通機能 – セクション 39.5.9：データの送受信手順 – 図 642：I^2C スレーブ･レシーバの転送バス図 – 表 397：CAN サブシステム I/O 信号 – 図 669：メッセージ RAM 設定 – セクション 44.4.3：FDCAN データ･ビット･タイミングおよびプリスケーラ･レジスタ（FDCAN_DBTP） 追加： – セクション 21.5：低電力モードでのA/D コンバータ – 表 174：低電力モードが A/D コンバータに与える影響 – セクション 30.4.18：6 ステップ PWM 生成

表 452. 日本語版文書改版履歴

日付	版	変更内容
2022 年 8 月	1	日本語版　初版発行

索引

A

ADC_AWD2CR 700
ADC_AWD3CR 701
ADC_CALFACT 702
ADC_CCR 705
ADC_CDR 709
ADC_CFGR 681
ADC_CFGR2 685
ADC_CR 678
ADC_CSR 704
ADC_DIFSEL 701
ADC_DR 696
ADC_GCOMP 703
ADC_IER 676
ADC_ISR 674
ADC_JDRy 699
ADC_JSQR 696
ADC_OFRy 698
ADC_SMPR1 687
ADC_SMPR2 689
ADC_SQR1 692
ADC_SQR2 693
ADC_SQR3 694
ADC_SQR4 695
ADC_TR1 690
ADC_TR2 691
ADC_TR3 692
AES_CR 1497
AES_DINR 1501
AES_DOUTR 1502
AES_IVR0 1504
AES_IVR1 1505
AES_IVR2 1505
AES_IVR3 1505
AES_KEYR0 1503
AES_KEYR1 1503
AES_KEYR2 1504
AES_KEYR3 1504
AES_KEYR4 1506
AES_KEYR5 1506
AES_KEYR6 1506
AES_KEYR7 1507
AES_SR 1499
AES_SUSPxR 1507

C

COMP_C1CSR 769
COMP_C2CSR 769
COMP_C3CSR 769
COMP_C4CSR 769
COMP_C5CSR 769
COMP_C6CSR 769
COMP_C7CSR 769
CORDIC_CSR 474
CORDIC_RDATA 477
CORDIC_WDATA 476
CRC_CR 457
CRC_DR 456
CRC_IDR 456
CRC_INIT 458
CRC_POL 458
CRS_CFGR 342
CRS_CR 341
CRS_ICR 345
CRS_ISR 343

D

DAC_CCR 751
DAC_CR 740
DAC_DHR12L1 745
DAC_DHR12L2 746
DAC_DHR12LD 748
DAC_DHR12R1 744
DAC_DHR12R2 746
DAC_DHR12RD 747
DAC_DHR8R1 745
DAC_DHR8R2 747
DAC_DHR8RD 748
DAC_DOR1 749
DAC_DOR2 749
DAC_MCR 752
DAC_SHHR 755
DAC_SHRR 755
DAC_SHSR1 754
DAC_SHSR2 754
DAC_SR 750
DAC_STR1 756
DAC_STR2 757
DAC_SWTRGR 743
DACx_STMODR 757
DBGMCU_APB1FZR1 2065
DBGMCU_APB1FZR2 2067
DBGMCU_APB2FZR 2067
DBGMCU_CR 2064
DBGMCU_IDCODE 2051

DMA_CCRx 411
DMA_CMARx 415
DMA_CNDTRx 414
DMA_CPARx 414
DMA_IFCR 410
DMA_ISR 407
DMAMUX_CFR 429
DMAMUX_CSR 429
DMAMUX_CxCR 428
DMAMUX_RGCFR 431
DMAMUX_RGSR 431
DMAMUX_RGxCR 430

E

EXTI_EMR1 444
EXTI_EMR2 448
EXTI_FTSR1 445
EXTI_FTSR2 450
EXTI_IMR1 444
EXTI_IMR2 448
EXTI_PR1 447
EXTI_PR2 451
EXTI_RTSR1 445
EXTI_RTSR2 449
EXTI_SWIER1 446
EXTI_SWIER2 450

F

FDCAN_CCCR 1941
FDCAN_CKDIV 1969
FDCAN_CREL 1937
FDCAN_DBTP 1938
FDCAN_ECR 1947
FDCAN_ENDN 1937
FDCAN_HPMS 1958
FDCAN_IE 1953
FDCAN_ILE 1956
FDCAN_ILS 1955
FDCAN_IR 1950
FDCAN_NBTP 1943
FDCAN_PSR 1947
FDCAN_RWD 1940
FDCAN_RXF0A 1960
FDCAN_RXF0S 1959
FDCAN_RXF1A 1961
FDCAN_RXF1S 1960
FDCAN_RXGFC 1956
FDCAN_TDCR 1950
FDCAN_TEST 1939
FDCAN_TOCC 1945
FDCAN_TOCV 1946
FDCAN_TSCC 1944
FDCAN_TSCV 1945
FDCAN_TXBAR 1964
FDCAN_TXBC 1961
FDCAN_TXBCF 1965
FDCAN_TXBCIE 1966
FDCAN_TXBCR 1964
FDCAN_TXBRP 1963
FDCAN_TXBTIE 1966
FDCAN_TXBTO 1965
FDCAN_TXEFA 1968
FDCAN_TXEFS 1967
FDCAN_TXFQS 1962
FDCAN_XIDAM 1958
FLASH_ACR 127, 170, 209
FLASH_CR 132, 175, 214
FLASH_ECCR 134, 177, 216
FLASH_KEYR 129, 172, 211
FLASH_OPTKEYR 129, 172, 211
FLASH_OPTR 136, 178, 217
FLASH_PCROP1ER 138, 180, 219
FLASH_PCROP1SR 138, 180, 219
FLASH_PCROP2ER 141
FLASH_PCROP2SR 140
FLASH_PDKEYR 128, 171, 210
FLASH_SEC1R 142, 182, 221
FLASH_SEC2R 143
FLASH_SR 130, 173, 212
FLASH_WRP1AR 139, 181, 220
FLASH_WRP1BR 140, 181, 220
FLASH_WRP2AR 141
FLASH_WRP2BR 142
FMAC_CR 501
FMAC_PARAM 500
FMAC_RDATA 504
FMAC_SR 502
FMAC_WDATA 504
FMAC_X1BUFCFG 498
FMAC_X2BUFCFG 498
FMAC_YBUFCFG 499
FMC_BCRx 540
FMC_BTRx 543
FMC_BWTRx 545
FMC_ECCR 558
FMC_PATT 557
FMC_PCR 554
FMC_PCSCNTR 547
FMC_PMEM 556
FMC_SR 555

G

GPIOx_AFRH 363
GPIOx_AFRL 362
GPIOx_BRR 363
GPIOx_BSRR 360
GPIOx_IDR 359
GPIOx_LCKR 361
GPIOx_MODER 357
GPIOx_ODR 359
GPIOx_OSPEEDR 358
GPIOx_OTYPER 357
GPIOx_PUPDR 358

H

HRTIM_ADC1R 1024
HRTIM_ADC2R 1025
HRTIM_ADC3R 1025
HRTIM_ADC4R 1027
HRTIM_ADCER 1036
HRTIM_ADCPS1 1040
HRTIM_ADCPS2 1041
HRTIM_ADCUR 1039
HRTIM_BDMADR 1035
HRTIM_BDMUPR 1033
HRTIM_BDTxUPR 1034
HRTIM_BMCMPR 1019
HRTIM_BMCR 1015
HRTIM_BMPER 1019
HRTIM_BMTRGR 1017
HRTIM_CHPxR 990
HRTIM_CMP1CxR 963
HRTIM_CMP1xR 962
HRTIM_CMP2xR 964
HRTIM_CMP3xR 965
HRTIM_CMP4xR 966
HRTIM_CNTxR 959
HRTIM_CPT1ACR 991
HRTIM_CPT1BCR 991
HRTIM_CPT1CCR 991
HRTIM_CPT1DCR 992
HRTIM_CPT1ECR 992
HRTIM_CPT1FCR 992
HRTIM_CPT1xR 967
HRTIM_CPT2xCR 993
HRTIM_CPT2xR 968
HRTIM_CR1 1006
HRTIM_CR2 1007
HRTIM_DLLCR 1029
HRTIM_DTxR 969
HRTIM_EECR1 1020
HRTIM_EECR2 1022
HRTIM_EECR3 1023
HRTIM_EEFxR1 975
HRTIM_EEFxR2 977
HRTIM_FLTINR1 1030
HRTIM_FLTINR2 1031
HRTIM_FLTINR3 1042
HRTIM_FLTINR4 1044
HRTIM_FLTxR 1001
HRTIM_ICR 1010
HRTIM_IER 1011
HRTIM_ISR 1009
HRTIM_MCMP1R 946
HRTIM_MCMP2R 947
HRTIM_MCMP3R 947
HRTIM_MCMP4R 948
HRTIM_MCNTR 945
HRTIM_MCR 938
HRTIM_MDIER 943
HRTIM_MICR 942
HRTIM_MISR 941
HRTIM_MPER 945
HRTIM_MREP 946
HRTIM_ODISR 1013
HRTIM_ODSR 1014
HRTIM_OENR 1012
HRTIM_OUTxR 998
HRTIM_PERxR 960
HRTIM_REPxR 961
HRTIM_RSTAR 978
HRTIM_RSTBR 980
HRTIM_RSTCR 982
HRTIM_RSTDR 984
HRTIM_RSTER 986
HRTIM_RSTFR 988
HRTIM_RSTx1R 973
HRTIM_RSTx2R 974
HRTIM_SETx1R 971
HRTIM_SETx2R 973
HRTIM_TIMxCR 949
HRTIM_TIMxCR2 1002
HRTIM_TIMxDIER 956
HRTIM_TIMxEEFR3 1005
HRTIM_TIMxICR 955
HRTIM_TIMxISR 953

I

I2C_CR1 1877
I2C_CR2 1880
I2C_ICR 1888
I2C_ISR 1886
I2C_OAR1 1882

I2C_OAR2 1883
I2C_PECR 1889
I2C_RXDR 1890
I2C_TIMEOUTR 1885
I2C_TIMINGR 1884
I2C_TXDR 1890
IWDG_KR 1896
IWDG_PR 1897
IWDG_RLR 1898
IWDG_SR 1899
IWDG_WINR 1900

L

LPTIM_ARR 1454
LPTIM_CFGR 1450
LPTIM_CMP 1454
LPTIM_CNT 1455
LPTIM_CR 1453
LPTIM_ICR 1448
LPTIM_IER 1449
LPTIM_ISR 1447
LPTIM1_OR 1456
LPUART_BRR 1688
LPUART_CR1 . . . . . 1677, 1680
LPUART_CR2 1683
LPUART_CR3 1685
LPUART_ICR 1697
LPUART_ISR 1689, 1693
LPUART_PRESC 1699
LPUART_RDR 1698
LPUART_RQR 1688
LPUART_TDR 1698

O

OPAMP1_CSR 784
OPAMP2_CSR 787
OPAMP3_CSR 790
OPAMP4_CSR 793
OPAMP5_CSR 796
OPAMP6_CSR 799
OPAMPx_TCMR 802-804, 806-808

P

PWR_CR1 249
PWR_CR2 250
PWR_CR3 251
PWR_CR4 252
PWR_CR5 264
PWR_PDCRA 257
PWR_PDCRB 258
PWR_PDCRC 259
PWR_PDCRD 260
PWR_PDCRE 261
PWR_PDCRF 262
PWR_PDCRG 263
PWR_PUCRA 256
PWR_PUCRB 258
PWR_PUCRC 259
PWR_PUCRD 260
PWR_PUCRE 261
PWR_PUCRF 262
PWR_PUCRG 263
PWR_SCR 256
PWR_SR1 253
PWR_SR2 254

Q

QUADSPI_ABR 585
QUADSPI_AR 585
QUADSPI_CCR 582
QUADSPI_CR 576
QUADSPI_DCR 579
QUADSPI_DLR 582
QUADSPI_DR 586
QUADSPI_FCR 581
QUADSPI_LPTR 588
QUADSPI_PIR 588
QUADSPI_PSMAR 587
QUADSPI_PSMKR 587
QUADSPI_SR 580

R

RCC_AHB1ENR 303
RCC_AHB1RSTR 295
RCC_AHB1SMENR 311
RCC_AHB2ENR 304
RCC_AHB2RSTR 296
RCC_AHB2SMENR 313
RCC_AHB3ENR 306
RCC_AHB3RSTR 298
RCC_AHB3SMENR 315
RCC_APB1ENR1 306
RCC_APB1ENR2 309
RCC_APB1RSTR1 298
RCC_APB1RSTR2 301
RCC_APB1SMENR1 316
RCC_APB1SMENR2 319
RCC_APB2ENR 310
RCC_APB2RSTR 301
RCC_APB2SMENR 320
RCC_BDCR 326

RCC_CCIPR 322
RCC_CCIPR2 331
RCC_CFGR 286
RCC_CICR 294
RCC_CIER 291
RCC_CIFR 292
RCC_CR 284
RCC_CRRCR 330
RCC_CSR 328
RCC_ICSCR 286
RCC_PLLCFGR 288
RNG_CR 820
RNG_DR 823
RNG_SR 822
RTC_ALRMAR 1538
RTC_ALRMASSR 1539
RTC_ALRMBR 1540
RTC_ALRMBSSR 1541
RTC_CALR 1535
RTC_CR 1531
RTC_DR 1527
RTC_ICSR 1528
RTC_MISR 1543
RTC_PRER 1530
RTC_SCR 1544
RTC_SHIFTR 1536
RTC_SR 1542
RTC_SSR 1528
RTC_TR 1526
RTC_TSDR 1537
RTC_TSSSR 1538
RTC_TSTR 1537
RTC_WPR 1534
RTC_WUTR 1530

S

SAI_ACLRFR 1817
SAI_ACR1 1796
SAI_ACR2 1802
SAI_ADR 1819
SAI_AFRCR 1806
SAI_AIM 1810
SAI_ASLOTR 1808
SAI_ASR 1813
SAI_BCLRFR 1818
SAI_BCR1 1799
SAI_BCR2 1804
SAI_BDR 1819
SAI_BFRCR 1807
SAI_BIM 1812
SAI_BSLOTR 1809
SAI_BSR 1815
SAI_PDMCR 1820
SAI_PDMDLY 1821
SPIx_CR1 1748
SPIx_CR2 1750
SPIx_CRCPR 1754
SPIx_DR 1754
SPIx_I2SCFGR 1756
SPIx_I2SPR 1757
SPIx_RXCRCR 1755
SPIx_SR 1752
SPIx_TXCRCR 1755
SYSCFG_CFGR1 367
SYSCFG_CFGR2 375
SYSCFG_EXTICR1 369
SYSCFG_EXTICR2 370
SYSCFG_EXTICR3 371
SYSCFG_EXTICR4 373
SYSCFG_MEMRMP 366
SYSCFG_SCSR 374
SYSCFG_SKR 376
SYSCFG_SWPR 376

T

TAMP_BKPxR 1560
TAMP_CR1 1553
TAMP_CR2 1554
TAMP_FLTCR 1555
TAMP_IER 1556
TAMP_MISR 1558
TAMP_SCR 1559
TAMP_SR 1557
TIM15_AF1 1382
TIM15_AF2 1384
TIM15_ARR 1375
TIM15_BDTR 1377
TIM15_CCER 1371
TIM15_CCMR1 . . . . . 1367, 1369
TIM15_CCR1 1376
TIM15_CCR2 1377
TIM15_CNT 1374
TIM15_CR1 1359
TIM15_CR2 1360
TIM15_DCR 1385
TIM15_DIER 1363
TIM15_DMAR 1385
TIM15_DTR2 1380
TIM15_EGR 1366
TIM15_PSC 1374
TIM15_RCR 1375
TIM15_SMCR 1362

TIM15_SR 1364
TIM15_TISEL 1381
TIMx_AF1 . . . . . 1187, 1305, 1406
TIMx_AF2 . . . . . 1190, 1306, 1409
TIMx_ARR 1172, 1293-1294, 1400, 1432
TIMx_BDTR 1177, 1402
TIMx_CCER 1167, 1290, 1396
TIMx_CCMR1 . .1159, 1161, 1284, 1286, 1393-1394
TIMx_CCMR2 . . . . . 1163-1164, 1288-1289
TIMx_CCMR3 1183
TIMx_CCR1 . . . . . .1173, 1295-1296, 1401
TIMx_CCR2 . . . . . .1174, 1297-1298
TIMx_CCR3 . . . . . .1175, 1299-1300
TIMx_CCR4 . . . . . .1176, 1301-1302
TIMx_CCR5 1181
TIMx_CCR6 1182
TIMx_CNT 1171, 1292, 1399, 1431
TIMx_CR1 . . . . . 1145, 1274, 1388, 1428
TIMx_CR2 . . . . . 1147, 1276, 1389, 1429
TIMx_DCR 1193, 1307, 1410
TIMx_DIER 1154, 1280, 1390, 1430
TIMx_DMAR 1194, 1308, 1411
TIMx_DTR2 . . . . . .1184, 1405
TIMx_ECR 1185, 1303
TIMx_EGR 1158, 1283, 1392, 1431
TIMx_OR1 1409
TIMx_PSC 1171, 1293, 1399, 1432
TIMx_RCR 1172, 1400
TIMx_SMCR 1150, 1277
TIMx_SR 1155, 1281, 1391, 1430
TIMx_TISEL 1186, 1304, 1406

U

UCPD_CFGR1 2026
UCPD_CFGR2 2028
UCPD_CR 2029
UCPD_ICR 2036
UCPD_IMR 2031
UCPD_RX_ORDEXTR1 2041
UCPD_RX_ORDEXTR2 2041
UCPD_RX_ORDSETR 2039
UCPD_RX_PAYSZR 2040
UCPD_RXDR 2040
UCPD_SR 2033
UCPD_TX_ORDSETR 2037
UCPD_TX_PAYSZR 2038
UCPD_TXDR 2038
USART_BRR 1627
USART_CR1 . . . . . .1611, 1615
USART_CR2 1619
USART_CR3 1623
USART_GTPR 1628
USART_ICR 1642
USART_ISR 1631, 1637
USART_PRESC 1645
USART_RDR 1643
USART_RQR 1630
USART_RTOR 1629
USART_TDR 1644
USB_ADDRn_RX 2001
USB_ADDRn_TX 2000
USB_BCDR 1994
USB_BTABLE 1993
USB_CNTR 1987
USB_COUNTn_RX 2002
USB_COUNTn_TX 2001
USB_DADDR 1992
USB_EPnR 1995
USB_FNR 1992
USB_ISTR 1989
USB_LPMCSR 1993

V

VREFBUF_CCR 763
VREFBUF_CSR 762

W

WWDG_CFR 1906
WWDG_CR 1906
WWDG_SR 1907

重要なお知らせ（よくお読み下さい）

STMicroelectronics NV およびその子会社（以下、ST）は、ST製品及び本書の内容をいつでも予告なく変更、修正、改善、改定及び改良する権利を留保します。購入される方は、発注前にST製品に関する最新の関連情報を必ず入手してください。ST 製品は、注文請書発行時点で有効なST の販売条件に従って販売されます。

ST製品の選択並びに使用については購入される方が全ての責任を負うものとします。購入される方の製品上の操作や設計に関してST は一切の責任を負いません。

明示又は黙示を問わず、STは本書においていかなる知的財産権の実施権も許諾致しません。

本書で説明されている情報とは異なる条件でST 製品が再販された場合、その製品についてSTが与えたいかなる保証も無効となります。

ST およびST ロゴはSTMicroelectronics の商標です。STの登録商標についてはSTウェブサイトをご覧ください。www.st.com/trademarks その他の製品またはサービスの名称は、それぞれの所有者に帰属します。

本書の情報は本書の以前のバージョンで提供された全ての情報に優先し、これに代わるものです。

この資料は、STMicroelectronics NV 並びにその子会社(以下ST)が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くためにSTマイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST及びSTマイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

© 2023 STMicroelectronics - All rights reserved