

**RM0367**

リファレンスマニュアル

超低消費電力 ARM Cortex-M0+ コア搭載32ビットマイコン STM32L0x3

概要

このリファレンスマニュアルは、アプリケーション開発者を対象としています。STM32L0x3 マイクロコントローラメモリとペリフェラルを使用する方法について、詳しく説明しています。

STM32L0x3 は、さまざまなメモリサイズ、パッケージ、およびペリフェラルを持つマイクロコントローラファミリ製品になります。

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

ARM® Cortex®-M0+ コアについては、Cortex®-M0+ テクニカルリファレンスマニュアルを参照してください。

関連ドキュメント

- Cortex®-M0+テクニカルリファレンスマニュアル（www.arm.com から入手可能）。
- STM32L0x3 データシート。

目次

| | | |
|----------|---|-----------|
| 1 | このマニュアルにおける表記の規則 | 38 |
| 1.1 | レジスタに関する略記 | 38 |
| 1.2 | 用語 | 38 |
| 1.3 | 使用可能なペリフェラル | 39 |
| 1.4 | 製品カテゴリの定義 | 39 |
| 2 | システムおよびメモリの概要 | 40 |
| 2.1 | システムアーキテクチャ | 40 |
| 2.1.1 | S0 : Cortex バス | 41 |
| 2.1.2 | S1 : DMA バス | 41 |
| 2.1.3 | バスマトリックス | 41 |
| 2.2 | メモリ構成 | 42 |
| 2.2.1 | 概要 | 42 |
| 2.2.2 | メモリマップとレジスタ境界アドレス | 43 |
| 2.3 | 内蔵 SRAM | 46 |
| 2.4 | ブート設定 | 46 |
| 3 | フラッシュプログラムメモリとデータ EEPROM (FLASH) | 48 |
| 3.1 | 概要 | 48 |
| 3.2 | NVM の主な機能 | 48 |
| 3.3 | NVM の機能詳細 | 48 |
| 3.3.1 | NVM 構成 | 48 |
| 3.3.2 | NVM の読み出し | 49 |
| 3.3.3 | NVM の書き込み／消去 | 57 |
| 3.4 | メモリ保護 | 70 |
| 3.4.1 | RDP (読み出し保護) | 70 |
| 3.4.2 | PcROP (独自仕様コード読み出し保護) | 72 |
| 3.4.3 | 不要な書き込み／消去動作に対する保護 | 73 |
| 3.4.4 | 書き込み／消去の保護管理 | 75 |
| 3.4.5 | 保護エラー | 75 |
| 3.5 | NVM 割り込み | 76 |
| 3.5.1 | バスエラー (ハードフォールト) | 76 |

| | | |
|--------|-------------------------------------|-----|
| 3.6 | メモリアインタフェース管理 | 77 |
| 3.6.1 | 動作の優先順位と展開 | 77 |
| 3.6.2 | 動作シーケンス | 78 |
| 3.6.3 | 読み出し中にウェイトステート数を変更 | 79 |
| 3.6.4 | パワーダウン | 79 |
| 3.7 | フラッシュレジスタの説明 | 80 |
| 3.7.1 | アクセス制御レジスタ (FLASH_ACR) | 81 |
| 3.7.2 | プログラミングと消去制御レジスタ (FLASH_PECR) | 83 |
| 3.7.3 | パワーダウンキーレジスタ (FLASH_PDKEYR) | 86 |
| 3.7.4 | PECR アンロックキーレジスタ (FLASH_PEKEYR) | 86 |
| 3.7.5 | キーレジスタのプログラミングと消去 (FLASH_PRGKEYR) | 86 |
| 3.7.6 | オプションバイトアンロックキーレジスタ (FLASH_OPTKEYR) | 87 |
| 3.7.7 | ステータスレジスタ (FLASH_SR) | 88 |
| 3.7.8 | オプションバイトレジスタ (FLASH_OPTR) | 90 |
| 3.7.9 | 書込み保護レジスタ (FLASH_WRPROT) | 92 |
| 3.7.10 | フラッシュレジスタマップ | 93 |
| 3.8 | オプションバイト | 94 |
| 3.8.1 | オプションバイトの説明 | 94 |
| 3.8.2 | 保護フラグのロード中の不一致 | 95 |
| 3.8.3 | ソフトウェアによるオプションバイトの再ロード | 95 |
| 4 | 巡回冗長検査計算ユニット (CRC) | 96 |
| 4.1 | 概要 | 96 |
| 4.2 | CRC の主な機能 | 96 |
| 4.3 | CRC の機能説明 | 97 |
| 4.4 | CRC レジスタ | 98 |
| 4.4.1 | データレジスタ (CRC_DR) | 98 |
| 4.4.2 | 独立型データレジスタ (CRC_IDR) | 99 |
| 4.4.3 | 制御レジスタ (CRC_CR) | 99 |
| 4.4.4 | CRC の初期値 (CRC_INIT) | 100 |
| 4.4.5 | CRC 多項式 (CRC_POL) | 100 |
| 4.4.6 | CRC レジスタマップ | 101 |

| | | |
|----------|---|------------|
| 5 | ファイアウォール (FW) | 102 |
| 5.1 | 概要 | 102 |
| 5.2 | ファイアウォールの主な機能 | 102 |
| 5.3 | ファイアウォールの機能詳細 | 102 |
| 5.3.1 | ファイアウォールの AMBA バススヌープ | 102 |
| 5.3.2 | 機能要件 | 103 |
| 5.3.3 | ファイアウォールセグメント | 104 |
| 5.3.4 | セグメントへのアクセスと特性 | 105 |
| 5.3.5 | ファイアウォールの初期化 | 106 |
| 5.3.6 | ファイアウォールの状態 | 107 |
| 5.4 | ファイアウォールレジスタ | 109 |
| 5.4.1 | コードセグメントの開始アドレス (FW_CSSA) | 109 |
| 5.4.2 | コードセグメント長 (FW_CSL) | 110 |
| 5.4.3 | 不揮発性データセグメントの開始アドレス (FW_NVDSOA) | 110 |
| 5.4.4 | 不揮発性データセグメント長 (FW_NVDSL) | 111 |
| 5.4.5 | 揮発性データセグメントの開始アドレス (FW_VDSOA) | 111 |
| 5.4.6 | 揮発性データセグメント長 (FW_VDSL) | 112 |
| 5.4.7 | 設定レジスタ (FW_CR) | 112 |
| 5.4.8 | ファイアウォールレジスタマップ | 114 |
| 6 | 電源制御 (PWR) | 115 |
| 6.1 | 電源 | 115 |
| 6.1.1 | 独立した A/D および DAC コンバータ用電源と基準電圧 | 116 |
| 6.1.2 | 独立した LCD 電源 | 117 |
| 6.1.3 | RTC および RTC バックアップレジスタ | 117 |
| 6.1.4 | 電圧レギュレータ | 117 |
| 6.1.5 | ダイナミック電圧スケーリングの管理 | 118 |
| 6.1.6 | ダイナミック電圧スケーリングの設定 | 120 |
| 6.1.7 | VDD が 1.71 V を下回った時の電圧レギュレータとクロックの管理 | 121 |
| 6.1.8 | VCORE のレンジ修正時の電圧レギュレータとクロックの管理 | 121 |
| 6.1.9 | VDD の範囲が 1.71 V から 2.0 V である場合の電圧レンジと制限 | 121 |
| 6.2 | 電源供給スーパバイザ | 122 |
| 6.2.1 | パワーオンリセット (POR) / パワーダウンリセット (PDR) | 124 |
| 6.2.2 | ブラウンアウトリセット (BOR) | 124 |
| 6.2.3 | プログラム可能な電圧検出器 (PVD) | 125 |
| 6.2.4 | 内部基準電圧 (VREFINT) | 126 |

| | | |
|----------|--|------------|
| 6.3 | 低電力モード | 126 |
| 6.3.1 | 低電力モードでのクロックの動作 | 128 |
| 6.3.2 | システムクロックの低速化 | 128 |
| 6.3.3 | ペリフェラルクロックゲーティング | 128 |
| 6.3.4 | 低電力 RUN モード (LP RUN) | 129 |
| 6.3.5 | SLEEP モード | 129 |
| 6.3.6 | 低電力 SLEEP モード (LP SLEEP) | 131 |
| 6.3.7 | STOP モード | 133 |
| 6.3.8 | STANDBY モード | 134 |
| 6.3.9 | RTC およびコンパレータを使用した STOP モードおよび STANDBY モードからデバイスのウェイクアップ | 136 |
| 6.4 | 電源制御レジスタ | 138 |
| 6.4.1 | PWR 電源制御レジスタ (PWR_CR) | 138 |
| 6.4.2 | PWR 電源制御/ステータスレジスタ (PWR_PWR) | 141 |
| 6.4.3 | PWR レジスタマップ | 142 |
| 7 | リセットおよびクロック制御 (RCC) | 143 |
| 7.1 | リセット | 143 |
| 7.1.1 | システムリセット | 143 |
| 7.1.2 | 電源リセット | 144 |
| 7.1.3 | RTC およびバックアップレジスタリセット | 144 |
| 7.2 | クロック | 145 |
| 7.2.1 | HSE クロック | 148 |
| 7.2.2 | HSI16 クロック | 149 |
| 7.2.3 | MSI クロック | 150 |
| 7.2.4 | HSI48 クロック | 150 |
| 7.2.5 | PLL | 151 |
| 7.2.6 | LSE クロック | 151 |
| 7.2.7 | LSI クロック | 152 |
| 7.2.8 | システムクロック (SYSCLK) の選択 | 152 |
| 7.2.9 | システムクロックソースの周波数と電圧レンジ | 153 |
| 7.2.10 | HSE クロックセキュリティシステム (CSS) | 153 |
| 7.2.11 | LSE クロックセキュリティシステム | 153 |
| 7.2.12 | RTC および LCD クロック | 154 |
| 7.2.13 | ウォッチドッグクロック | 154 |
| 7.2.14 | クロック信号出力 | 154 |
| 7.2.15 | TIM21 を使用した内部/外部クロックの測定 | 155 |

| | | |
|--------|--|-----|
| 7.2.16 | TIM2/TIM21/TIM22 のクロック独立型システムクロックソース | 156 |
| 7.3 | RCC レジスタ | 157 |
| 7.3.1 | クロック制御レジスタ (RCC_CR) | 157 |
| 7.3.2 | 内部クロックソース較正レジスタ (RCC_ICSCR) | 160 |
| 7.3.3 | クロックリカバリ RC レジスタ (RCC_CRRCR) | 161 |
| 7.3.4 | クロック設定レジスタ (RCC_CFGR) | 161 |
| 7.3.5 | クロック割り込み有効レジスタ (RCC_CIER) | 164 |
| 7.3.6 | クロック割り込みフラグレジスタ (RCC_CIFR) | 166 |
| 7.3.7 | クロック割り込みクリアレジスタ (RCC_CICR) | 167 |
| 7.3.8 | GPIO リセットレジスタ (RCC_IOPRSTR) | 169 |
| 7.3.9 | AHB ペリフェラルリセットレジスタ (RCC_AHBSTR) | 170 |
| 7.3.10 | APB2 ペリフェラルリセットレジスタ (RCC_APB2RSTR) | 171 |
| 7.3.11 | APB1 ペリフェラルリセットレジスタ (RCC_APB1RSTR) | 172 |
| 7.3.12 | GPIO クロック有効レジスタ (RCC_IOPENR) | 174 |
| 7.3.13 | AHB ペリフェラルクロック有効レジスタ (RCC_AHBENR) | 175 |
| 7.3.14 | APB2 ペリフェラルクロック有効レジスタ (RCC_APB2ENR) | 177 |
| 7.3.15 | APB1 ペリフェラルクロック有効レジスタ (RCC_APB1ENR) | 179 |
| 7.3.16 | SLEEP モード時 GPIO クロックイネーブルレジスタ (RCC_IOPSMENR) .. | 181 |
| 7.3.17 | SLEEP モード時 AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBSMENR) | 182 |
| 7.3.18 | SLEEP モード時 APB2 ペリフェラルクロックイネーブルレジスタ (RCC_APB2SMENR) | 183 |
| 7.3.19 | SLEEP モード時 APB1 ペリフェラルクロックイネーブルレジスタ (RCC_APB1SMENR) | 184 |
| 7.3.20 | クロック設定レジスタ (RCC_CCIPR) | 186 |
| 7.3.21 | 制御/ステータスレジスタ (RCC_CSR) | 187 |
| 7.3.22 | RCC レジスタマップ | 190 |
| 8 | クロックリカバリシステム (CRS) | 193 |
| 8.1 | 概要 | 193 |
| 8.2 | CRS の主な機能 | 193 |
| 8.3 | CRS の機能詳細 | 194 |
| 8.3.1 | CRS ブロック図 | 194 |
| 8.3.2 | 同期入力 | 194 |
| 8.3.3 | 周波数誤差測定 | 195 |
| 8.3.4 | 周波数誤差評価および自動トリミング | 196 |
| 8.3.5 | CRS の初期化と設定 | 196 |

| | | |
|----------|--|------------|
| 8.4 | CRS 低電力モード | 197 |
| 8.5 | CRS 割り込み | 197 |
| 8.6 | CRS レジスタ | 198 |
| 8.6.1 | CRS 制御レジスタ (CRS_CR) | 198 |
| 8.6.2 | CRS 設定レジスタ (CRS_CFGR) | 200 |
| 8.6.3 | CRS 割り込みおよびステータスレジスタ (CRS_ISR) | 201 |
| 8.6.4 | CRS 割り込みフラグクリアレジスタ (CRS_ICR) | 203 |
| 8.6.5 | CRS レジスタマップ | 203 |
| 9 | 汎用 I/O (GPIO) | 205 |
| 9.1 | 概要 | 205 |
| 9.2 | GPIO の主な機能 | 205 |
| 9.3 | GPIO の機能説明 | 205 |
| 9.3.1 | 汎用 I/O (GPIO) | 207 |
| 9.3.2 | I/O ピンオルタネート機能マルチプレクサと配置 | 208 |
| 9.3.3 | I/O ポート制御レジスタ | 208 |
| 9.3.4 | I/O ポートデータレジスタ | 209 |
| 9.3.5 | I/O データのビット単位の操作 | 209 |
| 9.3.6 | GPIO ロック機構 | 209 |
| 9.3.7 | I/O オルタネート機能の入力/出力 | 210 |
| 9.3.8 | 外部割り込み/ウェイクアップライン | 210 |
| 9.3.9 | 入力設定 | 210 |
| 9.3.10 | 出力設定 | 211 |
| 9.3.11 | オルタネート機能設定 | 212 |
| 9.3.12 | アナログ設定 | 213 |
| 9.3.13 | HSE または LSE オシレータのピンを GPIO として使用 | 213 |
| 9.3.14 | GPIO ピンを RTC 供給ドメインで使用 | 214 |
| 9.4 | GPIO レジスタ | 214 |
| 9.4.1 | GPIO ポートモードレジスタ (GPIOx_MODER) (x = A ~ D、H) | 214 |
| 9.4.2 | GPIO ポート出力タイプレジスタ (GPIOx_OTYPER) (x = A ~ D、H) | 215 |
| 9.4.3 | GPIO ポート出力スピードレジスタ (GPIOx_OSPEEDR) (x = A ~ D、H) | 215 |
| 9.4.4 | GPIO ポートプルアップ/プルダウンレジスタ (GPIOx_PUPDR) (x = A ~ D、H) | 216 |
| 9.4.5 | GPIO ポート入力データレジスタ (GPIOx_IDR) (x = A ~ D、H) | 216 |
| 9.4.6 | GPIO ポート出力データレジスタ (GPIOx_ODR) (x = A ~ D、H) | 217 |

| | | |
|-----------|--|------------|
| 9.4.7 | GPIO ポートビットセット/リセットレジスタ (GPIOx_BSRR) (x = A ~ D、H) | 217 |
| 9.4.8 | GPIO ポート設定ロックレジスタ (GPIOx_LCKR) (x = A ~ D、H) | 218 |
| 9.4.9 | GPIO オルタネート機能下位レジスタ (GPIOx_AFR1) (x = A ~ D、H) ... | 219 |
| 9.4.10 | GPIO オルタネート機能上位レジスタ (GPIOx_AFR2) (x = A ~ D、H) ... | 219 |
| 9.4.11 | GPIO ポートビットリセットレジスタ (GPIOx_BRR) (x = A ~ D、H) ... | 220 |
| 9.4.12 | GPIO レジスタマップ | 221 |
| 10 | システム設定コントローラ (SYSCFG) | 223 |
| 10.1 | 概要 | 223 |
| 10.2 | SYSCFG レジスタ | 224 |
| 10.2.1 | SYSCFG メモリ再配置レジスタ (SYSCFG_CFGR1) | 224 |
| 10.2.2 | SYSCFG ペリフェラルモード設定レジスタ (SYSCFG_CFGR2) | 225 |
| 10.2.3 | リファレンス制御およびステータスレジスタ (REF_CFGR3) | 227 |
| 10.2.4 | SYSCFG 外部割り込み設定レジスタ 1 (SYSCFG_EXTICR1) | 229 |
| 10.2.5 | SYSCFG 外部割り込み設定レジスタ 2 (SYSCFG_EXTICR2) | 229 |
| 10.2.6 | SYSCFG 外部割り込み設定レジスタ 3 (SYSCFG_EXTICR3) | 230 |
| 10.2.7 | SYSCFG 外部割り込み設定レジスタ 4 (SYSCFG_EXTICR4) | 230 |
| 10.2.8 | SYSCFG レジスタマップ | 231 |
| 11 | ダイレクトメモリアクセスコントローラ (DMA) | 232 |
| 11.1 | 概要 | 232 |
| 11.2 | DMA の主な機能 | 232 |
| 11.3 | DMA の機能説明 | 233 |
| 11.3.1 | DMA トランザクション | 233 |
| 11.3.2 | アービタ | 234 |
| 11.3.3 | DMA チャンネル | 234 |
| 11.3.4 | プログラム可能なデータ幅、データの整列、およびエンディアン | 236 |
| 11.3.5 | エラー管理 | 237 |
| 11.3.6 | DMA 割り込み | 237 |
| 11.3.7 | DMA リクエストマッピング | 238 |
| 11.4 | DMA レジスタ | 240 |
| 11.4.1 | DMA 割り込みステータスレジスタ (DMA_ISR) | 240 |
| 11.4.2 | DMA 割り込みフラグクリアレジスタ (DMA_IFCR) | 241 |
| 11.4.3 | DMA チャンネル x 設定レジスタ (DMA_CCRx) (x = 1 ~ 7、x = チャンネル番号) | 242 |

| | | |
|-----------|---|------------|
| 11.4.4 | DMA チャンネル x データ数レジスタ (DMA_CNDTRx) (x = 1 ~ 7、x = チャンネル番号) | 244 |
| 11.4.5 | DMA チャンネル x ペリフェラルアドレスレジスタ (DMA_CPARx) (x = 1 ~ 7、x = チャンネル番号) | 244 |
| 11.4.6 | DMA チャンネル x メモリアドレスレジスタ (DMA_CMARx) (x = 1 ~ 7、x = チャンネル番号) | 245 |
| 11.4.7 | DMA チャンネル選択レジスタ (DMA_CSELR) | 246 |
| 11.4.8 | DMA レジスタマップ | 248 |
| 12 | ネスト化されたベクタ割り込みコントローラ (NVIC) | 250 |
| 12.1 | 主な機能 | 250 |
| 12.2 | SysTick 較正值レジスタ | 250 |
| 12.3 | 割り込みベクタと例外ベクタ | 250 |
| 13 | 拡張割り込み／イベントコントローラ (EXTI) | 253 |
| 13.1 | 概要 | 253 |
| 13.2 | EXTI の主な機能 | 253 |
| 13.3 | EXTI の機能説明 | 253 |
| 13.3.1 | EXTI ブロック図 | 254 |
| 13.3.2 | ウェイクアップイベント管理 | 254 |
| 13.3.3 | ペリフェラル非同期割り込み | 255 |
| 13.3.4 | ハードウェア割り込みの選択 | 255 |
| 13.3.5 | ハードウェアイベントの選択 | 255 |
| 13.3.6 | ソフトウェア割り込み／イベントの選択 | 255 |
| 13.4 | EXTI 割り込み／イベントラインの配置 | 256 |
| 13.5 | EXTI レジスタ | 257 |
| 13.5.1 | EXTI 割り込みマスクレジスタ (EXTI_IMR) | 258 |
| 13.5.2 | EXTI イベントマスクレジスタ (EXTI_EMR) | 259 |
| 13.5.3 | EXTI 立ち上がりエッジトリガ選択レジスタ (EXTI_RTSTR) | 259 |
| 13.5.4 | 立ち下がりエッジトリガ選択レジスタ (EXTI_FTSR) | 260 |
| 13.5.5 | EXTI ソフトウェア割り込みイベントレジスタ (EXTI_SWIER) | 261 |
| 13.5.6 | EXTI ペンディングレジスタ (EXTI_PR) | 262 |
| 13.5.7 | EXTI レジスタマップ | 263 |

| | | |
|-----------|--|------------|
| 14 | アナログデジタルコンバータ (ADC) | 264 |
| 14.1 | 概要 | 264 |
| 14.2 | ADC の主な機能 | 265 |
| 14.3 | ADC ピンと内部信号 | 266 |
| 14.4 | ADC の機能詳細 | 267 |
| 14.4.1 | ADC 電圧レギュレータ (ADVREGEN) | 267 |
| 14.4.2 | 較正 (ADCAL) | 268 |
| 14.4.3 | ADC オン / オフ制御 (ADEN、ADDIS、ADRDY) | 270 |
| 14.4.4 | ADC クロック (CKMODE、PRESC[3:0]、LFMEN) | 271 |
| 14.4.5 | ADC の設定 | 273 |
| 14.4.6 | チャネル選択 (CHSEL、SCANDIR) | 273 |
| 14.4.7 | プログラム可能なサンプリング時間 (SMP) | 273 |
| 14.4.8 | シングル変換モード (CONT=0) | 274 |
| 14.4.9 | 連続変換モード (CONT=1) | 274 |
| 14.4.10 | 変換の開始 (ADSTART) | 275 |
| 14.4.11 | タイミング | 275 |
| 14.4.12 | 実行中の変換の停止 (ADSTP) | 276 |
| 14.5 | 外部トリガおよびトリガ極性での変換 (EXTSEL、EXTEN) | 277 |
| 14.5.1 | 不連続モード (DISCEN) | 278 |
| 14.5.2 | プログラム可能な分解能 (RES) - 高速変換モード | 278 |
| 14.5.3 | 変換の終了、サンプリングフェーズの終了 (EOC、EOSMP フラグ) | 279 |
| 14.5.4 | 変換シーケンスの終了 (EOSEQ フラグ) | 279 |
| 14.5.5 | タイミング図の例 (シングル / 連続モードハードウェア / ソフトウェアトリガ) | 279 |
| 14.6 | データ管理 | 281 |
| 14.6.1 | データレジスタおよびデータの配置 (ADC_DR、ALIGN) | 281 |
| 14.6.2 | ADC オーバーラン (OVR、OVRMOD) | 281 |
| 14.6.3 | DMA を使用しない変換データシーケンスの管理 | 282 |
| 14.6.4 | オーバーランなしでの DMA を使用しない変換データの管理 | 282 |
| 14.6.5 | DMA を使用した変換データの管理 | 282 |
| 14.7 | 低電力機能 | 284 |
| 14.7.1 | ウェイトモード変換 | 284 |
| 14.7.2 | オートオフモード (AUTOFF) | 284 |

| | | |
|-----------|--|------------|
| 14.8 | アナログウィンドウウォッチドッグ (AWDEN、AWDSGL、AWDCH、AWD_HTR/LTR、AWD) | 286 |
| 14.9 | オーバーサンプリング回路 | 287 |
| 14.9.1 | オーバーサンプリング時の ADC 動作モードのサポート | 289 |
| 14.9.2 | アナログウォッチドッグ | 289 |
| 14.9.3 | トリガモード | 289 |
| 14.10 | 温度センサと内部基準電圧 | 290 |
| 14.11 | VLCD 電圧監視 | 292 |
| 14.12 | ADC 割り込み | 293 |
| 14.13 | ADC レジスタ | 294 |
| 14.13.1 | ADC 割り込みおよびステータスレジスタ (ADC_ISR) | 294 |
| 14.13.2 | ADC 割り込み有効レジスタ (ADC_IER) | 295 |
| 14.13.3 | ADC 制御レジスタ (ADC_CR) | 297 |
| 14.13.4 | ADC 設定レジスタ 1 (ADC_CFGR1) | 299 |
| 14.13.5 | ADC 設定レジスタ 2 (ADC_CFGR2) | 302 |
| 14.13.6 | ADC サンプリング時間レジスタ (ADC_SMPR) | 303 |
| 14.13.7 | ADC ウォッチドッグ閾値レジスタ (ADC_TR) | 304 |
| 14.13.8 | ADC チャンネル選択レジスタ (ADC_CHSELR) | 305 |
| 14.13.9 | ADC データレジスタ (ADC_DR) | 305 |
| 14.13.10 | ADC 較正係数 (ADC_CALFACT) | 306 |
| 14.13.11 | ADC 共通設定レジスタ (ADC_CCR) | 307 |
| 14.13.12 | ADC レジスタマップ | 309 |
| 15 | D/A コンバータ (DAC) | 310 |
| 15.1 | 概要 | 310 |
| 15.2 | DAC1 の主な機能 | 310 |
| 15.3 | シングルモードの機能説明 | 312 |
| 15.3.1 | DAC チャンネルイネーブル | 312 |
| 15.3.2 | DAC 出力バッファイネーブル | 312 |
| 15.3.3 | DAC データフォーマット | 312 |
| 15.3.4 | DAC チャンネル変換 | 312 |
| 15.3.5 | DAC 出力電圧 | 313 |
| 15.3.6 | DAC トリガ選択 | 314 |

| | | |
|-----------|--|------------|
| 15.4 | ノイズ生成 | 315 |
| 15.5 | 三角波生成 | 316 |
| 15.6 | DMA リクエスト | 316 |
| 15.7 | DAC レジスタ | 317 |
| 15.7.1 | DAC 制御レジスタ (DAC_CR) | 317 |
| 15.7.2 | DAC ソフトウェアトリガレジスタ (DAC_SWTRIGR) | 320 |
| 15.7.3 | DAC チャンネル 1 の 12 ビット右詰めデータ保持レジスタ (DAC_DHR12R1) | 320 |
| 15.7.4 | DAC チャンネル 1 の 12 ビット左詰めデータ保持レジスタ (DAC_DHR12L1) | 321 |
| 15.7.5 | DAC チャンネル 1 の 8 ビット右詰めデータ保持レジスタ (DAC_DHR8R1) | 321 |
| 15.7.6 | DAC チャンネル 1 データ出力レジスタ (DAC_DOR1) | 321 |
| 15.7.7 | DAC ステータスレジスタ (DAC_SR) | 322 |
| 15.7.8 | DAC レジスタマップ | 323 |
| 16 | コンパレータ (COMP) | 324 |
| 16.1 | 概要 | 324 |
| 16.2 | COMP の主な機能 | 324 |
| 16.3 | COMP の機能説明 | 325 |
| 16.3.1 | COMP ブロック図 | 325 |
| 16.3.2 | COMP ピンおよび内部信号 | 325 |
| 16.3.3 | COMP のリセットおよびクロック | 326 |
| 16.3.4 | コンパレータのロック機構 | 326 |
| 16.4 | 電力モード | 326 |
| 16.5 | 割り込み | 326 |
| 16.6 | COMP レジスタ | 327 |
| 16.6.1 | コンパレータ 1 制御/ステータスレジスタ (COMP1_CSR) | 327 |
| 16.6.2 | コンパレータ 2 制御/ステータスレジスタ (COMP2_CSR) | 328 |
| 16.6.3 | COMP レジスタマップ | 330 |
| 17 | 液晶ディスプレイコントローラ (LCD) | 331 |
| 17.1 | 概要 | 331 |
| 17.2 | LCD の主な機能 | 332 |
| 17.3 | LCD の機能詳細 | 333 |
| 17.3.1 | 概要 | 333 |
| 17.3.2 | 周波数発生器 | 334 |

| | | |
|-----------|-------------------------------------|------------|
| 17.3.3 | コモンドライバ | 335 |
| 17.3.4 | セグメントドライバ | 338 |
| 17.3.5 | 電圧発生器とコントラスト制御 | 342 |
| 17.3.6 | ダブルバッファメモリ | 346 |
| 17.3.7 | COM および SEG の多重化 | 346 |
| 17.3.8 | フローチャート | 347 |
| 17.4 | LCD 低電力モード | 348 |
| 17.5 | LCD 割り込み | 349 |
| 17.6 | LCD レジスタ | 350 |
| 17.6.1 | LCD 制御レジスタ (LCD_CR) | 350 |
| 17.6.2 | LCD フレーム制御レジスタ (LCD_FCR) | 351 |
| 17.6.3 | LCD ステータスレジスタ (LCD_SR) | 354 |
| 17.6.4 | LCD クリアレジスタ (LCD_CLR) | 356 |
| 17.6.5 | LCD ディスプレイメモリ (LCD_RAM) | 357 |
| 17.6.6 | LCD レジスタマップ | 358 |
| 18 | タッチセンシングコントローラ (TSC) | 360 |
| 18.1 | 概要 | 360 |
| 18.2 | TSC の主な機能 | 360 |
| 18.3 | TSC の機能説明 | 361 |
| 18.3.1 | TSC ブロック図 | 361 |
| 18.3.2 | 表面電荷移動取得の概要 | 361 |
| 18.3.3 | リセットおよびクロック | 363 |
| 18.3.4 | 電荷移動取得シーケンス | 364 |
| 18.3.5 | スペクトル拡散機能 | 365 |
| 18.3.6 | 最大カウントエラー | 365 |
| 18.3.7 | サンプリングコンデンサ I/O および チャネル I/O のモード選択 | 366 |
| 18.3.8 | 取得モード | 367 |
| 18.3.9 | I/O ヒステリシスおよびアナログスイッチの制御 | 367 |
| 18.3.10 | 静電容量検出 GPIO | 368 |
| 18.4 | TSC 低電力モード | 368 |
| 18.5 | TSC 割り込み | 369 |
| 18.6 | TSC レジスタ | 369 |
| 18.6.1 | TSC 制御レジスタ (TSC_CR) | 369 |
| 18.6.2 | TSC 割り込み有効レジスタ (TSC_IER) | 371 |
| 18.6.3 | TSC 割り込みクリアレジスタ (TSC_ICR) | 372 |

| | | |
|-----------|---|------------|
| 18.6.4 | TSC 割り込みステータスレジスタ (TSC_ISR) | 373 |
| 18.6.5 | TSC I/O ヒステリシス制御レジスタ (TSC_IQHCR) | 374 |
| 18.6.6 | TSC I/O アナログスイッチ制御レジスタ (TSC_IOASCR) | 374 |
| 18.6.7 | TSC I/O サンプリング制御レジスタ (TSC_IOSCR) | 375 |
| 18.6.8 | TSC I/O チャンネル制御レジスタ (TSC_IOCRR) | 375 |
| 18.6.9 | TSC I/O グループ制御ステータスレジスタ (TSC_IQCSR) | 376 |
| 18.6.10 | TSC I/O グループ x カウンタレジスタ (TSC_IQxCR) (x = 1 ~ 8) | 376 |
| 18.6.11 | TSC レジスタマップ | 377 |
| 19 | AES ハードウェアアクセラレータ (AES) | 379 |
| 19.1 | 概要 | 379 |
| 19.2 | AES の主な機能 | 379 |
| 19.3 | AES の機能詳細 | 380 |
| 19.4 | 暗号化キーと復号化キー | 381 |
| 19.5 | AES 連鎖アルゴリズム | 382 |
| 19.5.1 | ECB (電子コードブック) | 382 |
| 19.5.2 | CBC (暗号ブロック連鎖) | 383 |
| 19.5.3 | CTR (カウンタモード) | 386 |
| 19.6 | データ型 | 387 |
| 19.7 | 動作モード | 390 |
| 19.7.1 | モード 1 : 暗号化 | 390 |
| 19.7.2 | モード 2 : キー派生 | 391 |
| 19.7.3 | モード 3 : 復号化 | 391 |
| 19.7.4 | モード 4 : キー派生と復号化 | 392 |
| 19.8 | AES DMA インタフェース | 393 |
| 19.9 | エラーフラグ | 394 |
| 19.10 | 処理時間 | 394 |
| 19.11 | AES 割り込み | 395 |
| 19.12 | AES レジスタ | 396 |
| 19.12.1 | AES 制御レジスタ (AES_CR) | 396 |
| 19.12.2 | AES ステータスレジスタ (AES_SR) | 398 |
| 19.12.3 | AES データ入力レジスタ (AES_DINR) | 399 |
| 19.12.4 | AES データ出力レジスタ (AES_DOUTR) | 399 |
| 19.12.5 | AES キーレジスタ 0 (AES_KEYR0) (LSB: キー [31:0]) | 400 |
| 19.12.6 | AES キーレジスタ 1 (AES_KEYR1) (キー [63:32]) | 400 |
| 19.12.7 | AES キーレジスタ 2 (AES_KEYR2) (キー [95:64]) | 401 |

| | | |
|-----------|--|------------|
| 19.12.8 | AES キーレジスタ 3 (AES_KEYR3) (MSB : キー [127:96]) | 401 |
| 19.12.9 | AES 初期化ベクトルレジスタ 0 (AES_IVR0) (LSB : IVR[31:0]) | 401 |
| 19.12.10 | AES 初期化ベクトルレジスタ 1 (AES_IVR1) (IVR[63:32]) | 402 |
| 19.12.11 | AES 初期化ベクトルレジスタ 2 (AES_IVR2) (IVR[95:64]) | 403 |
| 19.12.12 | AES 初期化ベクトルレジスタ 3 (AES_IVR3) (MSB : IVR[127:96]) | 403 |
| 19.12.13 | AES レジスタマップ | 404 |
| 20 | 乱数発生器 (RNG) | 405 |
| 20.1 | 概要 | 405 |
| 20.2 | RNG の主な機能 | 405 |
| 20.3 | RNG の機能説明 | 405 |
| 20.3.1 | 動作 | 406 |
| 20.3.2 | エラー管理 | 406 |
| 20.4 | RNG レジスタ | 406 |
| 20.4.1 | RNG 制御レジスタ (RNG_CR) | 407 |
| 20.4.2 | RNG ステータスレジスタ (RNG_SR) | 407 |
| 20.4.3 | RNG データレジスタ (RNG_DR) | 408 |
| 20.4.4 | RNG レジスタマップ | 409 |
| 21 | 汎用タイマ (TIM2) | 410 |
| 21.1 | TIM2 の概要 | 410 |
| 21.2 | TIM2 の主な機能 | 410 |
| 21.3 | TIM2 の機能詳細 | 411 |
| 21.3.1 | タイムベースユニット | 411 |
| 21.3.2 | カウンタモード | 413 |
| 21.3.3 | クロック選択 | 424 |
| 21.3.4 | キャプチャ/比較チャネル | 427 |
| 21.3.5 | 入力キャプチャモード | 429 |
| 21.3.6 | PWM 入力モード | 430 |
| 21.3.7 | 強制出力モード | 431 |
| 21.3.8 | 出力比較モード | 432 |
| 21.3.9 | PWM モード | 433 |
| 21.3.10 | ワンパルスモード | 436 |
| 21.3.11 | 外部イベントによる OCxREF 信号のクリア | 437 |
| 21.3.12 | エンコーダインタフェースモード | 438 |
| 21.3.13 | タイマ入力 XOR 機能 | 440 |

| | | |
|-----------|-------------------------------------|------------|
| 21.3.14 | タイマと外部トリガの同期 | 441 |
| 21.3.15 | タイマの同期 | 444 |
| 21.3.16 | デバッグモード | 449 |
| 21.4 | TIM2 レジスタ | 450 |
| 21.4.1 | TIMx 制御レジスタ 1 (TIMx_CR1) | 450 |
| 21.4.2 | TIMx 制御レジスタ 2 (TIMx_CR2) | 452 |
| 21.4.3 | TIMx スレーブモード制御レジスタ (TIMx_SMCR) | 453 |
| 21.4.4 | TIMx DMA / 割り込み有効レジスタ (TIMx_DIER) | 455 |
| 21.4.5 | TIMx ステータスレジスタ (TIMx_SR) | 456 |
| 21.4.6 | TIMx イベント生成レジスタ (TIMx_EGR) | 458 |
| 21.4.7 | TIMx キャプチャ/比較モードレジスタ 1 (TIMx_CCMR1) | 459 |
| 21.4.8 | TIMx キャプチャ/比較モードレジスタ 2 (TIMx_CCMR2) | 462 |
| 21.4.9 | TIMx キャプチャ/比較有効レジスタ (TIMx_CCER) | 463 |
| 21.4.10 | TIMx カウンタ (TIMx_CNT) | 465 |
| 21.4.11 | TIMx プリスケアラ (TIMx_PSC) | 465 |
| 21.4.12 | TIMx 自動再ロードレジスタ (TIMx_ARR) | 465 |
| 21.4.13 | TIMx キャプチャ/比較モードレジスタ 1 (TIMx_CCR1) | 466 |
| 21.4.14 | TIMx キャプチャ/比較モードレジスタ 2 (TIMx_CCR2) | 466 |
| 21.4.15 | TIMx キャプチャ/比較モードレジスタ 3 (TIMx_CCR3) | 467 |
| 21.4.16 | TIMx キャプチャ/比較モードレジスタ 4 (TIMx_CCR4) | 467 |
| 21.4.17 | TIMx DMA 制御レジスタ (TIMx_DCR) | 468 |
| 21.4.18 | 完全転送の TIMx DMA アドレス (TIMx_DMAR) | 468 |
| 21.4.19 | TIM2 オプションレジスタ (TIM2_OR) | 470 |
| 21.4.20 | TIMx レジスタマップ | 470 |
| 22 | 汎用タイマ (TIM21/22) | 473 |
| 22.1 | 概要 | 473 |
| 22.2 | TIM21/22 の主な機能 | 473 |
| 22.2.1 | TIM21/22 の主な機能 | 473 |
| 22.3 | TIM21/22 の機能説明 | 475 |
| 22.3.1 | タイムベースユニット | 475 |
| 22.3.2 | カウントモード | 477 |
| 22.3.3 | クロック選択 | 487 |
| 22.3.4 | キャプチャ/比較チャネル | 490 |
| 22.3.5 | 入力キャプチャモード | 492 |
| 22.3.6 | PWM 入力モード | 493 |
| 22.3.7 | 強制出力モード | 494 |

| | | |
|---------|--|-----|
| 22.3.8 | 出力比較モード | 494 |
| 22.3.9 | PWM モード | 496 |
| 22.3.10 | 外部イベントによる OCxREF 信号のクリア | 499 |
| 22.3.11 | ワンパルスモード | 500 |
| 22.3.12 | エンコーダインタフェースモード | 501 |
| 22.3.13 | TIM21/22 外部 トリガ同期 | 504 |
| 22.3.14 | タイマ同期 (TIM21/22) | 507 |
| 22.3.15 | デバッグモード | 507 |
| 22.4 | TIM21/22 レジスタ | 508 |
| 22.4.1 | TIM21/22 制御レジスタ 1 (TIMx_CR1) | 508 |
| 22.4.2 | TIM21/22 制御レジスタ 2 (TIMx_CR2) | 510 |
| 22.4.3 | TIM21/22 のスレーブモード制御レジスタ (TIMx_SMCR) | 511 |
| 22.4.4 | TIM21/22 割り込み有効レジスタ (TIMx_DIER) | 514 |
| 22.4.5 | TIM21/22 のステータスレジスタ (TIMx_SR) | 514 |
| 22.4.6 | TIM21/22 のイベント生成レジスタ (TIMx_EGR) | 516 |
| 22.4.7 | TIM21/22 のキャプチャ/比較モードレジスタ 1 (TIMx_CCMR1) | 517 |
| 22.4.8 | TIM21/22 のキャプチャ/比較有効レジスタ (TIMx_CCER) | 520 |
| 22.4.9 | TIM21/22 のカウンタ (TIMx_CNT) | 521 |
| 22.4.10 | TIM21/22 のプリスケアラ (TIMx_PSC) | 521 |
| 22.4.11 | TIM21/22 の自動再ロードレジスタ (TIMx_ARR) | 521 |
| 22.4.12 | TIM21/22 のキャプチャ/比較レジスタ 1 (TIMx_CCR1) | 522 |
| 22.4.13 | TIM21/22 のキャプチャ/比較レジスタ 2 (TIMx_CCR2) | 522 |
| 22.4.14 | TIM21 オプションレジスタ (TIM21_OR) | 523 |
| 22.4.15 | TIM22 オプションレジスタ (TIM22_OR) | 524 |
| 22.4.16 | TIM21/22 のレジスタマップ | 524 |
| 23 | 基本タイマ (TIM6) | 527 |
| 23.1 | 概要 | 527 |
| 23.2 | TIM6 の主な機能 | 527 |
| 23.3 | TIM6 の機能説明 | 528 |
| 23.3.1 | タイムベースユニット | 528 |
| 23.3.2 | カウントモード | 530 |
| 23.3.3 | クロックソース | 533 |
| 23.3.4 | デバッグモード | 534 |
| 23.4 | TIM6 レジスタ | 534 |
| 23.4.1 | TIM6 制御レジスタ 1 (TIMx_CR1) | 535 |
| 23.4.2 | TIM6 制御レジスタ 2 (TIMx_CR2) | 536 |

| | | |
|-----------|---|------------|
| 23.4.3 | TIM6 DMA / 割り込み有効レジスタ (TIMx_DIER) | 536 |
| 23.4.4 | TIM6 のステータスレジスタ (TIMx_SR) | 537 |
| 23.4.5 | TIM6 のイベント生成レジスタ (TIMx_EGR) | 537 |
| 23.4.6 | TIM6 のカウンタ (TIMx_CNT) | 537 |
| 23.4.7 | TIM6 のプリスケアラ (TIMx_PSC) | 538 |
| 23.4.8 | TIM6 の自動再ロードレジスタ (TIMx_ARR) | 538 |
| 23.4.9 | TIM6 レジスタマップ | 539 |
| 24 | 低電力タイマ (LPTIM) | 540 |
| 24.1 | 概要 | 540 |
| 24.2 | LPTIM の主な機能 | 540 |
| 24.3 | LPTIM の実装 | 540 |
| 24.4 | LPTIM の機能詳細 | 541 |
| 24.4.1 | LPTIM ブロック図 | 541 |
| 24.4.2 | LPTIM のリセットとクロック | 541 |
| 24.4.3 | グリッチフィルタ | 542 |
| 24.4.4 | プリスケアラ | 543 |
| 24.4.5 | トリガマルチプレクサ | 543 |
| 24.4.6 | 動作モード | 543 |
| 24.4.7 | タイムアウト機能 | 544 |
| 24.4.8 | 波形生成 | 544 |
| 24.4.9 | レジスタの更新 | 545 |
| 24.4.10 | カウンタモード | 546 |
| 24.4.11 | タイマ有効 | 546 |
| 24.4.12 | エンコーダモード | 546 |
| 24.5 | LPTIM 割り込み | 548 |
| 24.6 | LPTIM レジスタ | 549 |
| 24.6.1 | LPTIM 割り込みおよびステータスレジスタ (LPTIMx_ISR) | 549 |
| 24.6.2 | LPTIM 割り込みクリアレジスタ (LPTIMx_ICR) | 550 |
| 24.6.3 | LPTIM 割り込み有効レジスタ (LPTIMx_IER) | 551 |
| 24.6.4 | LPTIM 構成レジスタ (LPTIMx_CFGR) | 552 |
| 24.6.5 | LPTIM 制御レジスタ (LPTIMx_CR) | 555 |
| 24.6.6 | LPTIM 比較レジスタ (LPTIMx_CMP) | 556 |
| 24.6.7 | LPTIM 自動再ロードレジスタ (LPTIMx_ARR) | 556 |
| 24.6.8 | LPTIM カウンタレジスタ (LPTIMx_CNT) | 557 |
| 24.6.9 | LPTIM レジスタマップ | 558 |

| | | |
|-----------|---------------------------------------|------------|
| 25 | 独立型ウォッチドッグ (IWDG) | 559 |
| 25.1 | 概要 | 559 |
| 25.2 | IWDG の主な機能 | 559 |
| 25.3 | IWDG の機能説明 | 559 |
| 25.3.1 | IWDG ブロック図 | 559 |
| 25.3.2 | ウィンドウオプション | 560 |
| 25.3.3 | ハードウェアウォッチドッグ | 561 |
| 25.3.4 | レジスタのアクセス保護 | 561 |
| 25.3.5 | デバッグモード | 561 |
| 25.4 | IWDG レジスタ | 562 |
| 25.4.1 | キーレジスタ (IWDG_KR) | 562 |
| 25.4.2 | プリスケアラレジスタ (IWDG_PR) | 563 |
| 25.4.3 | 再ロードレジスタ (IWDG_RLR) | 564 |
| 25.4.4 | ステータスレジスタ (IWDG_SR) | 565 |
| 25.4.5 | ウィンドウレジスタ (IWDG_WINR) | 566 |
| 25.4.6 | IWDG レジスタマップ | 567 |
| 26 | システムウィンドウ型ウォッチドッグ (WWDG) | 568 |
| 26.1 | 概要 | 568 |
| 26.2 | WWDG の主な機能 | 568 |
| 26.3 | WWDG の機能説明 | 568 |
| 26.3.1 | ウォッチドッグの有効化 | 569 |
| 26.3.2 | ダウンカウンタの制御 | 569 |
| 26.3.3 | 高度なウォッチドッグ割り込み機能 | 569 |
| 26.3.4 | ウォッチドッグタイムアウトをプログラムする方法 | 570 |
| 26.3.5 | デバッグモード | 571 |
| 26.4 | WWDG レジスタ | 571 |
| 26.4.1 | 制御レジスタ (WWDG_CR) | 571 |
| 26.4.2 | 設定レジスタ (WWDG_CFR) | 572 |
| 26.4.3 | ステータスレジスタ (WWDG_SR) | 572 |
| 26.4.4 | WWDG レジスタマップ | 573 |

| | | |
|-----------|-------------------------------------|------------|
| 27 | リアルタイムクロック (RTC) | 574 |
| 27.1 | 概要 | 574 |
| 27.2 | RTC の主な機能 | 574 |
| 27.3 | RTC の機能説明 | 575 |
| 27.3.1 | RTC ブロック図 | 575 |
| 27.3.2 | RTC によって制御される GPIO | 576 |
| 27.3.3 | クロックとプリスケアラ | 577 |
| 27.3.4 | リアルタイムクロックとカレンダー | 578 |
| 27.3.5 | プログラム可能なアラーム | 579 |
| 27.3.6 | 周期的自動ウェイクアップ | 579 |
| 27.3.7 | RTC の初期化と設定 | 580 |
| 27.3.8 | カレンダーの読み出し | 581 |
| 27.3.9 | RTC のリセット | 582 |
| 27.3.10 | RTC の同期 | 583 |
| 27.3.11 | RTC リファレンスクロック検出 | 583 |
| 27.3.12 | RTC の高精度デジタル較正 | 584 |
| 27.3.13 | タイムスタンプ機能 | 586 |
| 27.3.14 | タンパ検出 | 586 |
| 27.3.15 | 較正クロック出力 | 589 |
| 27.3.16 | アラーム出力 | 589 |
| 27.4 | RTC 低電力モード | 589 |
| 27.5 | RTC 割り込み | 590 |
| 27.6 | RTC レジスタ | 591 |
| 27.6.1 | RTC 時刻レジスタ (RTC_TR) | 591 |
| 27.6.2 | RTC 日付レジスタ (RTC_DR) | 592 |
| 27.6.3 | RTC 制御レジスタ (RTC_CR) | 593 |
| 27.6.4 | RTC 初期化とステータスレジスタ (RTC_ISR) | 596 |
| 27.6.5 | RTC プリスケアラレジスタ (RTC_PRER) | 599 |
| 27.6.6 | RTC ウェイクアップタイムレジスタ (RTC_WUTR) | 600 |
| 27.6.7 | RTC アラーム A レジスタ (RTC_ALRMAR) | 601 |
| 27.6.8 | RTC アラーム B レジスタ (RTC_ALRMBR) | 602 |
| 27.6.9 | RTC 書き込み保護レジスタ (RTC_WPR) | 603 |
| 27.6.10 | RTC サブセカンドレジスタ (RTC_SSR) | 604 |
| 27.6.11 | RTC シフト制御レジスタ (RTC_SHIFTR) | 605 |
| 27.6.12 | RTC タイムスタンプ時刻レジスタ (RTC_TSTR) | 606 |
| 27.6.13 | RTC タイムスタンプ日付レジスタ (RTC_TSDR) | 607 |

| | | |
|-----------|---|------------|
| 27.6.14 | RTC タイムスタンプサブセカンドレジスタ (RTC_TSSSR) | 608 |
| 27.6.15 | RTC 較正レジスタ (RTC_CALR) | 609 |
| 27.6.16 | RTC タンパ設定レジスタ (RTC_TAMPCR) | 610 |
| 27.6.17 | RTC アラーム A サブセカンドレジスタ (RTC_ALRMSSR) | 613 |
| 27.6.18 | RTC アラーム B サブセカンドレジスタ (RTC_ALRMBSSR) | 614 |
| 27.6.19 | RTC オプションレジスタ (RTC_OR) | 615 |
| 27.6.20 | RTC バックアップレジスタ (RTC_BKPxR) | 616 |
| 27.6.21 | RTC レジスタマップ | 617 |
| 28 | I2C (Inter-integrated circuit) インタフェース | 619 |
| 28.1 | 概要 | 619 |
| 28.2 | I2C の主な機能 | 619 |
| 28.3 | I2C の実装 | 620 |
| 28.4 | I2C 機能の詳細 | 620 |
| 28.4.1 | I2C1 ブロック図 | 621 |
| 28.4.2 | I2C2 ブロック図 | 622 |
| 28.4.3 | I2C クロックの要件 | 622 |
| 28.4.4 | モード選択 | 623 |
| 28.4.5 | I2C の初期化 | 624 |
| 28.4.6 | ソフトウェアリセット | 628 |
| 28.4.7 | データ転送 | 629 |
| 28.4.8 | I2C スレーブモード | 631 |
| 28.4.9 | I2C マスタモード | 640 |
| 28.4.10 | I2Cx_TIMINGR レジスタの設定例 | 652 |
| 28.4.11 | SMBus 固有の機能 | 653 |
| 28.4.12 | SMBus 初期化 | 656 |
| 28.4.13 | SMBus : I2Cx_TIMEOCTR レジスタの設定例 | 658 |
| 28.4.14 | SMBus スレーブモード | 658 |
| 28.4.15 | アドレス一致時に STOP モードからウェイクアップ | 665 |
| 28.4.16 | エラー条件 | 666 |
| 28.4.17 | DMA リクエスト | 668 |
| 28.4.18 | デバッグモード | 669 |
| 28.5 | I2C 低電力モード | 669 |
| 28.6 | I2C 割り込み | 669 |
| 28.7 | I2C レジスタ | 670 |
| 28.7.1 | 制御レジスタ 1 (I2Cx_CR1) | 671 |

| | | |
|---------|--------------------------------------|-----|
| 28.7.2 | 制御レジスタ 2 (I2Cx_CR2) | 674 |
| 28.7.3 | Own Address 1 レジスタ (I2Cx_OAR1) | 677 |
| 28.7.4 | Own Address 2 レジスタ (I2Cx_OAR2) | 678 |
| 28.7.5 | タイミングレジスタ (I2Cx_TIMINGR) | 679 |
| 28.7.6 | タイムアウトレジスタ (I2Cx_TIMEOUTR) | 680 |
| 28.7.7 | 割り込みおよびステータスレジスタ (I2Cx_ISR) | 681 |
| 28.7.8 | 割り込みクリアレジスタ (I2Cx_ICR) | 683 |
| 28.7.9 | PEC レジスタ (I2Cx_PECR) | 684 |
| 28.7.10 | 受信データレジスタ (I2Cx_RXDR) | 685 |
| 28.7.11 | 送信データレジスタ (I2Cx_TXDR) | 685 |
| 28.7.12 | I2C レジスタマップ | 686 |

| | | |
|-----------|--|------------|
| 29 | USART | |
| | (Universal synchronous asynchronous receiver transmitter) | 688 |
| 29.1 | 概要 | 688 |
| 29.2 | USART の主な機能 | 688 |
| 29.3 | USART の拡張機能 | 689 |
| 29.4 | USART の実装 | 690 |
| 29.5 | USART の機能詳細 | 690 |
| 29.5.1 | USART キャラクタの説明 | 693 |
| 29.5.2 | トランスミッタ | 694 |
| 29.5.3 | レシーバ | 697 |
| 29.5.4 | ボーレート生成 | 704 |
| 29.5.5 | クロック偏差に対する USART レシーバの許容誤差 | 706 |
| 29.5.6 | 自動ボーレート検出 | 707 |
| 29.5.7 | マルチプロセッサ通信 | 708 |
| 29.5.8 | Modbus 通信 | 710 |
| 29.5.9 | パリティ制御 | 710 |
| 29.5.10 | LIN (Local Interconnection Network) モード | 711 |
| 29.5.11 | USART 同期モード | 714 |
| 29.5.12 | 単線半二重通信 | 716 |
| 29.5.13 | スマートカードモード | 717 |
| 29.5.14 | IrDA SIR ENDEC ブロック | 721 |
| 29.5.15 | DMA を使用した連続通信 | 723 |
| 29.5.16 | RS232 ハードウェアフロー制御および RS485 ドライバ有効 | 726 |
| 29.5.17 | STOP モードからのウェイクアップ | 728 |

| | | |
|-----------|---|------------|
| 29.6 | USART 割り込み | 728 |
| 29.7 | USART レジスタ | 730 |
| 29.7.1 | 制御レジスタ 1 (USARTx_CR1) | 730 |
| 29.7.2 | 制御レジスタ 2 (USARTx_CR2) | 733 |
| 29.7.3 | 制御レジスタ 3 (USARTx_CR3) | 737 |
| 29.7.4 | ボーレートレジスタ (USARTx_BRR) | 742 |
| 29.7.5 | ガード時間およびプリスケアラレジスタ (USARTx_GTPR) | 743 |
| 29.7.6 | レシーバタイムアウトレジスタ (USARTx_RTOR) | 744 |
| 29.7.7 | リクエストレジスタ (USARTx_RQR) | 745 |
| 29.7.8 | 割り込みおよびステータスレジスタ (USARTx_ISR) | 746 |
| 29.7.9 | 割り込みフラグクリアレジスタ (USARTx_ICR) | 751 |
| 29.7.10 | 受信データレジスタ (USARTx_RDR) | 752 |
| 29.7.11 | 送信データレジスタ (USARTx_TDR) | 753 |
| 29.7.12 | USART レジスタマップ | 754 |
| 30 | 低電力ユニバーサル非同期レシーバトランスミッタ (LPUART) | 755 |
| 30.1 | 概要 | 755 |
| 30.2 | LPUART の主な機能 | 756 |
| 30.3 | LPUART の実装 | 756 |
| 30.4 | LPUART 機能の詳細 | 757 |
| 30.4.1 | LPUART キャラクタの説明 | 759 |
| 30.4.2 | トランスミッタ | 761 |
| 30.4.3 | レシーバ | 763 |
| 30.4.4 | ボーレート生成 | 766 |
| 30.4.5 | マルチプロセッサ通信 | 767 |
| 30.4.6 | パリティ制御 | 769 |
| 30.4.7 | 単線半二重通信 | 770 |
| 30.4.8 | DMA を使用した連続通信 | 770 |
| 30.4.9 | RS232 ハードウェアフロー制御および RS485 ドライバ有効 | 773 |
| 30.4.10 | STOP モードからのウェイクアップ | 775 |
| 30.5 | LPUART 割り込み | 775 |
| 30.6 | LPUART レジスタ | 777 |
| 30.6.1 | 制御レジスタ 1 (LPUARTx_CR1) | 777 |
| 30.6.2 | 制御レジスタ 2 (LPUARTx_CR2) | 780 |
| 30.6.3 | 制御レジスタ 3 (LPUARTx_CR3) | 782 |
| 30.6.4 | ボーレートレジスタ (LPUARTx_BRR) | 784 |

| | | |
|-----------|---|------------|
| 30.6.5 | リクエストレジスタ (LPUARTx_RQR) | 785 |
| 30.6.6 | 割り込みおよびステータスレジスタ (LPUARTx_ISR) | 785 |
| 30.6.7 | 割り込みフラグクリアレジスタ (LPUARTx_ICR) | 789 |
| 30.6.8 | 受信データレジスタ (LPUARTx_RDR) | 790 |
| 30.6.9 | 送信データレジスタ (LPUARTx_TDR) | 790 |
| 30.6.10 | LPUART レジスタマップ | 791 |
| 31 | シリアルペリフェラルインタフェース / I²S (SPI/I2S) | 792 |
| 31.1 | 概要 | 792 |
| 31.1.1 | SPI の主な機能 | 792 |
| 31.1.2 | SPI の拡張機能 | 793 |
| 31.1.3 | I ² S の機能 | 793 |
| 31.2 | SPI/I2S の実装 | 794 |
| 31.3 | SPI の機能説明 | 795 |
| 31.3.1 | 概要 | 795 |
| 31.3.2 | マスタとスレーブの 1 対 1 の通信 | 796 |
| 31.3.3 | 標準マルチスレーブ通信 | 799 |
| 31.3.4 | スレーブ選択 (NSS) ピンの管理 | 800 |
| 31.3.5 | 通信フォーマット | 801 |
| 31.3.6 | SPI の設定 | 803 |
| 31.3.7 | SPI を有効にする手順 | 803 |
| 31.3.8 | データの送受信手順 | 804 |
| 31.3.9 | SPI を無効にする手順 | 806 |
| 31.3.10 | DMA (ダイレクトメモリアクセス) を使用する通信 | 807 |
| 31.3.11 | SPI ステータスフラグ | 809 |
| 31.3.12 | SPI エラーフラグ | 809 |
| 31.4 | SPI の特殊機能 | 811 |
| 31.4.1 | TI モード | 811 |
| 31.4.2 | CRC 計算 | 812 |
| 31.5 | SPI 割り込み | 813 |
| 31.6 | I ² S の機能説明 | 814 |
| 31.6.1 | I ² S の概要 | 814 |
| 31.6.2 | サポートされるオーディオプロトコル | 815 |
| 31.6.3 | クロックジェネレータ | 821 |
| 31.6.4 | I ² S マスタモード | 824 |
| 31.6.5 | I ² S スレーブモード | 826 |

| | | |
|-----------|---|------------|
| 31.6.6 | I ² S ステータスフラグ | 827 |
| 31.6.7 | I ² S エラーフラグ | 828 |
| 31.6.8 | I ² S 割り込み | 829 |
| 31.6.9 | DMA の機能 | 829 |
| 31.7 | SPI および I ² S レジスタ | 830 |
| 31.7.1 | SPI 制御レジスタ 1 (SPI_CR1) (I ² S モードでは使用しません) | 830 |
| 31.7.2 | SPI 制御レジスタ 2 (SPI_CR2) | 832 |
| 31.7.3 | SPI ステータスレジスタ (SPI_SR) | 833 |
| 31.7.4 | SPI データレジスタ (SPI_DR) | 834 |
| 31.7.5 | SPI CRC 多項式レジスタ (SPI_CRCPR) (I ² S モードでは使用しません) | 835 |
| 31.7.6 | SPI RX CRC レジスタ (SPI_RXCRCR) (I ² S モードでは使用しません) | 835 |
| 31.7.7 | SPI TX CRC レジスタ (SPI_TXCRCR) (I ² S では使用しません) | 836 |
| 31.7.8 | SPI_I2S 設定レジスタ (SPI_I2SCFGR) | 836 |
| 31.7.9 | SPI_I2S プリスケアラレジスタ (SPI_I2SPR) | 838 |
| 31.7.10 | SPI レジスタマップ | 839 |
| 32 | ユニバーサルシリアルバスフルスピードデバイスインタフェース (USB) | 840 |
| 32.1 | 概要 | 840 |
| 32.2 | USB の主な機能 | 840 |
| 32.3 | USB の実装 | 840 |
| 32.4 | USART の機能詳細 | 841 |
| 32.4.1 | USB ブロックの説明 | 842 |
| 32.5 | プログラミングに関する考慮事項 | 843 |
| 32.5.1 | 汎用 USB デバイスのプログラミング | 843 |
| 32.5.2 | システムリセットとパワーオンリセット | 843 |
| 32.5.3 | ダブルバッファエンドポイント | 848 |
| 32.5.4 | アイソクロナス転送 | 851 |
| 32.5.5 | サスペンド/レジャームイベント | 852 |
| 32.6 | USB レジスタ | 854 |
| 32.6.1 | 共通レジスタ | 854 |
| 32.6.2 | エンドポイント固有レジスタ | 863 |
| 32.6.3 | バッファディスクリプタテーブル | 868 |
| 32.6.4 | USB レジスタマップ | 871 |

| | | |
|-----------|--|------------|
| 33 | デバッグサポート (DBG) | 873 |
| 33.1 | 概要 | 873 |
| 33.2 | ARM リファレンス資料 | 874 |
| 33.3 | ピン名とデバッグポートピン | 874 |
| 33.3.1 | SWD ポートピン | 875 |
| 33.3.2 | SW-DP ピンの割り当て | 875 |
| 33.3.3 | SWD ピンでの内部プルアップ／プルダウン | 875 |
| 33.4 | ID コードとロック機構 | 876 |
| 33.4.1 | MCU デバイス ID コード | 876 |
| 33.5 | SWD ポート | 877 |
| 33.5.1 | SWD プロトコルの概要 | 877 |
| 33.5.2 | SWD プロトコルシーケンス | 877 |
| 33.5.3 | SW-DP ステートマシン (リセット、アイドル状態、ID コード) | 878 |
| 33.5.4 | DP と AP の読み出し／書き込みアクセス | 879 |
| 33.5.5 | SW-DP レジスタ | 880 |
| 33.5.6 | SW-AP レジスタ | 881 |
| 33.6 | コアデバッグ | 881 |
| 33.7 | BPU (ブレークポイントユニット) | 882 |
| 33.7.1 | BPU の機能 | 882 |
| 33.8 | DWT (データウォッチポイント) | 882 |
| 33.8.1 | DWT の機能 | 882 |
| 33.8.2 | DWT プログラムカウンタサンプルレジスタ | 882 |
| 33.9 | MCU デバッグコンポーネント (DBG) | 883 |
| 33.9.1 | 低電力モードのデバッグサポート | 883 |
| 33.9.2 | タイマ、ウォッチドッグ、および I ² C のデバッグサポート | 883 |
| 33.9.3 | デバッグ MCU 設定レジスタ (DBG_CR) | 884 |
| 33.9.4 | デバッグ MCU APB1 凍結レジスタ (DBG_APB1_FZ) | 885 |
| 33.9.5 | デバッグ MCU APB2 凍結レジスタ (DBG_APB2_FZ) | 887 |
| 33.10 | DBG レジスタマップ | 888 |
| 34 | デバイス電子署名 | 889 |
| 34.1 | メモリサイズレジスタ | 889 |
| 34.1.1 | フラッシュサイズレジスタ | 889 |
| 34.2 | ユニークデバイス ID レジスタ (96 ビット) | 889 |
| 35 | 改版履歴 | 895 |

表の一覧

| | | |
|-------|---|-----|
| 表 1. | STM32L0x3 のメモリ容量 | 39 |
| 表 2. | STM32L0x3 ペリフェラルレジスタ境界アドレス | 43 |
| 表 3. | ブートモード | 46 |
| 表 4. | NVM 構成 (STM32L0x3 デバイス) | 49 |
| 表 5. | マスタクロック電源範囲と周波数の関係 | 50 |
| 表 6. | メモリアクセス遅延およびウェイトステート数 | 50 |
| 表 7. | 内部バッファ管理 | 52 |
| 表 8. | バッファの設定と投機的読み出し | 55 |
| 表 9. | すべてのメモリインタフェース設定でのドライストーン性能 | 56 |
| 表 10. | NVM 書き込み/消去タイミング | 68 |
| 表 11. | NVM 書き込み/消去処理時間 | 68 |
| 表 12. | 保護レベルと RDP オプションバイトの内容 | 72 |
| 表 13. | FLASH_WRPROT レジスタの保護ビットとフラッシュプログラムメモリの保護アドレスの関係 | 73 |
| 表 14. | メモリアクセスとモード、保護、およびフラッシュプログラムメモリセクタ | 74 |
| 表 15. | フラッシュ割り込みリクエスト | 76 |
| 表 16. | フラッシュインタフェース - レジスタマップとリセット値 | 93 |
| 表 17. | オプションバイトのフォーマット | 94 |
| 表 18. | オプションバイトの構成 | 94 |
| 表 19. | CRC レジスタマップとリセット値 | 101 |
| 表 20. | ファイアウォールの状態に応じたセグメントへのアクセス | 105 |
| 表 21. | セグメントの粒度と領域の範囲 | 106 |
| 表 22. | ファイアウォールレジスタマップとリセット値 | 114 |
| 表 23. | VCORE のレンジにおける性能 | 119 |
| 表 24. | 低電力モードの概要 | 127 |
| 表 25. | Sleep-now | 130 |
| 表 26. | Sleep-on-exit | 130 |
| 表 27. | Sleep-now | 132 |
| 表 28. | Sleep-on-exit | 132 |
| 表 29. | STOP モード | 134 |
| 表 30. | STANDBY モード | 135 |
| 表 31. | PWR - レジスタマップとリセット値 | 142 |
| 表 32. | システムクロックソースの周波数 | 153 |
| 表 33. | RCC レジスタマップとリセット値 | 190 |
| 表 34. | 低電力モードが CRS に与える影響 | 197 |
| 表 35. | 割り込み制御ビット | 197 |
| 表 36. | CRS レジスタマップとリセット値 | 203 |
| 表 37. | ポートビット設定表 | 207 |
| 表 38. | GPIO レジスタマップとリセット値 | 221 |
| 表 39. | VLCD レールの GPIO ピンへの接続 | 226 |
| 表 40. | SYSCFG レジスタマップとリセット値 | 231 |
| 表 41. | プログラム可能なデータ幅およびエンディアン動作 (ビット PINC = MINC = 1 の場合) | 236 |
| 表 42. | DMA 割り込みリクエスト | 237 |
| 表 43. | チャンネルごとの DMA リクエストの概要 | 239 |
| 表 44. | DMA レジスタマップとリセット値 | 248 |
| 表 45. | ベクタテーブル | 250 |
| 表 46. | EXTI のラインの接続 | 257 |
| 表 47. | 拡張割り込み/イベントコントローラのレジスタマップとリセット値 | 263 |
| 表 48. | ADC 内部信号 | 266 |



| | | |
|--------|--------------------------------|-----|
| 表 49. | ADC ピン | 266 |
| 表 50. | トリガから変換開始までの遅延 | 272 |
| 表 51. | トリガ極性の設定 | 277 |
| 表 52. | 外部トリガ | 277 |
| 表 53. | tsAR タイミングは分解能に依存 | 278 |
| 表 54. | アナログウォッチドッグ比較 | 286 |
| 表 55. | アナログウォッチドッグチャネル選択 | 287 |
| 表 56. | 最大出力結果対 N と M。グレイの値は切り詰めを示す | 288 |
| 表 57. | ADC 割り込み | 293 |
| 表 58. | ADC レジスタマップとリセット値 | 309 |
| 表 59. | DAC ピン | 311 |
| 表 60. | 外部トリガ | 314 |
| 表 61. | DAC レジスタマップ とリセット値 | 323 |
| 表 62. | COMP レジスタマップとリセット値 | 330 |
| 表 63. | フレームレートの計算例 | 334 |
| 表 64. | 点滅周波数 | 342 |
| 表 65. | 低電力モードでの LCD の動作 | 348 |
| 表 66. | LCD 割り込みリクエスト | 349 |
| 表 67. | LCD レジスタマップとリセット値 | 358 |
| 表 68. | 取得シーケンスの概要 | 363 |
| 表 69. | スペクトル拡散偏差と AHB クロック周波数 | 365 |
| 表 70. | モードおよび IODEF ビットの値に応じた I/O の状態 | 366 |
| 表 71. | 静電容量検出 GPIO | 368 |
| 表 72. | 低電力モードが TSC に与える影響 | 368 |
| 表 73. | 割り込み制御ビット | 369 |
| 表 74. | TSC レジスタマップとリセット値 | 377 |
| 表 75. | 処理時間 (クロックサイクル数) | 394 |
| 表 76. | AES 割り込みリクエスト | 395 |
| 表 77. | AES レジスタマップ | 404 |
| 表 78. | RNG レジスタマップとリセット値 | 409 |
| 表 79. | カウント方向とエンコーダ信号 | 439 |
| 表 80. | TIM2 内部トリガ接続 | 454 |
| 表 81. | 標準 OCx チャネルの出力制御ビット | 464 |
| 表 82. | TIM2 レジスタマップとリセット値 | 470 |
| 表 83. | カウント方向とエンコーダ信号 | 502 |
| 表 84. | TIMx 内部トリガ接続 | 513 |
| 表 85. | 標準 OCx チャネルの出力制御ビット | 521 |
| 表 86. | TIM21/22 レジスタマップとリセット値 | 524 |
| 表 87. | TIM6 レジスタマップとリセット値 | 539 |
| 表 88. | STM32L0x3 LPTIM 機能 | 540 |
| 表 89. | プリスケアラの分周比 | 543 |
| 表 90. | エンコーダのカウントシナリオ | 547 |
| 表 91. | LPTIM 外部トリガ接続 | 554 |
| 表 92. | LPTIM レジスタマップとリセット値 | 558 |
| 表 93. | IWDG レジスタマップとリセット値 | 567 |
| 表 94. | WWDG レジスタマップとリセット値 | 573 |
| 表 95. | RTC ピン PC13 の設定 | 576 |
| 表 96. | RTC_OUT の配置 | 577 |
| 表 97. | 低電力モードが RTC に与える影響 | 589 |
| 表 98. | 割り込み制御ビット | 590 |
| 表 99. | RTC レジスタマップとリセット値 | 617 |
| 表 100. | STM32L0x3 I2C の機能 | 620 |

| | | |
|--------|---|-----|
| 表 101. | アナログフィルタとデジタルフィルタの比較 | 624 |
| 表 102. | I2C-SMBUS 仕様のデータのセットアップおよびホールド時間 | 627 |
| 表 103. | I2C 設定表 | 631 |
| 表 104. | I2C-SMBUS 仕様のクロックタイミング | 641 |
| 表 105. | fI2CCLK = 8 MHz でのタイミング設定の例 | 652 |
| 表 106. | fI2CCLK = 16 MHz でのタイミング設定の例 | 652 |
| 表 107. | SMBus タイムアウト仕様 | 655 |
| 表 108. | SMBUS の PEC 設定表 | 656 |
| 表 109. | さまざまな I2CCLK 周波数での TIMEOUTA の設定例 (最大値 $t_{\text{TIMEOUT}} = 25 \text{ ms}$) | 658 |
| 表 110. | さまざまな I2CCLK 周波数での TIMEOUTB の設定例 | 658 |
| 表 111. | さまざまな I2CCLK 周波数での TIMEOUTA の設定例 (最大値 $t_{\text{IDLE}} = 50 \text{ ms}$) | 658 |
| 表 112. | 低電力モード | 669 |
| 表 113. | I2C 割り込みリクエスト | 669 |
| 表 114. | I2C レジスタマップとリセット値 | 686 |
| 表 115. | STM32L0x3 USART の機能 | 690 |
| 表 116. | サンプリングされたデータからのノイズ検出 | 702 |
| 表 117. | 16 倍または 8 倍のオーバーサンプリングの両方の場合の $f_{\text{CK}} = 32 \text{ MHz}$ でのプログラムされた ボーレートの誤差計算 | 705 |
| 表 118. | BRR [3:0] = 0000 のときの USART レシーバの許容誤差 | 706 |
| 表 119. | BRR [3:0] が 0000 でないときの USART レシーバの許容誤差 | 706 |
| 表 120. | フレームフォーマット | 710 |
| 表 121. | USART 割り込みリクエスト | 728 |
| 表 122. | USART レジスタマップとリセット値 | 754 |
| 表 123. | $f_{\text{ck}} = 32,768 \text{ KHz}$ でプログラムされたボーレートのエラー計算 | 767 |
| 表 124. | フレームフォーマット | 769 |
| 表 125. | LPUART 割り込みリクエスト | 775 |
| 表 126. | LPUART レジスタマップとリセット値 | 791 |
| 表 127. | STM32L0x3 SPI の実装 | 794 |
| 表 128. | SPI 割り込みリクエスト | 813 |
| 表 129. | 標準 8 MHz HSE を使用した場合のオーディオ周波数精度 | 823 |
| 表 130. | I ² S 割り込みリクエスト | 829 |
| 表 131. | SPI レジスタマップとリセット値 | 839 |
| 表 132. | STM32L0x3 USB の実装 | 840 |
| 表 133. | ダブルバッファリングバッファフラグの定義 | 849 |
| 表 134. | バルクダブルバッファリングメモリバッファの用途 | 850 |
| 表 135. | アイソクロナスメモリバッファの使用 | 851 |
| 表 136. | レジャーイベント検出 | 853 |
| 表 137. | 受信ステータスエンコード | 866 |
| 表 138. | エンドポイントタイプエンコード | 866 |
| 表 139. | エンドポイントの種類の意味 | 866 |
| 表 140. | 送信ステータスエンコード | 867 |
| 表 141. | 割り当てられるバッファメモリの定義 | 871 |
| 表 142. | USB レジスタマップとリセット値 | 871 |
| 表 143. | SW デバッグポートピン | 875 |
| 表 144. | パケットリクエスト (8 ビット) | 877 |
| 表 145. | ACK 応答 (3 ビット) | 878 |
| 表 146. | データ転送 (33 ビット) | 878 |
| 表 147. | SW-DP レジスタ | 880 |
| 表 148. | シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ | 881 |
| 表 149. | コアデバッグレジスタ | 881 |
| 表 150. | DBG レジスタマップとリセット値 | 888 |
| 表 151. | 文書改版履歴 | 895 |

| | | |
|--------|--------------|-----|
| 表 152. | 文書改版履歴（日本語版） | 897 |
|--------|--------------|-----|

図の一覧

| | | |
|-------|---------------------------------------|-----|
| 図 1. | システムアーキテクチャ | 40 |
| 図 2. | メモリマップ | 42 |
| 図 3. | 1つの内部バッファの構造 | 51 |
| 図 4. | 命令をフェッチして実行するタイミング（プリフェッチ無効） | 54 |
| 図 5. | 命令をフェッチして実行するタイミング（プリフェッチ有効） | 55 |
| 図 6. | RDP レベル | 72 |
| 図 7. | CRC 計算ユニットのブロック図 | 97 |
| 図 8. | STM32L0x3 ファイアウォール接続図 | 103 |
| 図 9. | ファイアウォールの機能状態 | 107 |
| 図 10. | 電源の概要 | 116 |
| 図 11. | VDD と VCORE のレンジにおける性能 | 119 |
| 図 12. | 電源供給スーパバイザ | 123 |
| 図 13. | パワーオンリセット／パワーダウンリセット波形 | 124 |
| 図 14. | BOR の閾値 | 125 |
| 図 15. | PVD の閾値 | 126 |
| 図 16. | 簡略化されたリセット回路図 | 144 |
| 図 17. | クロックツリー | 147 |
| 図 18. | HSE/LSE クロックソース | 148 |
| 図 19. | 周波数を測定するための TIM21 チャンネル 1 の入力キャプチャの使用 | 155 |
| 図 20. | CRS ブロック図 | 194 |
| 図 21. | CRS カウンタの動作 | 195 |
| 図 22. | I/O ポートビットの基本構造 | 206 |
| 図 23. | 5 V トレラント I/O ポートビットの基本構造 | 206 |
| 図 24. | 入力フローティング／プルアップ／プルダウン設定 | 210 |
| 図 25. | 出力設定 | 211 |
| 図 26. | オルタネート機能設定 | 212 |
| 図 27. | 高インピーダンスアナログ設定 | 213 |
| 図 28. | DMA ブロック図 | 233 |
| 図 29. | DMA リクエストマッピング | 238 |
| 図 30. | 拡張割り込み／イベントコントローラ（EXTI）のブロック図 | 254 |
| 図 31. | 拡張割り込み／イベント GPIO の配置 | 256 |
| 図 32. | ADC ブロック図 | 267 |
| 図 33. | ADC 較正 | 269 |
| 図 34. | 較正係数によって強制される | 270 |
| 図 35. | ADC の有効化 / 無効化 | 271 |
| 図 36. | ADC クロック構成 | 271 |
| 図 37. | アナログ / デジタル変換時間 | 275 |
| 図 38. | ADC 変換タイミング | 276 |
| 図 39. | 実行中の変換の停止 | 276 |
| 図 40. | シーケンスのシングル変換、ソフトウェアトリガ | 279 |
| 図 41. | シーケンスの連続変換、ソフトウェアトリガ | 280 |
| 図 42. | シーケンスのシングル変換、ハードウェアトリガ | 280 |
| 図 43. | シーケンスの連続変換、ハードウェアトリガ | 280 |
| 図 44. | データの配置と分解能（オーバーサンプリング無効：OVSE = 0） | 281 |
| 図 45. | オーバーラン（OVR）の例 | 282 |
| 図 46. | ウェイトモード変換（連続モード、ソフトウェアトリガ） | 284 |
| 図 47. | WAIT=0、AUTOFF=1 での動作 | 285 |
| 図 48. | WAIT=1、AUTOFF=1 での動作 | 285 |
| 図 49. | アナログウォッチドッグによって保護される領域 | 286 |

| | | |
|--------|---------------------------------------|-----|
| 図 50. | 20 ビットから 16 ビットへの結果の切り詰め | 287 |
| 図 51. | 5 ビットシフトと丸めの数値例 | 288 |
| 図 52. | トリガオーバーサンプリングモード (TOVS ビット = 1) | 290 |
| 図 53. | 温度センサおよび VREFINT チャンネルのブロック図 | 291 |
| 図 54. | DAC ブロック図 | 311 |
| 図 55. | シングル DAC チャンネルモードのデータレジスタ | 312 |
| 図 56. | トリガ無効 (TEN = 0) 時の変換タイミング図 | 313 |
| 図 57. | DAC LFSR レジスタ計算アルゴリズム | 315 |
| 図 58. | LFSR 波形生成による DAC 変換 (SW トリガ有効) | 315 |
| 図 59. | DAC 三角波生成 | 316 |
| 図 60. | 三角波生成による DAC 変換 (SW トリガ有効) | 316 |
| 図 61. | コンパレータ 1 および 2 のブロック図 | 325 |
| 図 62. | LCD コントローラのブロック図 | 333 |
| 図 63. | 1/3 バイアス、1/4 デューティ | 335 |
| 図 64. | スタティックデューティの事例 1 | 336 |
| 図 65. | スタティックデューティの事例 2 | 337 |
| 図 66. | 1/2 デューティ、1/2 バイアス | 338 |
| 図 67. | 1/3 デューティ、1/3 バイアス | 339 |
| 図 68. | 1/4 デューティ、1/3 バイアス | 340 |
| 図 69. | 1/8 デューティ、1/4 バイアス | 341 |
| 図 70. | 1/2 1/3 1/4 バイアスの場合の VLCD ピン | 344 |
| 図 71. | デッドタイム | 345 |
| 図 72. | フローチャートの例 | 347 |
| 図 73. | TSC ブロック図 | 361 |
| 図 74. | 表面電荷移動アナログ I/O グループの構造 | 362 |
| 図 75. | サンプリングコンデンサの電圧の変化 | 363 |
| 図 76. | 電荷移動取得シーケンス | 364 |
| 図 77. | スペクトル拡散の変動の原理 | 365 |
| 図 78. | ブロック図 | 380 |
| 図 79. | ECB 暗号化モード | 382 |
| 図 80. | ECB 復号化モード | 383 |
| 図 81. | CBC モードの暗号化 | 384 |
| 図 82. | CBC モードの復号化 | 384 |
| 図 83. | サスペンドモードの管理例 | 385 |
| 図 84. | CTR モードの暗号化 | 386 |
| 図 85. | CTR モードの復号化 | 386 |
| 図 86. | 32 ビットカウンタ + ノンス構成 | 387 |
| 図 87. | データ型に応じた 128 ビットブロックの構成 | 389 |
| 図 88. | データ型に応じた 128 ビットブロックの構成 (続き) | 390 |
| 図 89. | モード 1: 暗号化 | 390 |
| 図 90. | モード 2: キー派生 | 391 |
| 図 91. | モード 3: 復号化 | 392 |
| 図 92. | モード 4: キー派生と復号化 | 392 |
| 図 93. | 入力フェーズ中の DMA リクエストとデータ転送 (AES_IN) | 393 |
| 図 94. | 出力フェーズ中の DMA リクエスト (AES_OUT) | 394 |
| 図 95. | ブロック図 | 405 |
| 図 96. | 汎用タイマのブロック図 | 411 |
| 図 97. | プリスケアラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 | 412 |
| 図 98. | プリスケアラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 | 413 |
| 図 99. | 内部クロック分周比が 1 の場合のカウンタのタイミング図 | 414 |
| 図 100. | 内部クロック分周比が 2 の場合のカウンタのタイミング図 | 414 |
| 図 101. | 内部クロック分周比が 4 の場合のカウンタのタイミング図 | 415 |

| | | |
|--------|--|-----|
| 図 102. | 内部クロック分周比が N の場合のカウンタのタイミング図 | 415 |
| 図 103. | ARPE=0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタの タイミング図 | 416 |
| 図 104. | ARPE=1 (TIMx_ARR はプリロードされる) のときの更新イベント時のカウンタの タイミング図 | 416 |
| 図 105. | 内部クロック分周比が 1 の場合のカウンタのタイミング図 | 417 |
| 図 106. | 内部クロック分周比が 2 の場合のカウンタのタイミング図 | 418 |
| 図 107. | 内部クロック分周比が 4 の場合のカウンタのタイミング図 | 418 |
| 図 108. | 内部クロック分周比が N の場合のカウンタのタイミング図 | 419 |
| 図 109. | 繰り返しカウンタが使用されていないときの更新イベント時のカウンタのタイミング図 | 419 |
| 図 110. | 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図 | 421 |
| 図 111. | 内部クロック分周比が 2 の場合のカウンタのタイミング図 | 421 |
| 図 112. | 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図 | 422 |
| 図 113. | 内部クロック分周比が N の場合のカウンタのタイミング図 | 422 |
| 図 114. | ARPE=1 (カウンタアンダーフロー) の場合の更新イベント時、カウンタタイミング図 | 423 |
| 図 115. | ARPE=1 (カウンタオーバーフロー) の場合の更新イベント時のカウンタのタイミング図 | 423 |
| 図 116. | 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図 | 424 |
| 図 117. | TI2 外部クロックの接続例 | 425 |
| 図 118. | 外部クロックモード 1 の制御回路 | 426 |
| 図 119. | 外部トリガ入力ブロック | 426 |
| 図 120. | 外部クロックモード 2 の制御回路 | 427 |
| 図 121. | キャプチャ/比較チャンネル (例: チャンネル 1 入力ステージ) | 428 |
| 図 122. | キャプチャ/比較チャンネル 1 メイン回路 | 428 |
| 図 123. | キャプチャ/比較チャンネル (チャンネル 1) の出力ステージ | 429 |
| 図 124. | PWM 入力モードタイミング | 431 |
| 図 125. | 出力比較モード、OC1 のトグル | 433 |
| 図 126. | エッジアライン PWM 波形 (ARR=8) | 434 |
| 図 127. | センターアライン PWM 波形 (ARR=8) | 435 |
| 図 128. | ワンパルスモードの例 | 436 |
| 図 129. | TIMx_OCxREF のクリア | 438 |
| 図 130. | エンコーダインタフェースモードにおけるカウンタの動作例 | 440 |
| 図 131. | TI1FP1 の極性を反転したエンコーダインタフェースモードの例 | 440 |
| 図 132. | リセットモードの制御回路 | 441 |
| 図 133. | ゲートモードの制御回路 | 442 |
| 図 134. | トリガモードの制御回路 | 443 |
| 図 135. | 外部クロックモード 2 + トリガモードの制御回路 | 444 |
| 図 136. | マスタ/スレーブタイマの例 | 444 |
| 図 137. | タイマ x の OC1REF によるタイマ y のゲート操作 | 445 |
| 図 138. | タイマ x のイネーブルによるタイマ y のゲート操作 | 446 |
| 図 139. | タイマ x の更新によるタイマ y のトリガ | 447 |
| 図 140. | タイマ x の有効化によるタイマ y のトリガ | 447 |
| 図 141. | タイマ x の TI1 入力によるタイマ x および y のトリガ | 449 |
| 図 142. | 汎用タイマのブロック図 (TIM21/22) | 474 |
| 図 143. | プリスケアラ分周比が 1 から 2 に変化したときのカウンタのタイミング図 | 476 |
| 図 144. | プリスケアラ分周比が 1 から 4 に変化したときのカウンタのタイミング図 | 476 |
| 図 145. | 内部クロック分周比が 1 の場合のカウンタのタイミング図 | 477 |
| 図 146. | 内部クロック分周比が 2 の場合のカウンタのタイミング図 | 478 |
| 図 147. | 内部クロック分周比が 4 の場合のカウンタのタイミング図 | 478 |
| 図 148. | 内部クロック分周比が N の場合のカウンタのタイミング図 | 479 |
| 図 149. | ARPE = 0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタの タイミング図 | 479 |

| | | |
|--------|---|-----|
| 図 150. | ARPE = 1 (TIMx_ARR がプリロードされる) の場合の更新イベント時のカウンタの タイミング図..... | 480 |
| 図 151. | 内部クロック分周比が 1 の場合のカウンタのタイミング図..... | 481 |
| 図 152. | 内部クロック分周比が 2 の場合のカウンタのタイミング図..... | 481 |
| 図 153. | 内部クロック分周比が 4 の場合のカウンタのタイミング図..... | 482 |
| 図 154. | 内部クロック分周比が N の場合のカウンタのタイミング図..... | 482 |
| 図 155. | 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図..... | 484 |
| 図 156. | 内部クロック分周比が 2 の場合のカウンタのタイミング図..... | 484 |
| 図 157. | 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図..... | 485 |
| 図 158. | 内部クロック分周比が N の場合のカウンタのタイミング図..... | 485 |
| 図 159. | ARPE = 1 (カウンタアンダーフロー) の場合の更新イベント時のカウンタのタイミング図..... | 486 |
| 図 160. | ARPE = 1 (カウンタオーバーフロー) の場合の更新イベント時のカウンタのタイミング図..... | 486 |
| 図 161. | 内部クロック分周比 1 の場合の、通常モードの制御回路..... | 487 |
| 図 162. | TI2 外部クロックの接続例..... | 488 |
| 図 163. | 外部クロックモード 1 の制御回路..... | 489 |
| 図 164. | 外部トリガ入力ブロック..... | 489 |
| 図 165. | 外部クロックモード 2 の制御回路..... | 490 |
| 図 166. | キャプチャ/比較チャンネル (例: チャンネル 1 入力ステージ)..... | 491 |
| 図 167. | キャプチャ/比較チャンネル 1 メイン回路..... | 491 |
| 図 168. | キャプチャ/比較チャンネル (チャンネル 1、2) の出力ステージ..... | 492 |
| 図 169. | PWM 入力モードのタイミング..... | 494 |
| 図 170. | 出力比較モード、OC1 の反転..... | 495 |
| 図 171. | エッジアライン PWM 波形 (ARR = 8)..... | 497 |
| 図 172. | センターアライン PWM 波形 (ARR = 8)..... | 498 |
| 図 173. | TIMx_OCxREF のクリア..... | 499 |
| 図 174. | ワンパルスモードの例..... | 500 |
| 図 175. | エンコーダインタフェースモードにおけるカウンタの動作例..... | 503 |
| 図 176. | TI1FP1 の極性を反転したエンコーダインタフェースモードの例..... | 503 |
| 図 177. | リセットモードの制御回路..... | 504 |
| 図 178. | ゲートモードの制御回路..... | 505 |
| 図 179. | トリガモードの制御回路..... | 506 |
| 図 180. | 基本タイマブロック図..... | 527 |
| 図 181. | プリスケアラ分周比が 1 から 2 に変化したときのカウンタのタイミング図..... | 529 |
| 図 182. | プリスケアラ分周比が 1 から 4 に変化したときのカウンタのタイミング図..... | 529 |
| 図 183. | 内部クロック分周比が 1 の場合のカウンタのタイミング図..... | 530 |
| 図 184. | 内部クロック分周比が 2 の場合のカウンタのタイミング図..... | 531 |
| 図 185. | 内部クロック分周比が 4 の場合のカウンタのタイミング図..... | 531 |
| 図 186. | 内部クロック分周比が N の場合のカウンタのタイミング図..... | 532 |
| 図 187. | ARPE = 0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタの タイミング図..... | 532 |
| 図 188. | ARPE = 1 (TIMx_ARR がプリロードされる) の場合の更新イベント時のカウンタの タイミング図..... | 533 |
| 図 189. | 内部クロック分周比 1 の場合の、通常モードの制御回路..... | 534 |
| 図 190. | 低電力タイマのブロック図..... | 541 |
| 図 191. | グリッチフィルタのタイミング図..... | 542 |
| 図 192. | 波形生成..... | 545 |
| 図 193. | エンコーダモードのカウントシーケンス..... | 548 |
| 図 194. | 独立型ウォッチドッグのブロック図..... | 559 |
| 図 195. | ウォッチドッグのブロック図..... | 569 |
| 図 196. | ウィンドウ型ウォッチドッグのタイミング図..... | 570 |
| 図 197. | RTC ブロック図..... | 575 |
| 図 198. | I2C1 ブロック図..... | 621 |

| | | |
|--------|--|-----|
| 図 199. | I2C2 ブロック図 | 622 |
| 図 200. | I2C バスプロトコル | 623 |
| 図 201. | セットアップおよびホールドタイミング | 625 |
| 図 202. | I2C 初期化フローチャート | 628 |
| 図 203. | データ受信 | 629 |
| 図 204. | データ送信 | 630 |
| 図 205. | スレーブ初期化フローチャート | 633 |
| 図 206. | I2C スレーブトランスミッタの転送シーケンスフローチャート (NOSTRETCH=0) | 635 |
| 図 207. | I2C スレーブトランスミッタの転送シーケンスフローチャート (NOSTRETCH=1) | 636 |
| 図 208. | I2C スレーブトランスミッタの転送バス図 | 637 |
| 図 209. | スレーブレシーバの転送シーケンスフローチャート (NOSTRETCH=0) | 638 |
| 図 210. | スレーブレシーバの転送シーケンスフローチャート (NOSTRETCH=1) | 639 |
| 図 211. | I2C スレーブレシーバの転送バス図 | 639 |
| 図 212. | マスタクロック生成 | 641 |
| 図 213. | マスタ初期化フローチャート | 643 |
| 図 214. | HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス | 643 |
| 図 215. | HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス | 644 |
| 図 216. | N<=255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート | 645 |
| 図 217. | N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート | 646 |
| 図 218. | I2C マスタトランスミッタの転送バス図 | 647 |
| 図 219. | N<=255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート | 649 |
| 図 220. | N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート | 650 |
| 図 221. | I2C マスタレシーバの転送バス図 | 651 |
| 図 222. | t _{LOW:SEXT} 、t _{LOW:MEXT} のタイムアウト間隔 | 655 |
| 図 223. | N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート | 659 |
| 図 224. | SMBus スレーブトランスミッタの転送バス図 (SBC=1) | 660 |
| 図 225. | N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート | 661 |
| 図 226. | SMBus スレーブレシーバのバス転送図 (SBC=1) | 662 |
| 図 227. | SMBus マスタトランスミッタのバス転送図 | 663 |
| 図 228. | SMBus マスタレシーバのバス転送図 | 665 |
| 図 229. | I2C 割り込みマッピング図 | 670 |
| 図 230. | USART のブロック図 | 692 |
| 図 231. | ワード長のプログラミング | 694 |
| 図 232. | 設定可能なストップビット | 695 |
| 図 233. | 送信時の TC/TXE の動作 | 697 |
| 図 234. | 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出 | 698 |
| 図 235. | データサンプリング (16 倍のオーバーサンプリング) | 702 |
| 図 236. | データサンプリング (8 倍のオーバーサンプリング) | 702 |
| 図 237. | アイドルライン検出を使用したミュートモード | 708 |
| 図 238. | アドレスマーク検出を使用したミュートモード | 709 |
| 図 239. | LIN モードでのブレイク検出 (11 ビットブレイク長、LBDL=1) | 713 |
| 図 240. | LIN モードでのブレイク検出とフレーミングエラー検出 | 714 |
| 図 241. | USART の同期送信の例 | 715 |
| 図 242. | USART データクロックタイミング図 (M ビット = 00) | 715 |
| 図 243. | USART データクロックタイミング図 (M ビット = 01) | 716 |
| 図 244. | RX データセットアップ/ホールド時間 | 716 |
| 図 245. | ISO 7816-3 非同期プロトコル | 717 |
| 図 246. | ストップビット 1.5 個を使用したパリティエラー検出 | 719 |
| 図 247. | IrDA SIR ENDEC- ブロック図 | 723 |
| 図 248. | IrDA データ変調 (3/16) - 通常モード | 723 |
| 図 249. | DMA を使用した送信 | 724 |
| 図 250. | DMA を使用した受信 | 725 |

| | | |
|--------|--|-----|
| 図 251. | 2 つの USART 間のハードウェアフロー制御 | 726 |
| 図 252. | RS232 RTS フロー制御 | 726 |
| 図 253. | RS232 CTS フロー制御 | 727 |
| 図 254. | USART 割り込みマッピング図 | 729 |
| 図 255. | LPUART ブロック図 | 758 |
| 図 256. | ワード長のプログラミング | 760 |
| 図 257. | 設定可能なストップビット | 761 |
| 図 258. | 送信時の TC / TXE の動作 | 763 |
| 図 259. | アイドルライン検出を使用したミュートモード | 768 |
| 図 260. | アドレスマーク検出を使用したミュートモード | 769 |
| 図 261. | DMA を使用した送信 | 771 |
| 図 262. | DMA を使用した受信 | 772 |
| 図 263. | 2 つの LPUART 間のハードウェアフロー制御 | 773 |
| 図 264. | RS232 RTS フロー制御 | 773 |
| 図 265. | RS232 CTS フロー制御 | 774 |
| 図 266. | LPUART 割り込みマッピング図 | 776 |
| 図 267. | SPI ブロック図 | 795 |
| 図 268. | 全二重シングルマスタ/シングルスレーブアプリケーション | 796 |
| 図 269. | 半二重シングルマスタ/シングルスレーブアプリケーション | 797 |
| 図 270. | 単方向シングルマスタ/シングルスレーブアプリケーション (送信専用モードのマスタ/受信専用モードのスレーブ) | 798 |
| 図 271. | マスタと 3 つの独立したスレーブ | 799 |
| 図 272. | ハードウェア/ソフトウェアスレーブ選択管理 | 800 |
| 図 273. | データクロックのタイミング図 | 802 |
| 図 274. | マスタ/全二重モードでの TXE/RXNE/BSY 動作 (BIDIMODE = 0、RXONLY = 0) 連続転送の場合 | 805 |
| 図 275. | スレーブ/全二重モードでの TXE/RXNE/BSY 動作 (BIDIMODE = 0、RXONLY = 0) 連続転送の場合 | 806 |
| 図 276. | DMA を使用した送信 | 808 |
| 図 277. | DMA を使用した受信 | 808 |
| 図 278. | TI モードでの転送 | 811 |
| 図 279. | I ² S ブロック図 | 814 |
| 図 280. | フィリップス I ² S プロトコルの波形 (16/32 ビットフル精度、CPOL = 0) | 816 |
| 図 281. | フィリップス I ² S 規格の波形 (24 ビットフレーム、CPOL = 0) | 816 |
| 図 282. | 0x8EAA33 の送信 | 816 |
| 図 283. | 0x8EAA33 の受信 | 817 |
| 図 284. | フィリップス I ² S 規格 (32 ビットパケットフレームに拡張された 16 ビット、CPOL = 0) | 817 |
| 図 285. | 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例 | 817 |
| 図 286. | MSB 詰め 16 ビットまたは 32 ビットフル精度長、CPOL = 0 | 818 |
| 図 287. | MSB 詰め 24 ビットフレーム長、CPOL = 0 | 818 |
| 図 288. | MSB 詰め 16 ビットを 32 ビットパケットフレームに拡張、CPOL = 0 | 818 |
| 図 289. | LSB 詰め 16 ビットまたは 32 ビットフル精度、CPOL = 0 | 819 |
| 図 290. | LSB 詰め 24 ビットフレーム長、CPOL = 0 | 819 |
| 図 291. | 0x3478AE を送信するために必要な動作 | 819 |
| 図 292. | 0x3478AE の受信に必要な動作 | 820 |
| 図 293. | LSB 詰め 16 ビットを 32 ビットパケットフレームに拡張、CPOL = 0 | 820 |
| 図 294. | 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例 | 820 |
| 図 295. | PCM 規格の波形 (16 ビット) | 821 |
| 図 296. | PCM 規格の波形 (32 ビットパケットフレームに拡張された 16 ビット) | 821 |
| 図 297. | オーディオサンプリング周波数の定義 | 822 |
| 図 298. | I ² S クロックジェネレータのアーキテクチャ | 822 |
| 図 299. | USB ペリフェラルのブロック図 | 841 |

| | | |
|--------|--|-----|
| 図 300. | パケットバッファ領域とバッファ記述テーブルの位置の例 | 845 |
| 図 301. | ブロック図 - STM32L0x3 MCU および Cortex®-M0+ レベルのデバッグサポート | 873 |



1 このマニュアルにおける表記の規則

1.1 レジスタに関する略記

レジスタの説明では、次の略記が使用されます。

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

1.2 用語

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

- **セクタ**：コード領域内の32 ページの書き込み保護単位。
- **ページ**：コード領域およびシステムメモリ領域では 32 ワード、データ領域、工場オプション領域、ユーザオプション領域では 1 ワード。
- **ワード**：32 ビット長のデータ。
- **ハーフワード**：16 ビット長のデータ。
- **バイト**：8 ビット長のデータ。
- **IAP (アプリケーション内プログラミング)**：IAP は、ユーザプログラム実行中にマイクロコントローラのフラッシュメモリを再プログラムする機能です。
- **ICP (インサーキットプログラミング)**：ICP は、ユーザアプリケーションボードにデバイスが搭載された状態で、JTAG プロトコルや SWD プロトコルまたはブートローダを用いて、マイクロコントローラのフラッシュメモリをプログラムする機能です。
- **オプションバイト**：フラッシュメモリに格納された製品設定ビット。
- **OBL**：オプションバイトローダ。
- **AHB**：アドバンスドハイパフォーマンスバス。
- **NVM**：不揮発性メモリ。
- **ECC**：エラーコード訂正。
- **DMA**：ダイレクトメモリアクセス。
- **MIF**：NVM インタフェース。
- **PCROP**：独自仕様コード読み出し保護。

1.3 使用可能なペリフェラル

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

1.4 製品カテゴリの定義

表 1 は、製品ファミリ別のメモリ容量の概要を示します。

表 1. STM32L0x3 のメモリ容量

| メモリ容量 | カテゴリ1 | カテゴリ2 | カテゴリ3 |
|--------|-------|--------------------------------|-------|
| 16 KB | - | - | - |
| 32 KB | - | STM32L053x STM32L063x (AES) | - |
| 64 KB | - | STM32L053x STM32L063x (AES) | - |
| 128 KB | - | - | - |
| 192 KB | - | - | - |

2 システムおよびメモリの概要

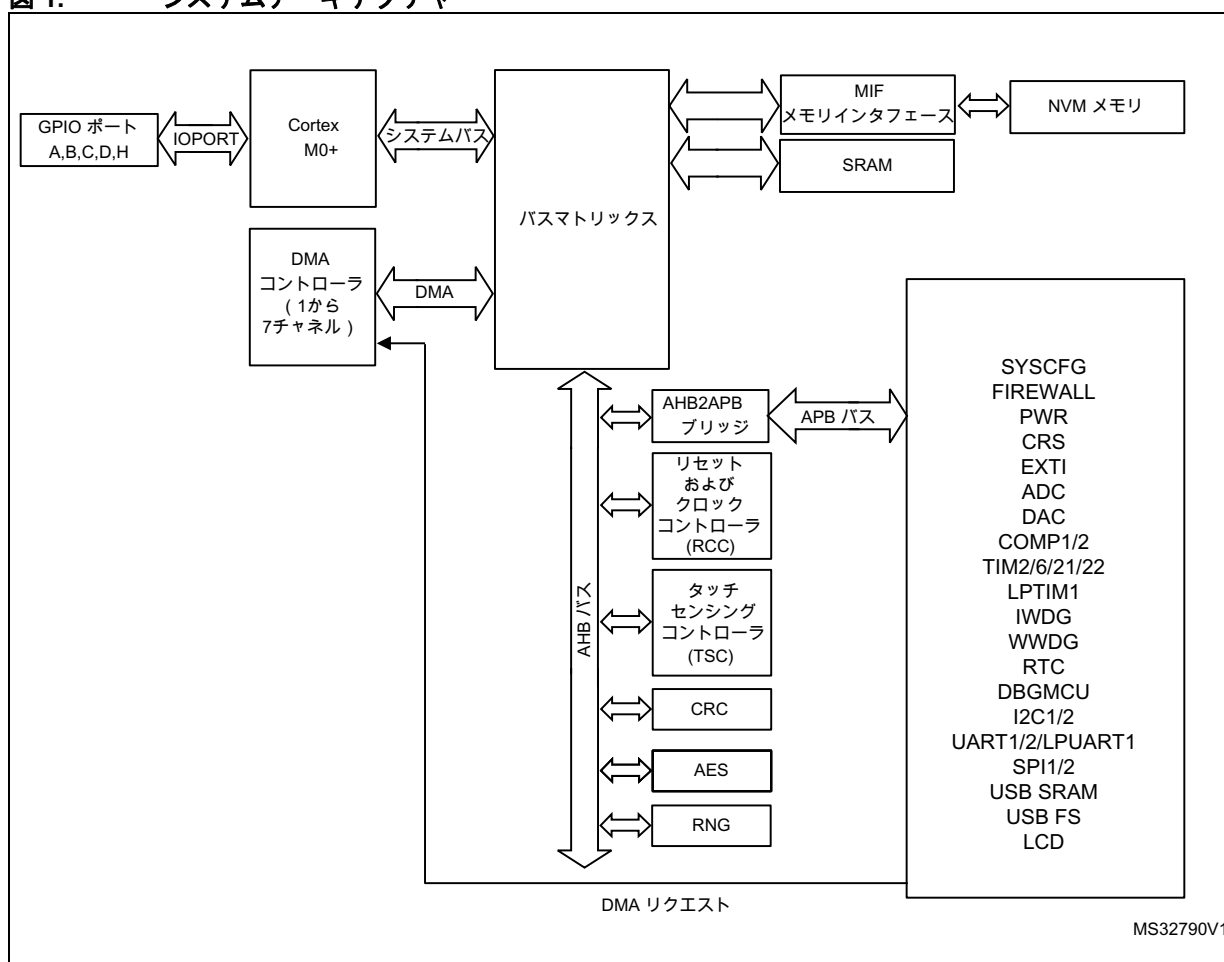
2.1 システムアーキテクチャ

メインシステムの構成は次のとおりです。

- 2つのマスタ：
 - Cortex[®]-M0+ コア (AHB-Lite バス)
 - GP-DMA (汎用 DMA)
- 3つのスレーブ：
 - 内蔵 8 KB SRAM
 - 内蔵 64 KB フラッシュメモリ
 - AHB - APB、すべての APB ペリフェラルに接続

図 1 に示すように、マルチレイヤ AHB バスアーキテクチャを使用して相互接続します。

図 1. システムアーキテクチャ



2.1.1 S0 : Cortex バス

このバスによって、Cortex[®]-M0+ コアの D-Code/I-Code バスが バスマトリックスに接続されます。コアは、このバスを使用して命令をフェッチし、データを取得し、AHB/APB リソースにアクセスします。

2.1.2 S1 : DMA バス

このバスによって DMA の AHB マスタインタフェースが バスマトリックス に接続され、異なるマスタのフラッシュメモリとデータ EEPROM、SRAM、および AHB/APB ペリフェラルへのアクセスが管理されます。

2.1.3 バスマトリックス

このバスマトリックスによって、マスタ間のアクセス調停を管理します。調停には、ラウンドロビン方式を使用します。バスマトリックスは 2 つのマスタ (CPU、DMA) および 3 つのスレーブ (NVM インタフェース、SRAM、AHB2APB1/2 ブリッジ) で構成されています。

AHB/APB ブリッジ

この AHB/APB ブリッジでは、AHB および 2 つの APB バス間に完全同期接続を提供します。APB1 および APB2 は、最大周波数 32 MHz で動作します。

このブリッジに接続されたペリフェラルのアドレスマッピングについては、[セクション 2.2.2 : メモリマップとレジスタ境界アドレス \(43 ページ\)](#) を参照してください。

各デバイスのリセット後、すべてのペリフェラルクロックは無効になります (SRAM と MIF は除く)。ペリフェラルを使用する前に、RCC_AHBENR、RCC_APB2ENR、RCC_APB1ENR、または RCC_IOPENR レジスタでクロックを有効にする必要があります。

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

2.2 メモリ構成

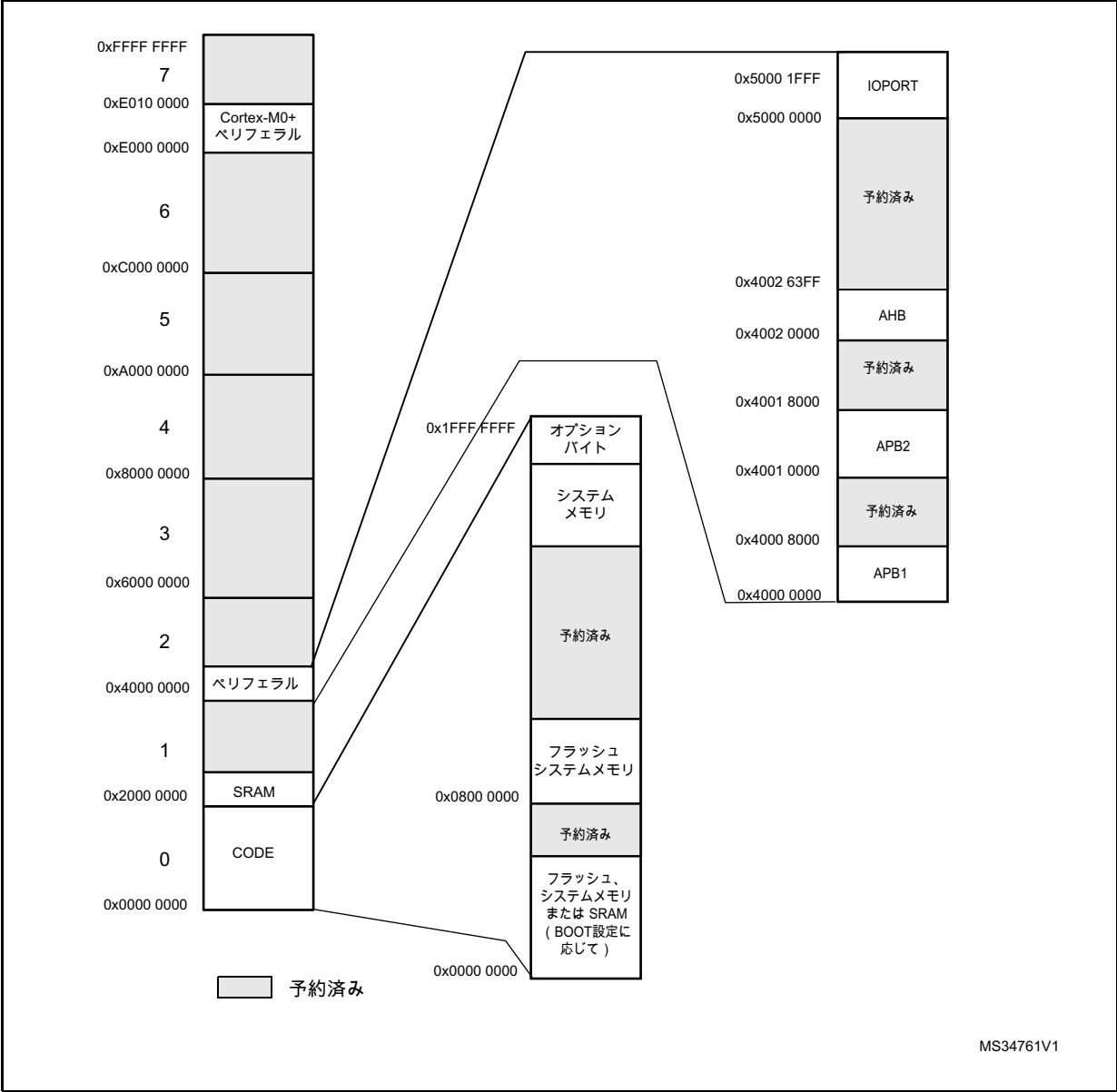
2.2.1 概要

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

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

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

図 2. メモリマップ



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

[ジスタ境界アドレス](#)および周辺の章を参照してください。

2.2.2 メモリマップとレジスタ境界アドレス

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

次の表に、デバイスで使用可能なペリフェラルの境界アドレスを示します。

表 2. STM32L0x3 ペリフェラルレジスタ境界アドレス

| バス | 境界アドレス | サイズ (バイト) | ペリフェラル | ペリフェラルレジスタマップ |
|--------|---------------------------|-----------|------------------------|---|
| IOPORT | 0X5000 1C00 - 0X5000 1FFF | 1K | GPIOH | セクション 9.4.12 : GPIO レジスタマップ |
| | 0X5000 1000 - 0X5000 1BFF | 3K | 予約済み | |
| | 0X5000 0C00 - 0X5000 0FFF | 1K | GPIOD | セクション 9.4.12 : GPIO レジスタマップ |
| | 0X5000 0800 - 0X5000 0BFF | 1K | GPIO C | セクション 9.4.12 : GPIO レジスタマップ セクション 9.4.12 : GPIO レジスタマップ |
| | 0X5000 0400 - 0X5000 07FF | 1K | GPIOB | セクション 9.4.12 : GPIO レジスタマップ |
| | 0X5000 0000 - 0X5000 03FF | 1K | GPIOA | セクション 9.4.12 : GPIO レジスタマップ |
| AHB | 0X4002 6400 - 0X4002 67FF | 49K | 予約済み | |
| | 0X4002 6000 - 0X4002 63FF | 1 K | AES (カテゴリ 2、AES のみ) | セクション 19.12.13 : AES レジスタマップ |
| | 0X4002 5400 - 0X4002 5FFF | 3 K | 予約済み | |
| | 0X4002 5000 - 0X4002 53FF | 1 K | RNG | セクション 20.4.4 : RNG レジスタマップ |
| | 0X4002 4400 - 0X4002 4FFF | 3 K | 予約済み | |
| | 0X4002 4000 - 0X4002 43FF | 1 K | TSC | セクション 18.6.11 : TSC レジスタマップ |
| | 0X4002 3400 - | 3 K | 予約済み | |
| | 0X4002 3000 - 0X4002 33FF | 1 K | CRC | セクション 4.4.6 : CRC レジスタマップ |
| | 0X4002 2400 - 0X4002 2FFF | 3 K | 予約済み | |
| | 0X4002 2000 - 0X4002 23FF | 1 K | FLASH | セクション 3.7.10 : フラッシュレジスタマップ |
| | 0X4002 1400 - 0X4002 1FFF | 3 K | 予約済み | |
| | 0X4002 1000 - 0X4002 13FF | 1 K | RCC | セクション 7.3.22 : RCC レジスタマップ |
| | 0X4002 0400 - 0X4002 0FFF | 3 K | 予約済み | |
| | 0X4002 0000 - 0X4002 03FF | 1 K | DMA1 | セクション 11.4.8 : DMA レジスタマップ |

表 2. STM32L0x3 ペリフェラルレジスタ境界アドレス（続き）

| バス | 境界アドレス | サイズ (バイト) | ペリフェラル | ペリフェラルレジスタマップ |
|------|----------------------------|-----------|----------|-----------------------------------|
| APB2 | 0X4001 5C00 - 0X4001 FFFF | 42K | 予約済み | |
| | 0X4001 5800 - 0X4001 5BFF | 1 K | DBG | セクション 33.10 : DBG レジスタマップ |
| | 0X4001 3C00 - 0X4001 57FF | 7 K | 予約済み | |
| | 0X4001 3800 - 0X4001 3BFF | 1 K | USART1 | セクション 29.7.12 : USART レジスタマップ |
| | 0X4001 3400 - 0X4001 37FF | 1 K | 予約済み | |
| | 0X4001 3000 - 0X4001 33FF | 1 K | SPI1 | セクション 31.7.10 : SPI レジスタマップ |
| | 0X4001 2800 - 0X4001 2FFF | 2 K | 予約済み | |
| | 0X4001 2400 - 0X4001 27FF | 1 K | ADC1 | セクション 14.13.12 : ADC レジスタマップ |
| | 0X4001 2000 - 0X4001 23FF | 1 K | 予約済み | |
| | 0X4001 1C00 - 0X4001 1FFF | 1 K | ファイアウォール | セクション 5.4.8 : ファイアウォールレジスタマップ |
| | 0X4001 1800 - 0X4001 1BFF | 1 K | 予約済み | |
| | 0X4001 1400 - 0X4001 17FF | 1 K | TIM22 | セクション 22.4.16 : TIM21/22 のレジスタマップ |
| | 0X4001 0C000 - 0X4001 13FF | 2 K | 予約済み | |
| | 0X4001 0800 - 0X4001 0BFF | 1 K | TIM21 | セクション 22.4.16 : TIM21/22 のレジスタマップ |
| | 0X4001 0400 - 0X4001 07FF | 1 K | EXTI | セクション 13.5.7 : EXTI レジスタマップ |
| | 0X4001 0000 - 0X4001 03FF | 1 K | SYSCFG | セクション 10.2.8 : SYSCFG レジスタマップ |



この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

表 2. STM32L0x3 ペリフェラルレジスタ境界アドレス（続き）

| バス | 境界アドレス | サイズ (バイト) | ペリフェラル | ペリフェラルレジスタマップ |
|------|----------------------------|-----------|-----------------------|--------------------------------|
| APB1 | 0X4000 8000 - 0X4000 FFFF | 32 K | 予約済み | |
| | 0X4000 7C00 - 0X4000 7FFF | 1 K | LPTIM1 | セクション 24.6.9 : LPTIM レジスタマップ |
| | 0X4000 7800 - 0X4000 7BFF | 1 K | 予約済み | |
| | 0X4000 7400 - 0X4000 77FF | 1 K | DAC1 | セクション 15.7.8 : DAC レジスタマップ |
| | 0X4000 7000 - 0X4000 73FF | 1 K | PWR | セクション 6.4.3 : PWR レジスタマップ |
| | 0X4000 6C00 - 0X4000 6FFF | 1 K | CRS | セクション 8.6.5 : CRS レジスタマップ |
| | 0X4000 6800 - 0X4000 6BFF | 1 K | 予約済み | |
| | 0X4000 6000 - 0X4000 67FF | 2 K | USB (SRAM 512x16 ビット) | |
| | 0X4000 5C00 - 0X4000 5FFF | 1 K | USB FS | セクション 32.6.4 : USB レジスタマップ |
| | 0X4000 5800 - 0X4000 5BFF | 1 K | I2C2 | セクション 28.7.12 : I2C レジスタマップ |
| | 0X4000 5400 - 0X4000 57FF | 1 K | I2C1 | セクション 28.7.12 : I2C レジスタマップ |
| | 0X4000 4C000 - 0X4000 53FF | 2 K | 予約済み | |
| | 0X4000 4800 - 0X4000 4BFF | 1 K | LPUART1 | セクション 30.6.10 : LPUART レジスタマップ |
| | 0X4000 4400 - 0X4000 47FF | 1 K | USART2 | セクション 29.7.12 : USART レジスタマップ |
| | 0X4000 3C000 - 0X4000 43FF | 2 K | 予約済み | |
| | 0X4000 3800 - 0X4000 3BFF | 1 K | SPI2 | セクション 31.7.10 : SPI レジスタマップ |
| | 0X4000 3400 - 0X4000 37FF | 1 K | 予約済み | |
| | 0X4000 3000 - 0X4000 33FF | 1 K | IWDG | セクション 25.4.6 : IWDG レジスタマップ |
| | 0X4000 2C00 - 0X4000 2FFF | 1 K | WWDG | セクション 26.4.4 : WWDG レジスタマップ |
| | 0X4000 2800 - 0X4000 2BFF | 1 K | RTC + BKP_REG | セクション 27.6.21 : RTC レジスタマップ |
| | 0X4000 2400 - 0X4000 27FF | 1 K | LCD | セクション 17.6.6 : LCD レジスタマップ |
| | 0X4000 1400 - 0X4000 23FF | 4 K | 予約済み | |
| | 0X4000 1000 - 0X4000 13FF | 1 K | TIMER6 | セクション 23.4.9 : TIM6 レジスタマップ |
| | 0X4000 0400 - 0X4000 0FFF | 3 K | 予約済み | |
| | 0X4000 0000 - 0X4000 03FF | 1 K | TIMER2 | セクション 21.4.20 : TIMx レジスタマップ |

表 2. STM32L0x3 ペリフェラルレジスタ境界アドレス（続き）

| バス | 境界アドレス | サイズ (バイト) | ペリフェラル | ペリフェラルレジスタマップ |
|------|---------------------------|-----------|--|---------------|
| SRAM | 0X2000 2000 - 0X3FFF FFFF | ~524 M | 予約済み | |
| | 0X2000 0000 - 0X2000 1FFF | 8 K | SRAM | - |
| NVM | 0X0800 0000 - 0X0800 FFFF | 64 K | フラッシュ プログラム メモリ | - |
| | 0x0808 0000 - 0x0808 07FF | 2 K | データ EEPROM | - |
| | 0x1FF8 0000 - 0x1FF0 0FFF | 4 K + 32 | 情報ブロック (システム メモリ、ユーザ および工場 オプション バイト) | - |

2.3 内蔵 SRAM

STM32L0x3 デバイスには、8 KB のSRAM を搭載しています。

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

SRAM の開始アドレスは 0x2000 0000 です。

CPU は、SRAM からのブートが選択されている場合、または物理的な再割当てが選択されている場合に、システムバスまたは I-Code/D-Code バスを通じて SRAM にアクセスできます（SYSCFG コントローラの [セクション 10.2.1 : SYSCFG メモリ再配置レジスタ \(SYSCFG_CFGR1\)](#) レジスタを参照）。SRAM 実行の最高のパフォーマンスを実現するには、物理的な再割当てを選択する必要があります（ブートまたはソフトウェア選択）。

2.4 ブート設定

STM32L0x3 では、次の表に示すように、ユーザオプションバイトで BOOT0 ピンおよび nBOOT1 ビットによって 3 種類のブートモードを選択できます。

表 3. ブートモード

| ブートモード選択 | | ブートモード | エイリアシング |
|----------------------|-------|---------------|-------------------------------|
| BOOT1 ⁽¹⁾ | BOOT0 | | |
| x | 0 | フラッシュプログラムメモリ | フラッシュプログラムメモリがブート領域として選択されます。 |
| 0 | 1 | システムメモリ | システムメモリがブート領域として選択されます。 |
| 1 | 1 | 内蔵 SRAM | 内蔵 SRAM がブート領域として選択されます。 |

1. BOOT1 の値は、nBOOT1 オプションビットの逆の値です。



BOOT0 ピンと nBOOT1 ビットの両方の値は、リセット後、SYSCLK の 4 番目の立上りエッジでラッチされます。ユーザは、nBOOT1 と BOOT0 をセットして必要なブートモードを指定します。

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

選択したブートメモリに応じて、フラッシュプログラムメモリ、システムメモリ、または SRAM に、次のようにアクセスできます。

- フラッシュプログラムメモリからブート：フラッシュプログラムメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x0800 0000）からアクセスすることも可能です。言い換えると、フラッシュメモリの内容は、0x0000 0000 または 0x0800 0000 から始まるアドレスからアクセスできます。
- システムメモリからブート：システムメモリのエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x1FF0 0000）からアクセスすることも可能です。
- 内蔵 SRAM からブート：SRAM のエイリアスがブートメモリ空間（0x0000 0000）に作成されますが、引き続き元のメモリ空間（0x2000 0000）からアクセスすることも可能です。

物理的な再割当て

ブートピンとビットを選択すると、アプリケーションソフトウェアはコード領域でアクセス可能なメモリを変更できます。この変更は、SYSCFG メモリ再配置レジスタ（SYSCFG_CFGR1）に MEM_MODE ビットをプログラミングすることによって実行できます。

内蔵ブートローダ

内蔵ブートローダは、システムメモリに配置されるよう生産時に ST によってプログラムされています。次のシリアルインタフェースのいずれかを使用して、フラッシュメモリを再プログラムするために使用します。USART1（PA9、PA10）、USART2（PA2、PA3）、SPI1（PA4、PA5、PA6、PA7）、または SPI2（PB12、PB13、PB14、PB15）

詳細については、AN2606 を参照してください。

3 フラッシュプログラムメモリとデータ EEPROM (FLASH)

3.1 概要

不揮発性メモリ (NVM) の構成は次のとおりです。

- 16 Kワード (16 K×32 ビット) で構成される、最大 64 KB のフラッシュプログラムメモリこの領域は、アプリケーションコードを保管するために使用します。
- 2 KB のデータ EEPROM
- 情報ブロック：
 - － 最大 4 KB のシステムメモリ
 - － 最大 8×4 バイトのユーザオプションバイト
 - － 最大 96 バイトの工場オプションバイト

3.2 NVM の主な機能

NVM インタフェースの機能：

- ワード、ハーフワード、またはバイトで構成された各領域での読み出しインタフェース
- ワードまたはハーフページで実行されるフラッシュメモリのプログラミング
- ワードで実行されるオプションバイト領域のプログラミング
- ワード、ハーフワード、またはバイトで実行されるデータ EEPROM のプログラミング
- ページ単位の消去操作 (フラッシュメモリ、データ EEPROM、およびオプションバイト)
- オプションバイトローダ
- ECC (エラー訂正コード)：ただ 1 つのエラーを検出して修正するために各ワードに 6 ビット格納
- 全体消去操作
- 読み出し／書込み保護
- PCROP 保護
- 低電力モード

3.3 NVM の機能詳細

3.3.1 NVM 構成

NVM は、コード、データ、ブートコード、またはオプションバイトを保管するために使用できる、32 ビットのメモリセルとして構成されています。

メモリアレイは、ページ単位に分割されています。1 ページは、フラッシュプログラムメモリおよびシステムメモリで 32 ワード (または 128 バイト)、データ EEPROM およびオプションバイト領域 (ユーザおよび工場) で 1 シングルワード (または 4 バイト) で構成されています。

フラッシュセクタは 32 ページ (または 4 KB) で構成されています。セクタは書込み保護の粒度を表します。

表 4. NVM 構成 (STM32L0x3 デバイス)

| NVM | NVM アドレス | サイズ (バイト) | 名前 | 説明 |
|-------------------|---------------------------|-----------|---------|-------------|
| フラッシュ プログラムメモリ | 0x0800 0000 - 0x0800 007F | 128 バイト | 0 ページ | セクタ 0 |
| | 0x0800 0080 - 0x0800 00FF | 128 バイト | 1 ページ | |
| | - | - | - | |
| | 0x0800 0F80 - 0x0800 0FFF | 128 バイト | 31 ページ | |
| | ⋮ | ⋮ | ⋮ | ⋮ |
| | 0x0800 7000 - 0x0800 707F | 128 バイト | 224 ページ | セクタ 7 |
| | 0x0800 7080 - 0x0800 70FF | 128 バイト | 225 ページ | |
| | - | - | - | |
| | 0x0800 7F80 - 0x0800 7FFF | 128 バイト | 255 ページ | |
| | ⋮ | ⋮ | ⋮ | ⋮ |
| | 0x0800 F000 - 0x0800 F07F | 128 バイト | 480 ページ | セクタ 15 |
| | 0x0800 F080 - 0x0800 F0FF | 128 バイト | 481 ページ | |
| | - | - | - | |
| | 0x0800 FF80 - 0x0800 FFFF | 128 バイト | 511 ページ | |
| データ EEPROM | 0x0808 0000 - 0x0808 07FF | 2 KB | | データ EEPROM |
| 情報ブロック | 0x1FF0 0000 - 0x1FF0 0FFF | 4 KB | | システムメモリ |
| | 0x1FF8 0020 - 0x1FF8 007F | 96 バイト | | 工場オプション |
| | 0x1FF8 0000 - 0x1FF8 001F | 32 バイト | | ユーザオプションバイト |

3.3.2 NVM の読み出し

読み出しプロトコル

NVM の内容を読み出すには、表 4 から任意のアドレスを使用します。メモリインタフェースのクロックは実行中である必要があります。

クロックの周波数に応じて、NVM の読み出しに 0 または 1 のウェイトステートが必要となる場合があります。

ウェイトステート数 (FLASH_ACR レジスタの LATENCY ビット) は、正しくセットする必要があります。ウェイトステート数に関して、使用する周波数または電力が正しいかどうかを検証する制御は行われません。ウェイトステート数の設定を誤ると、誤った読み出し値が生成されるか (高い周波数と 0 ウェイトステート)、コードの実行に時間がかかります (低い周波数と 1 ウェイトステート)。

NVM は、ワード (4 バイト)、ハーフワード (2 バイト)、またはバイトで読み出すことができます。

NVM にはバンクが 1 つしかありません。これは、書き込み/消去操作中は読み出すことができないことを意味しています。書き込み/消去操作が進行中の場合は、読み出しはこの操作が完了するまでウェイトステートとなり、アドレスが読み出し保護されている場合を除いて、読み出し動作を要求したマスタを停止します。読み出し保護の場合、バスエラーまたはメモリインタフェースフラグによってマスタにエラーが送信されます。これにより、マスタの停止処理なく、読み出しのウェイトもありません。

CPU 周波数／動作モード／NVM 読み出し時間の関係

デバイス（および NVM）は異なる電源範囲で動作します。すべての範囲にいくつかのマスタクロック周波数をセットできます。NVM に正しい時間でアクセスできるよう、表 5 に電源範囲と周波数の関係を示します。

表 5. マスタクロック電源範囲と周波数の関係

| 名前 | 電源範囲 | 最大周波数 (1 ウェイトステート) | 最大周波数 (ウェイトステートなし) |
|------|-----------------|-----------------------|-----------------------|
| レンジ1 | 1.65 V - 1.95 V | 32 MHz | 16 MHz |
| レンジ2 | 1.35 V - 1.65 V | 16 MHz | 8 MHz |
| レンジ3 | 1.05 V - 1.35 V | 4.2 MHz | 4.2 MHz |

表 6 に、NVM におけるワードの読み出し遅延時間を示します。ワードの読み出しにかかる全体時間 (Ttotal) とクロック周期を比較すると、レンジ3 ではデバイスで許容される最大周波数 (4.2 MHz) を適用しても、ウェイトステートが不要であることが分かります。Ttotal は、NVM が値を返すまでに必要とする時間であり、読み出し（メモリインタフェースを介してメモリからコアまで）にかかる全体時間ではありません。残りのすべての時間は失われます。

表 6. メモリアクセス遅延およびウェイトステート数

| 名前 | Ttotal | 周波数 | 周期 | 必要な ウェイトステート数 |
|------|----------|--------|-------|------------------|
| レンジ1 | 46.1 ns | 32 MHz | 31.25 | 1 |
| | | 16 MHz | 62.5 | 0 |
| レンジ2 | 86.8 ns | 16 MHz | 62.5 | 1 |
| | | 8 MHz | 125 | 0 |
| レンジ3 | 184.6 ns | 4 MHz | 250 | 0 |
| | | 2 MHz | 500 | 0 |

CPU 周波数の変更

リセット後、使用されるクロックは MSI (2.1 MHz) であり、FLASH_ACR レジスタでは 0 ウェイトステートが設定されます。CPU 周波数に合わせて NVM にアクセスするために必要なウェイトステート数の調整するには、以下のソフトウェアシーケンスに従う必要があります。

CPU クロックやウェイトステート数の設定の変更が有効になるまでには、時間がかかる場合があります。AHB プリスケールファクターとクロックソースステータス値を確認することで、設定されている正しい CPU クロック周波数を確認できます。同様に、FLASH_ACR を読み出すことで、プログラムしたウェイトステートが有効であることを確認できます。

CPU 周波数の増加（同じ電圧範囲内）

1. 必要に応じて、FLASH_ACR レジスタの LATENCY ビットに 1 ウェイトステートをプログラムします。
2. FLASH_ACR レジスタを読み出し、新しいウェイトステート数が考慮されていることを確認します。ウェイトステート数が変更されると、メモリインタフェースでは NVM への読み出しアクセス方法を修正します。読み出し動作が進行中の場合はウェイトステート数を修正できないため、メモリインタフェースは NVM で読み出しが行われなくなるまで待ちます。マスタが FLASH_ACR レジスタの内容を読み戻した場合、この読み出し（および、読み出しを要求したマスタ）は、実際にウェイトステート数が変更されるまで停止します。ユーザがレジスタを読み



戻さない場合、次の NVM へのアクセスはクロック周波数を増加していても 0 ウェイトステートで実行され、結果として誤った値となります。

- リセットおよびクロックコントローラ (RCC) の CPU クロックソースや AHB クロックプリスケラを修正します。
- リセットおよびクロックコントローラ (RCC) でクロックソースステータスや AHB プリスケラの値を読み出して、新しい CPU クロックソースや新しい CPU クロックプリスケラの値がそれぞれ考慮されていることを確認します。いくつかのクロックは使用可能になるまでに時間がかかるため、この確認は重要です。

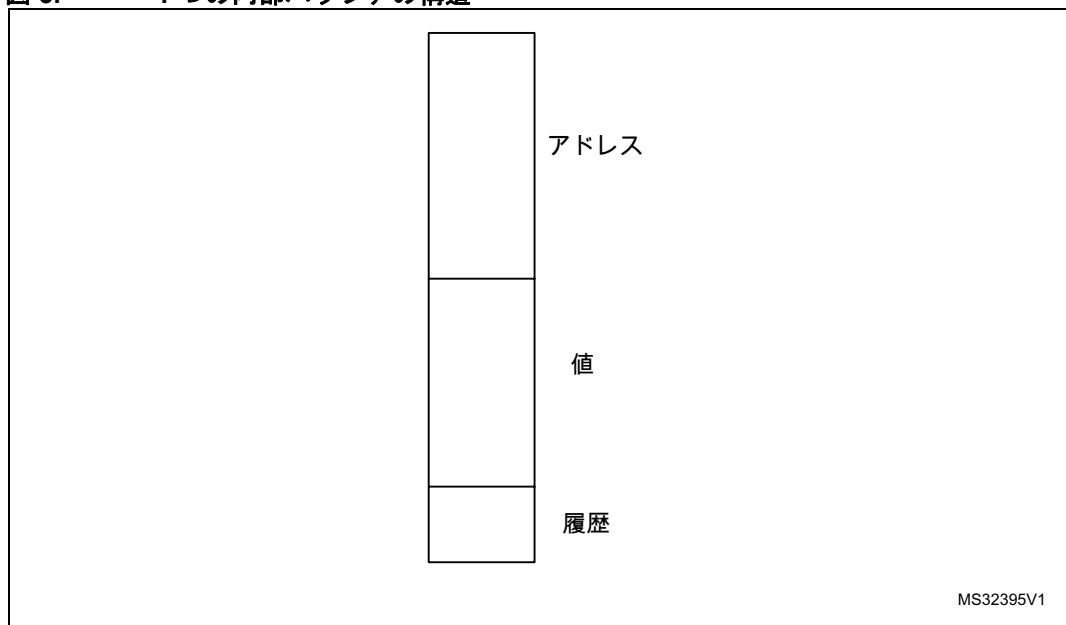
CPU 周波数の減少 (同じ電圧範囲内)

- リセットおよびクロックコントローラ (RCC) の CPU クロックソースや AHB クロックプリスケラを修正します。
- リセットおよびクロックコントローラ (RCC) でクロックソースステータスや AHB プリスケラの値を読み出して、新しい CPU クロックソースや新しい CPU クロックプリスケラの値がそれぞれ考慮されていることを確認します。
- 必要に応じて、FLASH_ACR レジスタの LATENCY ビットに 0 ウェイトステートをプログラムします。
- FLASH_ACR レジスタを読み出し、新しいウェイトステート数が考慮されていることを確認します。前の節で説明した理由で、レジスタを読み戻す必要があります。

データのバッファリング

NVM では、読み出し動作中に 6 つのバッファ (フェッチ用およびデータ用の両方) でパフォーマンスに影響を与えることができます (また、ある条件下で消費電力の削減に役立ちます)。1 つのバッファの構造を [図 3](#) に示します。

図 3. 1 つの内部バッファの構造



各バッファには、3 つの異なるタイプの情報 (アドレス、データ、および履歴) が格納されています。読み出し動作で、アドレスを検出した場合、メモリインタフェースは NVM にアクセスすることなくデータを返すことができます。バッファに格納されたデータは 32 ビット幅であるため (マスタが 8 ビットまたは 16 ビットしか読み出さない場合でも)、前回の読み出し時に使用されたサイズにかかわ

らず、値を返すことができます。履歴は、バッファの内容が有効であるかどうかを確認し、(新しい値で) 古い値を上書きするために使用します。

バッファは、通常の読み出し動作中に NVM で受け取った値を格納するため、また投機的読み出しを行うために使用します。投機的読み出しを無効化すると、バッファが有効になっている場合は (デフォルト)、マスタに要求されたデータのみが格納されます。これにより、すでにバッファに値が格納されている場合はウェイトステートが不要となるため、パフォーマンスを向上することができます。また、メモリでの読み出し数が削減され、メモリからのすべての組み合わせパスが安定することから、消費電力を削減することもできます。

バッファは、さまざまなタスクを管理するためにグループに分けられています。ユーザが選択した設定によって、各グループに含まれるバッファの数は変わります (表 7 を参照)。使用するバッファの合計数は、常に 6 です (有効な場合)。履歴は必ずグループが管理します。

メモリインタフェースでは、バッファのグループを確認せずに特定のアドレスがすべてのバッファにあるか、また、読み出しがフェッチであるか、データであるかについて、常に検索しています。

リセットまたはいくつかのアドレスを変更する書き込み/消去操作を行うと、バッファはすべて空になり、履歴は EMPTY に設定されます。ワード、ハーフワード、またはバイトで実行されるプログラムを行うと、関係するアドレスを含むバッファのみが空になります。

表 7. 内部バッファ管理

| DISAB_BUF | PREFTEN | PRE_READ | バッファ (フェッチ用) | | | バッファ (データ用) | |
|-----------|---------|----------|--------------|----------------|--------------|----------------|--------------|
| | | | バッファ (ジャンプ用) | バッファ (プリフェッチ用) | バッファ (最後の値用) | バッファ (事前読み出し用) | バッファ (最後の値用) |
| 1 | - | - | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 3 | 0 | 1 | 0 | 2 |
| 0 | 1 | 0 | 2 | 1 | 1 | 0 | 2 |
| 0 | 0 | 1 | 3 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 2 | 1 | 1 | 1 | 1 |

バッファ内の値が空ではない場合、履歴には読み出したとき、または書き込んだときからの経過時間が示されます。履歴は、一番新しい値から一番古い値までをリストして構成されています。ある時点で特定の履歴の値を持つのは、グループ内で 1 つのバッファだけです (空の値を除く)。バッファを最新の位置に移動すると、グループ内の他のすべてのバッファも 1 ステップずつ進み、順序を維持します。履歴は、バッファの読み出し時 (マスタがバッファの内容を要求) または書き込み時 (NVM の新しい値) に、最新の位置に変更されます。メモリインタフェースでは、メモリに新しいアドレスが必要になると、必ず正しいグループの一番古いバッファ (ある場合は、空のバッファ 1 つ) を書き込みます。

バッファリングの管理には、FLASH_ACR レジスタの 3 つの設定ビットが使用されます。

- DISAB_BUF
このビットをセットすると、すべてのバッファが無効になります。このビットが 1 の時は、プリフェッチ動作や事前読み出し動作を有効にすることはできず、たとえばマスタが同じアドレスを 2 回要求した場合などは、NVM に 2 つの読み出しが生成されます。
- PRFTEN
このビットを 1 (DISAB_BUF は 0) にセットすると、プリフェッチが有効になります。メモリインタフェースで進行中の動作がない場合は、最後にフェッチしたアドレスの次のアドレスを読み出して、バッファに格納します。

- PRE_READ
このビットを 1 (DISAB_BUF は 0) にセットすると、事前読み出しが有効になります。メモリインタフェースで進行中の動作または実行するプリフェッチがない場合は、最後のデータアドレスの次のアドレスを読み出してバッファに格納します。

フェッチとプリフェッチ

メモリインタフェースのフェッチとは、オペレーションを実行するための NVM からの読み出しです。メモリインタフェースでは、読み出し動作を実行するマスタや読み出し位置の確認はしませんが、唯一、読み出し動作が行われたかどうかを検証します（読み出した内容が実行されるため）。これは、フェッチを次のように実行できることを意味します。

- 全領域で実行
- 全サイズ（16 または 32 ビット）で実行

メモリインタフェースでは、バッファに次の情報を格納しています。

- ジャンプアドレス。ループ時に初回だけ NVM にアクセスすれば、次回以降はジャンプアドレスを使用できます。
- 最後の読み出しアドレス。16 ビットでフェッチを実行すると、他の 16 ビットも使用できます。

フェッチを管理するために、メモリインタフェースではリセット時に 4 つのバッファを使用します (DISAB_BUF = 0、PRFTEN = 0、PRE_READ = 0)。3 つのバッファはジャンプを管理するため、1 つのバッファはフェッチした最後の値を格納するために使用されます。この設定により、フェッチで使用する 4 つのバッファは別々の履歴を持つ 2 つのグループ（ループ用のグループとフェッチした最後の値用のグループ）に分けられます。

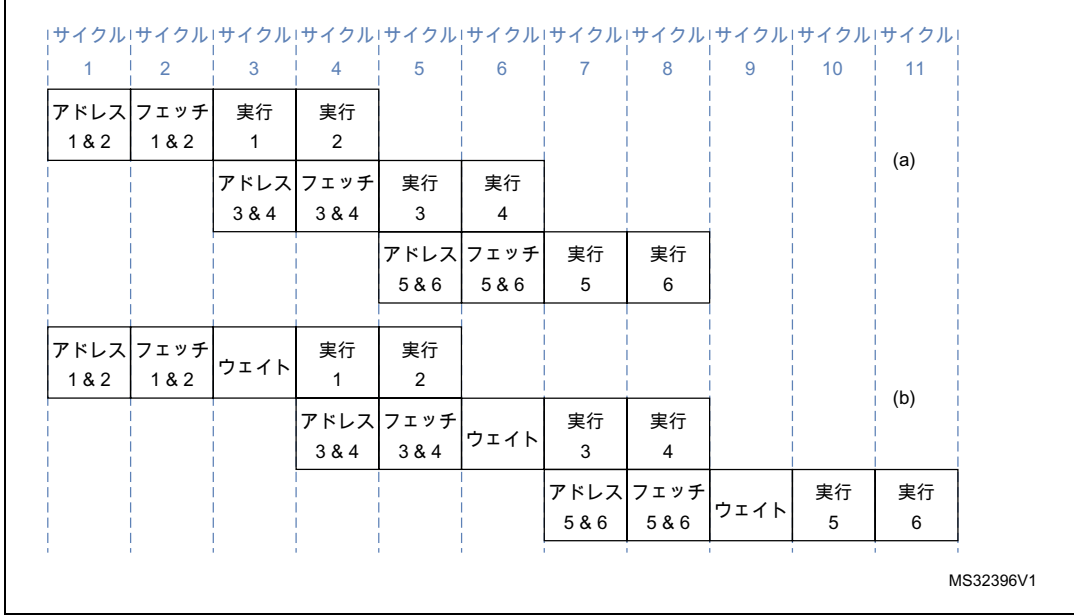
PRFTEN ビットを 1 にセットすると、プリフェッチが有効になります。プリフェッチは NVM の投機的読み出しです。マスタから読み出しが要求されず、メモリインタフェースがフェッチした最後のアドレス（4 バイト（1 ワード）増加した）から読み出す場合に実行されます。この読み出しの優先順位は低く、マスタがプリフェッチしたアドレスとは別のアドレスに読み出し（データまたはフェッチ）を要求すると、アボートされます。プリフェッチを有効にすると、ループ用のバッファが 1 つ、プリフェッチした値を格納するために新しいグループ（1 つのバッファだけで構成される）に移されます。2 つのバッファは引き続きジャンプを格納し、1 つのバッファはプリフェッチに使用され、もう 1 つのバッファは最後の値に使用されます。

メモリインタフェースでプリフェッチできるアドレスは 1 つだけです。フェッチが行われず、プリフェッチがすでに完了している場合、この機能は一時的に無効化されます。プリフェッチ後、マスタがプリフェッチした値を要求した場合は、プリフェッチバッファの内容が最後の値用のバッファにコピーされ、新しいプリフェッチが有効になります。代わりに、マスタが別のアドレスを要求した場合、プリフェッチバッファの内容は失われ、必要に応じて NVM の読み出しが始まります。完了すると、フェッチした新しいアドレスに 4 増加して新しいプリフェッチが有効になります。

プリフェッチでパフォーマンスを向上できるのは、1 ウェイトステートで読み出しているときと、ほぼ線形コードに対してのみです。ユーザは、すべての状況でプリフェッチを有効化するかどうかの是非を評価する必要があります。NVM でさらに多くの読み出しが行われるようになるため（すべてがマスタで使用されるわけではない）、プリフェッチによって消費電力が増加します。ドライストーンコードでのプリフェッチの利点を確認するには、[ドライストーン性能セクション](#)を参照してください。

プリフェッチが無効な場合に NVM の線形コードをフェッチするタイミングを、0 ウェイトステート (a) および 1 ウェイトステート (b) の両方について、[図 4](#)に示します。プリフェッチが有効な場合に、0 ウェイトステートおよび 1 ウェイトステートの線形コードでプリフェッチする利点を理解するために、この 2 つのシーケンスを[図 5](#)の内容と比較することができます。

図 4. 命令をフェッチして実行するタイミング (プリフェッチ無効)



1. (a) 0 ウェイトステートに対応
2. (b) 1 ウェイトステートに対応

プリフェッチが有効化された場合に NVM から命令をフェッチして実行するタイミングを、0 ウェイトステート (a) および 1 ウェイトステート (b) の両方について、図 5 に示します。プリフェッチが実行した読み出しは、緑色で示されています。

データとしての読み出しおよび事前読み出し

メモリーインタフェースから読み出されたデータは、フェッチ以外のすべての読み出し動作に対応します。マスタでは、動作定数とパラメータをデータとして読み出します。DMA による読み出し (アドレス間のコピー) は、すべてデータとして読み出されます。読み出したデータの位置では確認は行われません (NVM の領域内の任意の位置の場合がある)。

リセット時 (DISAB_BUF = 0、PRFTEN = 0、PRE_READ = 0)、メモリーインタフェースでは 1 つのグループとして構成された 2 つのバッファを使用して、データとして最後に読み出された 2 つの値を格納します。

特殊なケースでは (たとえば DMA によって NVM から大量の連続したワードが読み出される場合など)、事前読み出しを有効化すると有用である場合があります (PRE_READ = 1、DISAB_BUF = 0)。事前読み出しは、最後のデータアドレス (4 バイト (1 ワード) 増加した) における、投機的読み出しであり、プリフェッチとまったく同じように動作します。この設定により、事前読み出し値を格納するために 1 つのバッファのデータは新しいグループに移され、2 番目のバッファには引き続き読み出した最後の値が格納されます。プリフェッチの場合、マスタがそのアドレスを要求すると事前読み出し値は最後の読み出し値にコピーされ、マスタが別のアドレスを要求すると失われます。

事前読み出しの優先順位は、通常の読み出しやプリフェッチ動作よりも低く、他のタイプの読み出しが進行中でない場合のみ、開始します。誤った状況における事前読み出しの使用は有害である点に注意してください。データの読み出しが線形に行われなかったコードでは、最後の読み出し値に使用したバッファ数を削減すると (2 から 1 に削減)、NVM へのアクセス数 (および値の読み出しにかかる時間) が増加します。さらに、これによってプリフェッチに遅延が生成されます。この状況の例として、ドライストーンコードが挙げられます。この結果については、対応するセクションを参照してください。

プリフェッチ動作については、必ずユーザが事前読み出しを有効化および無効化するタイミングを選択する必要があります。

図 5. 命令をフェッチして実行するタイミング（プリフェッチ有効）

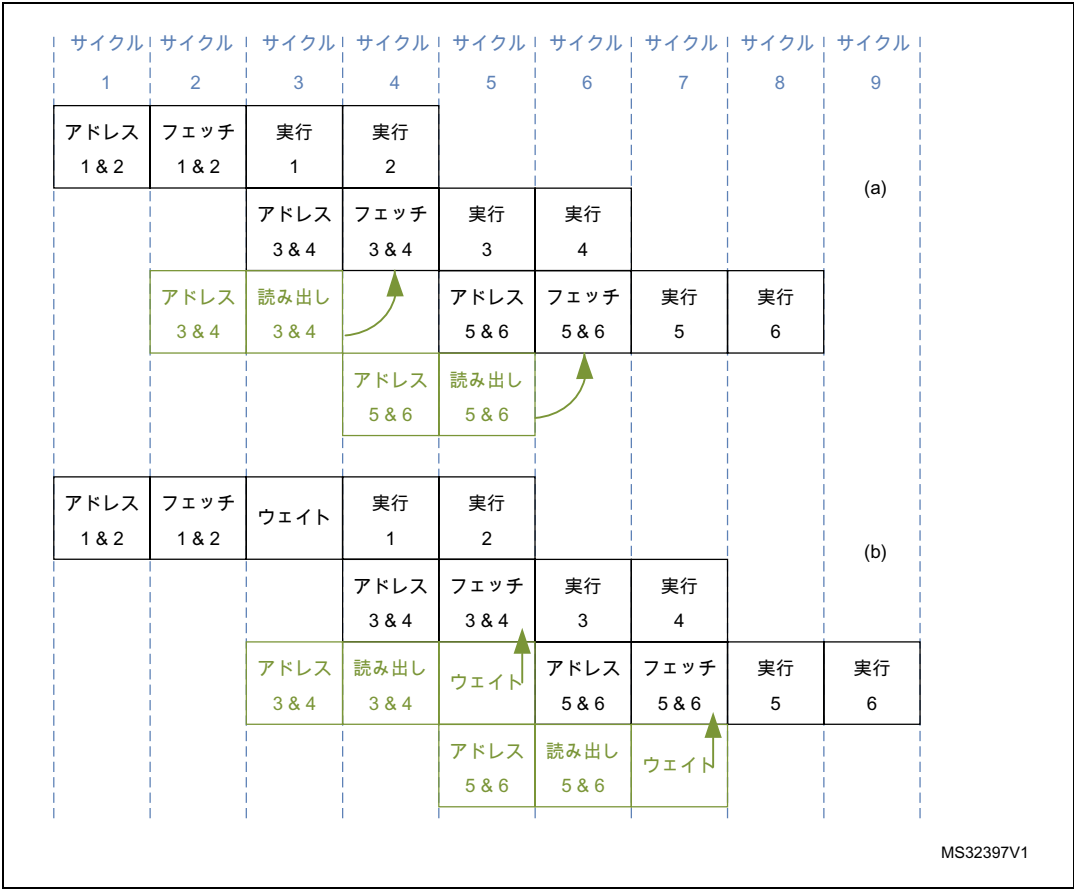


表 8 に、実現可能な設定の概要を示します。

表 8. バッファの設定と投機的読み出し

| DISAB_BUF | PRFTEN | PRE_READ | 説明 |
|-----------|--------|----------|---|
| 1 | X | X | バッファ無効 |
| 0 | 0 | 0 | バッファ有効。投機的読み出しなし |
| 0 | 1 | 0 | プリフェッチ有効。フェッチでの投機的読み出し有効 |
| 0 | 0 | 1 | 事前読み出し有効。データでの投機的読み出し有効 |
| 0 | 1 | 1 | プリフェッチおよび事前読み出し有効。フェッチおよびデータでの投機的読み出し有効 |

ドライストーン性能

ドライストーンテストは、メモリインタフェース性能を評価するために使用されます。このテストは、すべてのメモリインタフェース設定で実行されています。結果の概要については、表 9 を参照してください。

一般的なパラメータは次のとおりです。

- マトリックスのサイズ 20 x 20
- ループの実行回数 1757 回
- ARM コンパイラのバージョン 4.1 [ビルド 561]

次に、結果について簡単に説明します。

表 9. すべてのメモリインタフェース設定でのドライストーン性能

| ウェイト ステート数 | DISAB_BUF | PRFTEN | PRE_READ | DMIPS 数 (x1000) | DMIPS x MHz |
|---------------|-----------|--------|----------|--------------------|-------------|
| 0 | 1 | 0 | 0 | 953 | 15.25 |
| 0 | 0 | 0 | 0 | 953 | 15.25 |
| 0 | 0 | 1 | 0 | 953 | 15.25 |
| 0 | 0 | 0 | 1 | 953 | 15.25 |
| 0 | 0 | 1 | 1 | 953 | 15.25 |
| 1 | 1 | 0 | 0 | 677 | 21.66 |
| 1 | 0 | 0 | 0 | 690 | 22.08 |
| 1 | 0 | 1 | 0 | 823 | 26.34 |
| 1 | 0 | 0 | 1 | 691 | 22.11 |
| 1 | 0 | 1 | 1 | 816 | 26.11 |

- 事前読み出しは、このテストには適していません。プリフェッチありで有効化すると、最後に読み出されたデータを格納するためにバッファが 1 つしか使用されず、またこのコードでは、マスタがデータを線形に読み出すことがほとんどないため、メモリインタフェースのパフォーマンスが低下します。これによって、プリフェッチなしで有効化されたとき、パフォーマンスがわずかに向上します。
- 1 ウェイトステートのバッファ（投機的読み出しなし）では、コストなしで検討することができます。
- 0 ウェイトステートでは、コンパイル中の異なるコードの配置によって、最高のパフォーマンス（ARM 認定）が得られます。

3.3.3 NVM の書き込み／消去

NVM の内容を変更するには、さまざまな方法があります。メモリインタフェースは、不要な変更が発生する可能性を削減し、ハードウェアによって、消去または書き込みに必要なすべてのシーケンスを異なるメモリ領域に実装するために役立ちます。

書き込み／消去プロトコル

保護が解除されたときにメモリの内容を書き込み／消去するには、次の手順に従ってください。

1. 実行する動作を設定します。
2. NVM に 1 つまたは複数のアドレスを書き込んで、メモリインタフェースに正しいデータ数を送信します。
3. 動作の完了を待ちます。

待っている間に、新しい設定を書き込んだり、次の書き込み／消去動作のデータの書き込みを開始したりして、次の動作を準備できます（特殊なケースを除く）。

待ち時間は、動作のタイプに依存します。書き込み／消去動作の時間は、 T_{prog} (3.2 ms) から $2 \times T_{glob}$ (3.7 ms) + T_{prog} (3.2 ms)の間です。メモリインタフェースは、1 回の待ち時間だけでハーフページ（フラッシュプログラムメモリで 16 ワード）書き込むように設定できます。これにより、大量のデータのプログラムにかかる時間を削減できます。

シングルプログラミング動作とマルチプログラミング動作の 2 つの異なるプロトコルを使用できます。

シングルプログラミング動作

このプロトコルでは、ソフトウェアが NVM の保護されていないアドレスに値を書き込む必要があります。メモリインタフェースが書き込み要求を受信すると、いくつかのクロックパルスの間、マスタを停止（詳細については、表 10 を参照）して保護状態と前回の値を確認し、NVM 内に新しい値をラッチします。その後、ソフトウェアでは次の動作の設定を開始できます。FLASH_SR レジスタの EOP ビットが 1 となれば、動作は完了します（動作開始時に 0 だった場合）。すべての動作の動作時間を、表 12 に示します。

マルチプログラミング動作（ハーフページ）

フラッシュプログラムメモリにハーフページ（16 ワード）を書き込むことができます。このプロトコルを実行するには、次の条件を満たす必要があります。

- FLASH_SR レジスタの PGAERR ビットがゼロであること（前回のアライメントエラーがないこと）
- 最初のアドレスが、ハーフページで整列されていること（アドレスの 6 個の下位ビットはゼロである必要がある）
- 16 ワードは、すべて同じハーフページ内にあること（16 ワードすべてについて、アドレスビット 7 から 31 が同じであること）これは、最初のアドレスでハーフページをセットすると、以降のアドレスをこのハーフページ内にセットする必要があることを意味しています。書き込まれたデータは、以降のアドレスに順番に格納されます。アドレスは、メモリインタフェースによって内部的に自動で増加するため、アドレスの増加や変更は重要ではありません（たとえば、同じアドレスを 16 回使用することができます）。
- ワード（32 ビット）のみを書き込むことができます。

メモリインタフェースが最初のアドレスを受信すると、いくつかのクロックパルスの間、マスタを停止して保護状態と前回の値を確認し、NVM 内に新しい値をラッチします（詳細については、表 10 を参照）。次に、メモリインタフェースでは 2 番目のアドレスを待ちます。読み出しは受け付けません。フェッチのみが実行されますが、実行中の書き込み動作をアボートします。2 番目のアドレスの後で、次のアドレスを待つ前に、メモリインタフェースではコアを短い間（前回よりも短い時間）停止して

確認を行い、NVM 内にラッチします。このシーケンスは 16 ワードがすべて NVM 内にラッチされるまで続きます。アライメントやサイズを誤ると、書き込み動作がアボートされます。16 個のアドレスが正しくラッチされると、メモリインタフェースでは書き込み動作を開始します。FLASH_SR レジスタの EOP ビットが 1 となれば、動作は完了します（動作開始時に 0 だった場合）。動作時間を、表 12 に示します。

このプロトコルは、RAM から実行されるアプリケーションコード、または RAM またはコアシリーピングから実行されるアプリケーションコードを持つ DMA のいずれかを介して使用できます。

アンロック／ロック動作

書き込み／消去動作を実行する前に、有効化する必要があります。ユーザはフラッシュプログラムメモリ、データ EEPROM、およびオプションバイト領域に書き込むことができます。

書き込み／消去動作を実行するには、FLASH_PECR レジスタの PELOCK ビットをアンロックします。ビットをアンロックすると（値 0）、同じレジスタの他のビットを修正できます。PELOCK が 0 である場合、書き込み／消去動作をデータ EEPROM で実行できます。

フラッシュプログラムメモリに書き込む／消去するには、FLASH_PECR レジスタの PRGLOCK ビットをアンロックします。このビットは、PELOCK が 0 である場合にのみアンロックできます。

ユーザオプションバイトに書き込む／消去するには、FLASH_PECR レジスタの OPTLOCK ビットをアンロックします。このビットは、PELOCK が 0 である場合にのみアンロックできます。PRGLOCK と OPTLOCK に関連性はありません。1 つ目をアンロックした場合、2 つ目はロックされます（逆も同様）。

データ EEPROM および FLASH_PECR レジスタのアンロック

リセット後、FLASH_PECR レジスタの PELOCK ビットがセットされているため、データ EEPROM および FLASH_PECR レジスタに書き込みモードでアクセスすることはできません。同じアンロックシーケンスでは、同時に両方の保護を解除します。

次のシーケンスは、データ EEPROM および FLASH_PECR レジスタをアンロックするために使用します。

- FLASH_PEKEYR レジスタに PEKEY1 = 0x89ABCDEF を書き込みます。
- FLASH_PEKEYR レジスタに PEKEY2 = 0x02030405 を書き込みます。

キーシーケンスを誤ると、次のリセットまで FLASH_PECR がロックされ、バスエラーが生成されます。マスタが 2 つのキーシーケンス間に別のレジスタに書き込もうとした場合や誤ったキーを使用した場合も同様です。読み出しアクセスではエラーは生成されず、シーケンスに割り込みません。次の 4 つのケースすべてにおいて、バスエラーが返されます。

- 最初の書き込みアクセス後、入力した PEKEY1 の値がエラーである場合
- 2 番目の書き込みアクセス中、PEKEY1 は正しく入力されるが、PEKEY2 の値が一致しない場合
- PEKEYR の 3 番目の値の書き込みを試みた場合（注意：デバッグでも同様）
- PEKEY1 および PEKEY2 間に、メモリインタフェースの異なるレジスタの書き込みを試みた場合

正しく実行された場合、アンロックシーケンスでは FLASH_PECR レジスタの PELOCK ビットがクリアされます。

FLASH_PECR とデータ EEPROM を再度ロックするには、ソフトウェアでは FLASH_PECR レジスタの PELOCK ビットのセットのみを行う必要があります。再度ロックされた場合、PELOCK ビットでは 0 に戻るための新しいシーケンスが必要です。

フラッシュプログラムメモリのアンロック

フラッシュプログラムメモリの書き込み／消去を行うには、追加の保護が実装されます。

リセット後、フラッシュプログラムメモリは書き込みモードではアクセスできなくなります。FLASH_PECR レジスタの PRGLOCK ビットがセットされます。フラッシュプログラムメモリへの書き込みアクセスは、PRGLOCK ビットをクリアすることで与えられます。

フラッシュプログラムメモリのアンロックには、次のシーケンスを使用します。

- FLASH_PECR レジスタをアンロックします ([データ EEPROM および FLASH_PECR レジスタのアンロック](#) セクションを参照)。
- FLASH_PRGKEYR レジスタに PRGKEY1 = 0x8C9DAEBF を書き込みます。
- FLASH_PRGKEYR レジスタに PRGKEY2 = 0x13141516 を書き込みます。

1 にセットされた PELOCK でキーを書き込んだ場合はエラーは生成されず、PRGLOCK は 1 のまま維持されます。PELOCK = 0 でシーケンスを再実行しているときに、アンロックされます。

キーシーケンスを誤ると、次のリセットまで FLASH_PECR レジスタの PRGLOCK が ロックされ、バスエラーを返します。次の 4 つのケースすべてにおいて、バスエラーが返されます。

- 最初の書き込みアクセス後、入力した PRGKEY1 の値がエラーである場合
- 2 番目の書き込みアクセス中、PRGKEY1 は正しく入力されるが、PRGKEY2 の値が一致しない場合
- PRGKEYR の 3 番目の値の書き込みを試みた場合 (デバッグでも同様)
- PRGKEY1 および PRGKEY2 間に、メモリインタフェースの異なるレジスタの書き込みを試みた場合

正しく実行された場合、アンロックシーケンスでは PRGLOCK ビットがクリアされ、フラッシュプログラムメモリが書き込みアクセス可能になります。

フラッシュプログラムメモリを再度ロックするには、ソフトウェアでは FLASH_PECR レジスタの PRGLOCK ビットのセットのみを行う必要があります。再度ロックされた場合、PRGLOCK ビットでは 0 に戻るための新しいシーケンスが必要です。PELOCK が 1 に戻ると (ロック状態)、PRGLOCK も自動的にロックされます。

オプションバイト領域のアンロック

追加の書き込み保護が、オプションバイト領域に実装されます。オプションバイト領域を再ロードまたは書き込み／消去するには、OPTLOCK をアンロックする必要があります。

リセット後、オプションバイト領域は書き込みモードではアクセスできなくなります。FLASH_PECR レジスタの OPTLOCK ビットがセットされます。オプションバイト領域への書き込みアクセスは、OPTLOCK ビットをクリアすることで与えられます。

オプションバイト領域のアンロックには、次のシーケンスを使用します。

1. FLASH_PECR レジスタをアンロックします ([データ EEPROM および FLASH_PECR レジスタのアンロック](#) セクションを参照)。
2. FLASH_OPTKEYR レジスタに OPTKEY1 = 0xFBEAD9C8 を書き込みます。
3. FLASH_OPTKEYR レジスタに OPTKEY2 = 0x24252627 を書き込みます。

キーが PELOCK = 1 で書き込まれている場合、エラーは生成されず、OPTLOCK は 1 のまま維持されます。PELOCK を 0 にセットしてシーケンスを再実行する際にアンロックされます。

キーシーケンスを誤ると、次のリセットまで FLASH_PECR レジスタの OPTLOCK が ロックされ、バスエラーを返します。次の 4 つのケースすべてにおいて、バスエラーが返されます。

- 最初の書き込みアクセス後、入力した OPTKEY1 の値がエラーである場合
- 2 番目の書き込みアクセス中、OPTKEY1 は正しく入力されるが、OPTKEY2 の値が一致しない場合
- OPTKEYR の 3 番目の値の書き込みを試みた場合（デバッグでも同様）
- OPTKEY1 および OPTKEY2 間に、メモリアンタフェースの異なるレジスタの書き込みを試みた場合

正しく実行された場合、アンロックシーケンスでは OPTLOCK ビットがクリアされ、オプションバイト領域が書き込みアクセス可能になります。

オプションバイト領域を再度ロックするには、ソフトウェアでは FLASH_PECR レジスタの OPTLOCK ビットのセットのみを行う必要があります。再度ロックされた場合、OPTLOCK ビットでは 0 に戻るための新しいシーケンスが必要です。PELOCK が 1 に戻ると（ロック状態）、OPTLOCK も自動的にロックされます。

さまざまなタイプの動作からの選択

必要なアンロックシーケンスを実行すると（PELOCK、PRGLOCK、および OPTLOCK）、ユーザは正しい設定を FLASH_PECR レジスタに書き込むことで、さまざまな書き込み／消去動作を実行することができます。関連するビットは、次のとおりです。

- PRG
- DATA
- FIX
- ERASE
- FPRG

NVM 書き込み／消去動作の詳細な説明

このセクションでは、さまざまな書き込み／消去動作のタイプについて、それぞれに必要なビットを示しながら説明します。

データ EEPROM への書き込み

- **目的**
データ EEPROM に、特定の値を持つワードを 1 つ書き込みます。
- **サイズ**
バイト、ハーフワード、またはワードで書き込みます。
- **アドレス**
データ EEPROM で、有効なアドレスを選択します。
- **プロトコル**
シングルプログラミング動作。
- **リクエスト**
PELOCK = 0。
- **エラー**
PELOCK = 1 またはメモリが読み出し保護されている場合、WRPERR は 1 にセットされます（書き込み動作は実行されません）。

- **説明**

この動作は、ワードまたはワードの一部をデータ EEPROM に書き込むことを目的としています。ユーザは、正しいアドレスに正しいサイズで正しい値を書き込む必要があります。メモリインタフェースでは、必要に応じて自動的に消去動作を実行します（すべてのビットが現時点で 0 にセットされている場合、書き込む前に古い内容を削除する必要はない）。同様に、書き込むデータが 0 である場合、消去動作のみが実行されます。書き込み動作または消去動作の一方だけが実行される場合、処理時間は Tprog (3.2 ms) です。両方が実行される場合、処理時間は 2 x Tprog (6.4 ms) です。消去動作および書き込み動作の両方で FIX フラグが 1 にセットされるたびに、メモリインタフェースを強制的に実行することができます。

- **処理時間**

Tprog (3.2 ms) または 2 x Tprog (6.4 ms)。

- **オプション**

消去動作（古い内容を削除）および書き込み動作（新しいデータを書き込む）を実行するたびに、メモリインタフェースを強制的に実行するよう FIX ビットをセットします。これにより、すべてのデータ値および前回データの動作に、一定の時間を与えることができます。

データ EEPROM の消去

- **目的**

データ EEPROM から 1 行削除します。

- **サイズ**

ワードでのみ消去可能。

- **アドレス**

データ EEPROM で、有効なアドレスを 1 つ選択します。

- **プロトコル**

シングルプログラミング動作。

- **リクエスト**

PELOCK = 0、ERASE = 1（オプション DATA = 1）。

- **エラー**

PELOCK = 1 またはメモリが読み出し保護されている場合、WRPERR は 1 にセットされます。サイズがワード以外の場合、SIZERR は 1 にセットされます。

- **説明**

この動作は、データ EEPROM の行の内容を削除することを目的としています。行には、1 ワードのみが含まれます。ユーザは、正しいアドレスにワードサイズで値を書き込む必要があります。データは重要ではありません。消去のみが実行されます（ゼロ以外のデータの場合も同様）。

- **処理時間**

Tprog (3.2 ms)。

オプションバイトの書き込み

● 目的

オプションバイト領域に、特定の値を持つワードを 1 つ書き込みます。

● サイズ

ワードでのみ書き込み可能。

● アドレス

オプションバイト領域で、有効なアドレスを選択します。

● プロトコル

シングルプログラミング動作。

● リクエスト

PELOCK = 0、OPTLOCK = 0。

● エラー

PELOCK = 1 または OPTLOCK = 1 の場合、WRPERR は 1 にセットされます。

実際の読み出し保護レベルが 2 の場合、WRPERR は 1 にセットされます (オプションバイト領域をレベル 2 で書き込むことはできない)。

サイズがワード以外の場合、SIZERR は 1 にセットされます。

● 説明

この動作は、オプションバイト領域にワードを書き込むことを目的としています。オプションバイト領域は、レベル 0 または 1 でのみ書き込み可能です。

ワードでは、16 個の上位ビット (16 から 31) は 16 個の下位ビット (0 から 15) の補数である必要がある点を考慮しなければなりません。データの上位部分と下位部分に不一致がある場合、オプションバイトのローディング時にエラーが生成され ([セクション 3.8 : オプションバイト](#)を参照)、強制的にメモリーインタフェースにデフォルト値をロードします。データの補数が正しい場合、メモリーインタフェースでは書き込み時に確認を行いません。ユーザは、正しいアドレスにワードサイズで希望する値を書き込む必要があります。

データ EEPROM の場合、メモリーインタフェースは必要に応じて書き込む前に前回の内容を削除します。書き込むデータが 0 の場合、メモリーインタフェースでは不要な書き込み動作は実行しません。書き込み動作のみ、または消去動作のみが実行された場合、処理時間は Tprog (3.2 ms) です。両方実行された場合の処理時間は 2 x Tprog (6.4 ms) です。消去動作および書き込み動作の両方で FIX フラグが 1 にセットされるたびに、メモリーインタフェースを強制的に実行することができます。

いくつかの設定では保護レベルが変更されるため、さらなる注意が必要です。メモリーインタフェースでは、全体消去のオプションバイトの書き込みを変更したり、強制的に一部のビットの保護レベルを下げないようにします。詳細については、[セクション 3.4.4 : 書き込み／消去の保護管理](#)を参照してください。

● 処理時間

Tprog (3.2 ms) または 2 x Tprog (6.4 ms)。

● オプション

消去動作 (古い内容を削除) および書き込み動作 (新しいデータを書き込む) を実行するたびに、メモリーインタフェースを強制的に実行するよう FIX ビットをセットできます。これにより、すべてのデータ値および前回データをプログラムするために、一定の時間を与えることができます。

オプションバイトの消去

- **目的**
オプションバイト領域から 1 行削除します。
- **サイズ**
ワードでのみ消去可能。
- **アドレス**
オプションバイト領域で、有効なアドレスを選択します。
- **プロトコル**
シングルプログラミング動作。
- **リクエスト**
PELOCK = 0、OPTLOCK = 0、ERASE = 1 (オプション OPT = 1)。
- **エラー**
PELOCK = 1 または OPTLOCK = 1 の場合、WRPERR は 1 にセットされます。
実際の保護レベルが 2 の場合、WRPERR は 1 にセットされます (オプションバイト領域をレベル 2 で消去することはできない)。
サイズがワード以外の場合、SIZERR は 1 にセットされます。
- **説明**
この動作は、オプションバイト領域の行の内容を削除することを目的としています。行には、1 ワードのみが含まれます。データは重要ではありません。消去のみが実行されます (ゼロ以外のデータの場合も同様)。ユーザは、正しいアドレスにワードサイズで値を書き込む必要があります。
詳細については、[セクション：オプションバイトの書き込み](#)を参照してください。
消去動作後、すべてのビットが 0 にセットされるため、オプションバイトのローディング時に不一致が生じ、デフォルト値がロードされます。
- **処理時間**
Tprog (3.2 ms)。

フラッシュプログラムメモリでのシングルワードのプログラミング

- **目的**
フラッシュプログラムメモリに、特定の値を持つワードを 1 つ書き込みます。
- **サイズ**
ワードでのみ書き込み可能。
- **アドレス**
フラッシュプログラムメモリで、アドレスを 1 つ選択します。
- **プロトコル**
シングルプログラミング動作。
- **リクエスト**
PELOCK = 0、PRGLOCK = 0。
- **エラー**
PELOCK = 1 または PRGLOCK = 1 の場合、WRPERR は 1 にセットされます。
ユーザが書き込み保護されたセクタに書き込もうとした場合、WRPERR は 1 にセットされます

([PcROP \(独自仕様コード読み出し保護\)](#) セクションを参照)。

ユーザが非ゼロ値をワードに書き込もうとした場合、NOTZEROERR は 1 にセットされます。このエラーで書き込み動作が停止することはありません。これは、書き込んだアドレスを読み戻すことで、書き込んだ値とは異なる値が返されることを意味しています。

サイズがワード以外の場合、SIZERR は 1 にセットされます。

● 説明

この動作は、フラッシュプログラムメモリにワードを書き込むことを目的としています。ユーザは、正しいアドレスにワードサイズで正しい値を書き込む必要があります。メモリインタフェースでは、書き込み前に前回の内容を削除するために消去を実行することはできません。前回の内容がヌルでない場合、メモリに書き込まれた実際の値は前回の値と新しい値の論理和です (メモリインタフェースでは、前回 0 のビットに 1 を書き込む)。これは、データとデータの ECC 部分の両方に対して行われます。後でデータを読み出したときに、古いデータ、新しいデータ、またはその論理和に対応しない場合があります。ECC は、データとの互換性を失います。

● 処理時間

Tprog (3.2 ms)。

フラッシュプログラムメモリでのハーフページのプログラミング

● 目的

フラッシュプログラムメモリにハーフページ (16 ワード) を 1 つ書き込みます。

● サイズ

ワードでのみ書き込み可能。

● アドレス

最初のアドレス用に、ハーフページで整列された、フラッシュプログラムメモリのアドレスを 1 つ選択します。以降の 15 個のアドレス用に、2 番目のアドレスによって選択された同じハーフページ内にあるアドレスを 1 つ選択します。

● プロトコル

マルチプログラミング動作。

● リクエスト

PELOCK = 0、PRGLOCK = 0、FPRG = 1、PRG = 1。

● エラー

PELOCK = 1 または PRGLOCK = 1 の場合、WRPERR は 1 にセットされます。

ユーザが書き込み保護されたセクタに書き込もうとした場合、WRPERR は 1 にセットされます ([PcROP \(独自仕様コード読み出し保護\)](#) セクションを参照)。

ユーザが非ゼロ値をワードに書き込もうとした場合、NOTZEROERR は 1 にセットされます。このエラーで書き込み動作が停止することはありません。これは、書き込んだアドレスを読み戻すことで、書き込んだ値とは異なる値が返されることを意味しています。チェックは動作の始めに 16 個のアドレスすべてに対して行われ、1 つでも失敗したチェックがある場合は、すべてのチェックが終了した後でのみエラーが表示されます。

サイズがワード以外の場合、SIZERR は 1 にセットされます。

最初のアドレスがハーフページで整列されていない場合、PGAERR は 1 にセットされます。

次のいずれかのアドレス (2 から 16 のアドレス) が、最初のアドレスによって決められたハーフページに含まれない場合、PGAERR は 1 にセットされます。アドレスが増加した場合や変更された場合に、検証のためのチェックは行われません。これは、メモリインタフェースによって

自動的に行われます。重要な点は、最初のアドレスがハーフページで整列され、以降のアドレスも同じハーフページに含まれるということです。

NVM でマスタがフェッチされ、書き込みがアボートされた場合、FWWERR は 1 にセットされます。データとしての読み出しで、書き込み動作を停止しません。

- **説明**

この動作は、フラッシュプログラムメモリにハーフページを書き込むことを目的としています。ユーザは、正しいアドレスにワードサイズで希望する 16 個の値を書き込む必要があります (マルチプログラミング動作での説明を参照)。メモリインタフェースでは、書き込む前に前回の内容を削除するために、消去を実行することはできません (ユーザは書き込む前にページを削除する必要があります)。シングルプログラミング動作の場合は、書き込み値は前回および新しいデータの論理和です。ハーフページ動作が開始すると、メモリインタフェースでは 16 個のアドレス / データを待ち、フェッチ以外のすべての読み出しアクセスをアボート (バスエラーを生成) します (フェッチとプリフェッチを参照)。フェッチによってハーフページ動作は停止しますが、FLASH_SR レジスタで FWWERR エラーがセットされます。

- **処理時間**

Tprog (3.2 ms)。

フラッシュプログラムメモリでのページの消去

- **目的**

フラッシュプログラムメモリで、ページ (32 ワード) を 1 つ削除します。

- **サイズ**

ワードでのみ消去可能 (ワードサイズで書き込まれたフラッシュプログラムメモリのページを 1 つ削除)。

- **アドレス**

フラッシュプログラムメモリで、有効なアドレスを 1 つ選択します。

- **プロトコル**

シングルプログラミング動作。

- **リクエスト**

PELOCK = 0、PRGLOCK = 0、ERASE = 1、PRG = 1。

- **エラー**

PELOCK = 1 または PRGLOCK = 1 の場合、WRPERR は 1 にセットされます。

保護されたセクタに行が含まれる場合、WRPERR は 1 にセットされます ([PcROP \(独自仕様コード読み出し保護\)](#) を参照)。

サイズがワード以外の場合、SIZERR は 1 にセットされます。

- **説明**

この動作は、フラッシュプログラムメモリの行の内容を削除することを目的としています。ユーザは、正しいアドレスにワードサイズで値を書き込む必要があります。データは重要ではありません。消去のみが実行されます (ゼロ以外のデータの場合も同様)。アドレスをページで整列する必要はありません。メモリインタフェースによって、アドレスを含むページが削除されます。

- **処理時間**

Tprog (3.2 ms)。

全体消去

● 目的

フラッシュプログラムメモリとデータ EEPROM の読み出し／書き込み保護を解除します。

● サイズ

ワードでのみ消去可能。

● アドレス

全体消去を生成するには、実際のレベルをレベル 1 として、最初のオプションバイトアドレス（ビット 31 から 25、および 15 から 9 は使用されないため補数なし、チェックなし）に 0x015500AA を書き込む必要があります。

● プロトコル

シングルプログラミング動作。

● リクエスト

PELOCK = 0、OPTLOCK = 0、保護レベル = 1。データの下位ニブルは、0xAA（レベル0）、3番目のニブルが0x55でなければなりません。

● エラー

PELOCK = 1 または OPTLOCK = 1 の場合、WRPERR は 1 にセットされます。

実際の保護レベルが 2 の場合、WRPERR は 1 にセットされます（オプションバイト領域をレベル 2 で書き込むことはできない）。

サイズがワード以外の場合、SIZERR は 1 にセットされます。

● 説明

この動作は、ユーザオプションバイトへの書き込み動作と似ています。メモリインタフェースでは、実際の保護レベルが 1、要求された保護レベルが 0 の場合に、全体消去によって変更します。ユーザは、オプションバイト領域の最初のアドレスにワードサイズで希望する値を書き込む必要があります。

全体消去によって、フラッシュプログラムメモリとデータ EEPROM の内容が削除され、保護レベルがレベル 0 に変更され、PcROP (WPRMOD = 0) が無効化されます。ビットの書き込み保護と BOR_LEVEL は変更されません。

その他の動作と異なり、ソフトウェアは全体消去の進行中に新しい書き込み動作を要求することはできません。全体消去が確実に完了したことをソフトウェアで確認するには、書き込み動作を行う前に FLASH_SR レジスタの EOP ビットをリセットし、EOP がいつ 1（プログラム終了）になるかを確認します。この制限が順守されない場合、保護レベルを書き込むときにフラッシュプログラムメモリとデータ EEPROM に誤った値が書き込まれ、これによって不要な保護が追加され（不一致も同様）、デバイスを使えない状態にする可能性があります。

● 処理時間

2 x Tprog (6.4 ms) + Tglob (3.7 ms)

タイミング表

表 10. NVM 書き込み／消去タイミング

| 動作 | 最初のアドレス／データを ラッチするための遅延 (AHB クロックパルス) | 次のアドレス／データを ラッチするための遅延 (AHB クロックパルス) |
|------------------------------------|---|--|
| データ EEPROM への書き込み | 18 | -- |
| データ EEPROM の消去 | 17 | -- |
| オプションバイトの書き込み | 18 | -- |
| オプションバイトの消去 | 17 | -- |
| フラッシュプログラムメモリでの シングルワードのプログラミング | 78 | -- |
| フラッシュプログラムメモリでの ハーフページのプログラミング | 63 | 6 |
| フラッシュプログラムメモリでの ページの消去 | 76 | -- |

表 11. NVM 書き込み／消去処理時間

| 動作 | パラメータ／条件 | 処理時間 |
|--|---|-------------------------------------|
| データ EEPROM への 書き込み | 前回のデータ = 0 FIX = 0 | Tprog (3.2 ms) |
| | 前回のデータ != 0 新しいデータ = 0 サイズ = ワード FIX = 0 | Tprog (3.2 ms) |
| | その他の状況 | 2 x Tprog (6.4 ms) |
| データ EEPROM の消去 | -- | Tprog (3.2 ms) |
| オプションバイトの書き込み | 前回のデータ = 0 FIX = 0 | Tprog (3.2 ms) |
| | 前回のデータ != 0 新しいデータ = 0 FIX = 0 | Tprog (3.2 ms) |
| | その他の状況 | 2 x Tprog (6.4 ms) |
| オプションバイトの消去 | -- | Tprog (3.2 ms) |
| フラッシュプログラム メモリでのシングルワードの プログラミング | -- | Tprog (3.2 ms) |
| フラッシュプログラム メモリでのハーフページの プログラミング | -- | Tprog (3.2 ms) |
| フラッシュプログラムメモリ でのページの消去 | -- | Tprog (3.2 ms) |
| 全体消去 | | 2 x Tprog (6.4 ms) + Tglob (3.7 ms) |

ステータスレジスタ

FLASH_SR ステータスレジスタでは、メモリアインタフェース上または NVM ステータス（進行中の動作）と発生したエラーに関する情報を提供します。

BSY

このフラグはハードウェアによってセット／リセットされます。メモリアインタフェースが書き込み／消去動作を実行するたびに 1 にセットされ、他の動作は実行不可であることを通知します。新しい動作が要求された場合は、異なる動作を実行する場合があります。

- 読み出し、書き込み／消去、またはオプションローディングを待ち状態：
書き込み／消去動作の実行中（HVOFF = 0）に、ソフトウェアが書き込み動作を要求すると、メモリアインタフェースはマスタを停止し、書き込み／消去動作が完了し次第、ペンディング中の動作を実行します。
- バスエラー：
メモリアインタフェースが次のアドレス／データを待っているときに（BSY はすでに 1、ただし、HVOFF = 0）、ソフトウェアがハーフページ動作でデータの読み出しを要求した場合、メモリアインタフェースではバスエラーを生成し（読み出しを実行できないため）、欠落しているアドレスを待ち続けます。
- RDERR エラー：
書き込み／消去動作の実行中（HVOFF = 0）でアドレスが保護されているときに、ソフトウェアが読み出し動作を要求した場合、メモリアインタフェースではフラグを立てて、引き続き書き込み／消去動作が終了するのを待ちます。
- 書き込みのアボート：
メモリアインタフェースがハーフページ動作でアドレス／データを待っているときにソフトウェアが NVM でフェッチする場合、書き込み／消去動作はアボートされ、FWWERR フラグが立てられ、フェッチが実行されます。

EOP

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。ソフトウェアでは、ステータスレジスタに 1 を書き込むとリセットできます。このビットは、書き込み／消去動作が完了するとセットされ、メモリアインタフェースは他の動作（またはペンディング中の動作）を実行できるようになります。

実際の動作がいつ完了したかを知るためには、新しい書き込み／消去動作を開始する前に、これをクリアしておくことが重要です。全体消去が進行中の場合は、新しい動作を要求する前に、このフラグが立つまで待つことが重要です。

HVOFF

このフラグはハードウェアによってセット／リセットされる、NVM から入手したメモリアインタフェースの情報のコピーです。高圧レギュレータのオン (= 0) またはオフ (= 1) の状態を通知します。

PGAERR

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。アライメントエラーの発生を通知します。このフラグは、次の状況で立ちます。

- ハーフページ動作の最初のアドレスが、ハーフページに整列されていない場合（下位の 6 個のビットがゼロと等しい）
- ハーフページ動作でハーフページの変更が行われた場合（ハーフページ動作の 2 から 16 のアドレスが、最初のアドレスが選択した同じハーフページ内にない場合）

アライメントエラーによって書き込み／消去動作がアボートされ、割り込みが生成されます（FLASH_PECR レジスタで ERRIE = 1 の場合）。NVM の内容は変更されません。

このフラグをセットすると、メモリインタフェースによって、その他のすべてのハーフページ動作がブロックされます。

このフラグをリセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

SIZERR

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。サイズエラーの発生を通知します。このフラグは、次の状況で立ちます。

- フラッシュプログラムメモリおよびオプションバイトで、バイトおよびハーフワードでの書き込みが発生します。
- バイトおよびハーフワードでの消去 (FLASH_PECR レジスタでビット ERASE = 1) は、すべての領域で発生します。

サイズエラーによって書き込み／消去動作がアボートされ、割り込みが生成されます (FLASH_PECR レジスタで ERRIE = 1 の場合)。NVM の内容は変更されません。

このフラグをリセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

NOTZEROERR

このフラグは、ハードウェアによってセットされ、ソフトウェアによってリセットされます。実際の内容が 0 ではないために破壊が生じる可能性のある値をソフトウェアが NVM に書き込むと通知します。メモリインタフェースでは、書き込む前に古い内容を削除する消去動作を実行できません。

ハーフページの書き込みでは、16 ワードはすべて最初と 2 番目のアドレス／値の間でチェックされ、すべてのワードがチェックされた場合のみフラグがセットされます。フラグがセットされた場合、少なくとも 1 ワードはゼロではなく実際の値であることを意味します。

ワードの書き込みで、関係するワードのみがチェックされ、内容が非ゼロ値である場合はフラグがただちにセットされます。

非ゼロエラーでは書き込み／消去動作はアボートされませんが、FLASH_PECR レジスタで ERRIE = 1 の場合に割り込みを生成することがあります。

このフラグをリセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

3.4 メモリ保護

NVM の一部 (フラッシュプログラムメモリ、データ EEPROM、およびオプションバイト領域) を不要な書き込みやコードハッキング (不要な読み出し) から保護できます。

実装される保護は次の 3 種類です。

3.4.1 RDP (読み出し保護)

このタイプの保護は、NVM の内容を不要な読み出し (ハッキング) から保護することを目的としています。この保護は FLASH_OTPR レジスタの RDPROT ビットフィールドによって管理されています。この値は、ブート中にオプションバイト領域からロードされ、読み出し専用レジスタにコピーされます。

3 つの保護レベルが定義されています。

- レベル 0 : 保護なし
レベル 0 は、RDPROT が 0xAA の場合にセットされます。このレベルが有効で、他に有効な保護がない場合、フラッシュプログラムメモリ、データ EEPROM、およびオプションバイト領域で、制約なく読み出しおよび書き込みを行うことができます。バックアップレジスタの読み出しおよび書き込みを自由に行うことができます。



- レベル 1 : メモリ読み出し保護

レベル 1 は、レベル 0 およびレベル 2 にそれぞれ使用される 0xAA および 0xCC 以外の値が RDPROT にセットされている場合にセットされます。オプションバイト消去後、または RDPROT フィールドに不一致がある場合のデフォルトの保護レベルです。


このメモリインタフェースは、RAM におけるデバイスのブート、またはシステムメモリ、またはデバッグ機能（単線）が接続されている場合に、内部的に格納されます。この情報は、パワーダウン時（システムリセットまたはオプションローディングが不十分）、またはデバイスがレベル 2 に移った場合にのみリセットされます。フラッシュプログラムメモリとデータ EEPROM を保護します。

このレベルを有効にすると：

- デバッグ機能（単線）、または RAM におけるデバイスのブート、またはシステムメモリが接続されている場合、フラッシュプログラムメモリとデータ EEPROM へのアクセス（フェッチおよびデータの読み出しと書き込み）およびバックアップレジスタの読み出しは実行されません。フラッシュプログラムメモリとデータ EEPROM を読み出そうとすると、バスエラーが生成されます。その他の領域に対する制約はありません。オプションバイト領域の読み出しおよび書き込み／消去や、システムメモリの実行または読み出しを行うことができます。
- すべての動作は、フラッシュプログラムメモリをブートすると実行できるようになります。
- 保護レベルをレベル 0 に変更する値を最初のオプションバイトに書き込むと（バイト 0 は 0xAA、バイト 2 は 0x55 であること）、全体消去が生成されます。全体消去ではフラッシュプログラムメモリとデータ EEPROM、および最初のオプションバイトを削除し、その後レベル 0 を有効化して PCROP (WPRMOD = 0) を無効化するために再度書き込みを行い、バックアップレジスタの内容を削除します。

- レベル 2 : デバッグおよびチップ読み出し保護の無効化

レベル 2 は、RDPROT が 0xCC の場合にセットされます。このレベルを有効にすると、フラッシュプログラムメモリからのみブート可能になり、デバッグ機能（単線）は無効になります。オプションバイトは書き込み／消去から保護され、保護レベルを変更できなくなります。アプリケーションでは、フラッシュプログラムメモリとデータ EEPROM への書き込み／消去を行うことができ（フラッシュプログラムメモリからのみブート可能、また顧客コードを実行可能）、バックアップレジスタにアクセスできます。オプションバイトローディングの実行中にレベル 2 が有効である場合、RAM またはシステムメモリの古いデバッグ情報やブート情報は削除されます。

 図 6 に保護レベルを変更する方法を示します。また、表 12 にオプションバイトで読み出した値と保護レベルの関係を示します。

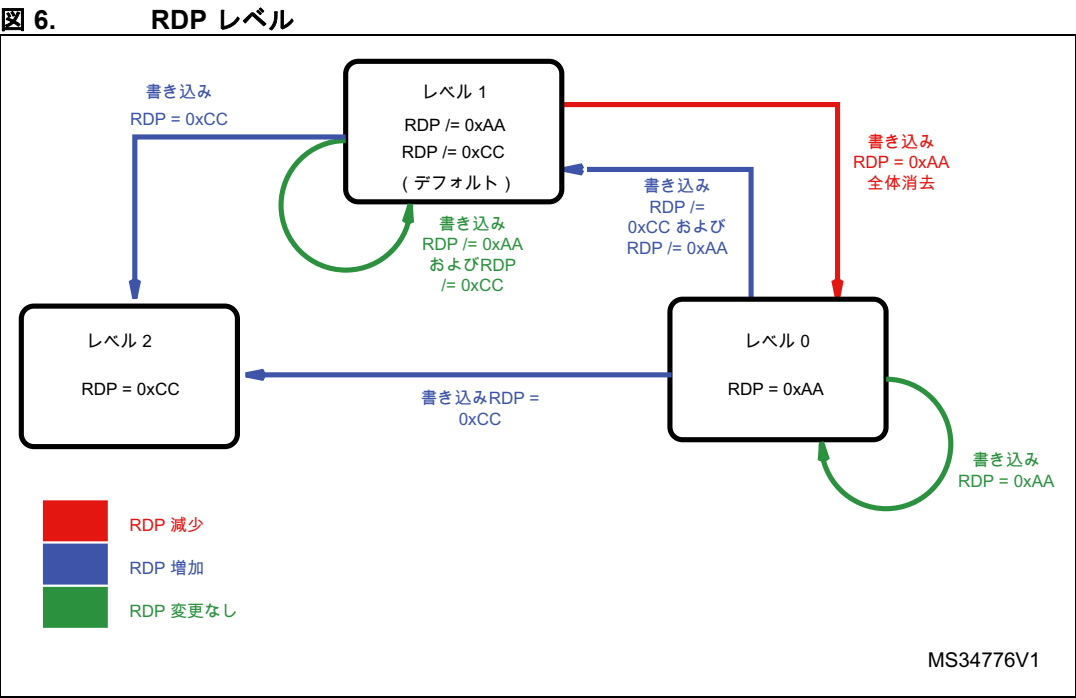


表 12. 保護レベルと RDP オプションバイトの内容

| RDP バイト値 | RDP 補数 | 読み出し保護ステータス |
|----------|----------------|-------------|
| 0xAA | 0x55 | レベル 0 |
| 0xCC | 0x33 | レベル 2 |
| その他の値 | RDP バイトの補数 | レベル 1 |
| 任意の値 | RDP バイトの補数ではない | レベル 1 |

3.4.2 PcROP (独自仕様コード読み出し保護)

フラッシュプログラムメモリを、ハッキングコードに読み出されないよう保護することができます。読み出しデータをブロックします (フェッチには適用されない)。保護されたコードは、リテラルプールを含む、保護されたゾーンにあるデータにアクセスすることはできません。

フラッシュプログラムメモリを、ハッキングコードの読み出しから保護することができます。これによって、ネイティブコードが PcROP オプションに応じてコンパイルされているとみなし、データの読み出し (フェッチを含まない) をブロックします。このモードは FLASH_OTPR レジスタで WPRMOD = 1 をセットして有効化します。

保護の粒度はセクタです (1 セクタ = 32 ページ = 4 KB)。セクタを保護するには、WRPROT 設定で該当ビットを 0 にセットします。0 は読み出し/書き込み保護を意味し、1 は保護されていないことを意味します。

表 13 に WRPROT 設定ビットと、フラッシュメモリセクタのアドレスを示します。

保護されたセクタのデータに読み出しアクセス (データとしての読み出しおよび事前読み出しを参照) が実行されることで、FLASH_SR レジスタの RDERR フラグがトリガされます。すべての読み出し保護されたセクタは書き込み保護されており、これらのいずれかのセクタへの書き込みアクセスによって、FLASH_SR レジスタの WRPERR フラグがトリガされます。

表 13. FLASH_WRPROT レジスタの保護ビットとフラッシュプログラムメモリの保護アドレスの関係

| ビット | 開始アドレス | 終了アドレス | ビット | 開始アドレス | 終了アドレス |
|-----|-------------|-------------|-----|-------------|-------------|
| 0 | 0x0800 0000 | 0x0800 0FFF | 8 | 0x0800 8000 | 0x0800 8FFF |
| 1 | 0x0800 1000 | 0x0800 1FFF | 9 | 0x0800 9000 | 0x0800 9FFF |
| 2 | 0x0800 2000 | 0x0800 2FFF | 10 | 0x0800 A000 | 0x0800 AFFF |
| 3 | 0x0800 3000 | 0x0800 3FFF | 11 | 0x0800 B000 | 0x0800 BFFF |
| 4 | 0x0800 4000 | 0x0800 4FFF | 12 | 0x0800 C000 | 0x0800 CFFF |
| 5 | 0x0800 5000 | 0x0800 5FFF | 13 | 0x0800 D000 | 0x0800 DFFF |
| 6 | 0x0800 6000 | 0x0800 6FFF | 14 | 0x0800 E000 | 0x0800 EFFF |
| 7 | 0x0800 7000 | 0x0800 7FFF | 15 | 0x0800 F000 | 0x0800 FFFF |

WPRMOD = 1 (PcROP は有効) の場合、セクタの保護を下げることはできません。新しいゼロを (新しいセクタを保護するために) セットすることはできませんが、新しい 1 を (セクタから保護を解除するために) 追加することはできません。これは、保護レベル (RDPROT 設定) にかかわらず、有効です。WPRMOD がアクティブな場合にユーザが WPRMOD をリセットまたはセクタから保護を解除しようとすると、プログラミングは起動しますが、WPRMOD や保護されたセクタは変更されません。

セクタから保護を解除する唯一の方法は、全体消去を要求することです (保護レベルが 0 に変更され、PcROP が無効化される)。PcROP の無効化により、セクタの保護を自由に変更できます。

3.4.3 不要な書き込み／消去動作に対する保護

メモリインタフェースには、不要な書き込み／消去動作に対する 2 つの保護方法が実装されています。すべてのマトリックス、またはフラッシュプログラムメモリの特定のセクタに対してのみ有効です。

[アンロック／ロック動作](#)セクションで説明しているとおり、ユーザは次のように実行できます。

- FLASH_PECR レジスタで PELOCK = 0 の場合のみ、データ EEPROM に書き込み／消去します。
- FLASH_PECR レジスタで PELOCK = 0 および OPTLOCK = 0 の場合のみ、オプションバイト領域に書き込み／消去します。
- FLASH_PECR レジスタで PELOCK = 0 および PRGLOCK = 0 の場合のみ、フラッシュプログラムメモリに書き込み／消去します。

PELOCK、PRGLOCK、および OPTLOCK をセットするシーケンスについては、[データ EEPROM および FLASH_PECR レジスタのアンロック、フラッシュプログラムメモリのアンロック、および オプションバイト領域のアンロック](#)セクションを参照してください。

フラッシュプログラムメモリでは、セクタの粒度によって別の書き込み保護を追加できます。PcROP を無効にすると (WPRMODE = 0)、WRPROT のビットを使用して、セクタの書き込み保護が有効化されます。極性は相対的に PcROP に反します。セクタを保護するには、ビットを 1 にセットする必要があります。保護を解除するには、ビットを 0 にセットする必要があります。[表 13](#) も、書き込み保護に有効です。説明したとおり、PcROP を有効化すると読み出しに対して保護されていたセクタは書き込み／消去に対しても保護されます。レベル 0 およびレベル 1 の両方で、いつでもセクタの書き込み保護を変更することができます (オプションバイトが書き込み／消去可能である場合、また PcROP が無効である場合)。

[表 14](#) に保護について示します。

表 14. メモリアクセスとモード、保護、およびフラッシュプログラムメモリセクタ

| フラッシュプログラムメモリ セクタ | モード | | | | |
|--------------------------------------|---|-------|-------|---|-------------------|
| | ユーザ (アプリケーション内 プログラミングを含む) デバッグなし、または RAM でのブートなし、または システムメモリでのブートなし | | | ユーザ デバッグ時、または RAM でのブートあり、または システムメモリでのブートあり | |
| RDP | レベル 1 レベル 0 | レベル 2 | レベル 0 | レベル 1 | レベル 2 |
| フラッシュプログラムメモリ (FLASH_PRGLOCK = 1) | R | R | R | 保護 (アクセスなし) | NA ⁽¹⁾ |
| フラッシュメモリ (FLASH_PRLOCK = 0) | R / W | R / W | R / W | 保護 (アクセスなし) | NA ⁽¹⁾ |
| フラッシュプログラムメモリ (WRP ページ内) | R | R | R | 保護 (アクセスなし) | NA ⁽¹⁾ |
| フラッシュプログラムメモリ (PCROP ページ内) | フェッチ | フェッチ | フェッチ | 保護 (アクセスなし) | NA ⁽¹⁾ |
| データ EEPROM (FLASH_PELock = 1) | R | R | R | 保護 (アクセスなし) | NA ⁽¹⁾ |
| データ EEPROM (FLASH_PELock = 0) | R / W | R / W | R / W | 保護 (アクセスなし) | NA ⁽¹⁾ |
| オプションバイト (FLASH_OPTLOCK = 1) | R | R | R | R | NA ⁽¹⁾ |
| オプションバイト (FLASH_OPTLOCK = 0) | R / W | R | R / W | R / W | NA ⁽¹⁾ |

1. NA は「Not Applicable (該当なし)」の略。

3.4.4 書き込み／消去の保護管理

ここでは、前回の保護をすべて変更するための規則の概要を示します。

- 保護レベルが 2 の場合は、保護は変更できません。
- レベル 0 または 1 の場合は、最初のオプションバイトワードに `xx33xxCC` (x は任意の値を持つ 16 進の数字) を書き込むことで、いつでもレベル 2 に移ることができます。
- レベル 0 の場合は、最初のオプションバイトワードに `xx33xxCC` (レベル 2) または `xx55xxAA` (レベル 0) 以外を書き込むことで、レベル 1 に移ることができます。
- レベル 1 の場合は、最初のオプションバイトワードに `xx55xxAA` を書き込むことで、保護をレベル 0 に下げることができます。これにより、全体消去が生成され、PcROP フィールドも削除されます。
- PcROP は、最初のオプションバイトワードに `x0xx1xx` を書き込むことで、いつでも有効にすることができます (レベル 2 以外)。このフラグで、オプションバイトローディング時に不一致がある場合、PcROP が有効化されます。
- PcROP は全体消去 (レベル 1 からレベル 0 への移動) の要求時に削除できます。
- PcROP が無効化されると、オプションバイトの 3 番目のワードで、書き込み保護をセクタに追加するか (1 を書き込む)、削除する (0 を書き込む) ことができます。不一致は、すべての書き込み保護されたセクタに関わります (PcROP が無効の場合)。
- PcROP が有効な場合、保護されたセクタを追加できますが (0 を書き込む)、削除することはできません。不一致は、すべての読み出し／書き込み保護されたセクタに関わります (PcROP が有効の場合)。
- 全体消去によってオプションバイトの 3 番目のワードが削除されることはありません。ユーザは正確に書き込む必要があります。

3.4.5 保護エラー

書き込み保護エラーフラグ (WRPERR)

フラッシュプログラムメモリとデータ EEPROM の書き込み保護されているページで消去／プログラム動作を開始すると、FLASH_SR レジスタに書き込み保護エラーフラグ (WRPERR) がセットされます。

結果として、ソフトウェアが次の動作を実行しようとする、WRPERR フラグがセットされます。

- WRP ページへの書き込み。
- システムメモリページまたは工場オプションバイトへの書き込み。
- フラッシュプログラムメモリ、データ EEPROM、またはオプションバイトへの書き込み (PEKEY、PRGKEY、または OPTKEY によってアンロックされていない場合)
- フラッシュプログラムメモリ、データ EEPROM、またはオプションバイトへの書き込み (RDP オプションバイトがセットされていて、デバイスがデバッグモードになっているか、RAM またはシステムメモリからブートされる場合)

書き込み保護エラーによって書き込み／消去動作がアボートされ、割り込みが生成されます (FLASH_PECR レジスタで ERRIE = 1 の場合)。

このフラグをリセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

読み出しエラー (RDERR)

ソフトウェアが、PcROP で保護されたセクタを読み出そうとすると、FLASH_SR レジスタの RDERR フラグが立ちます。バスで受信したデータは 0 です。

エラー割り込みが有効である場合 (FLASH_PECR レジスタで ERRIE = 1)、割り込みが生成されます。

このフラグをリセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

3.5 NVM 割り込み

FLASH_PECR レジスタのプログラム終了割り込み有効化ビット (EOPIE) をセットすると、消去またはプログラミング動作が正常に終了したときの割り込み生成を有効化します。この場合、FLASH_SR レジスタのプログラム終了 (EOP) ビットがセットされます。リセットするには、ソフトウェアではこれを 1 に書き込む必要があります。

FLASH_PECR レジスタのエラー割り込み有効化ビット (ERRIE) をセットすると、プログラミングまたは消去動作の要求時にエラーが発生したときの割り込み生成を有効化します。この場合、FLASH_SR レジスタに 1 つまたは複数のエラーフラグがセットされます。

- RDERR (PCROP 読み出し保護エラーフラグ)
- WRPERR (書き込み保護エラーフラグ)
- PGAERR (プログラム配置エラーフラグ)
- OPTVERR (オプション有効性エラーフラグ)
- SIZERR (サイズエラーフラグ)
- FWWERR (書き込み中フェッチエラーフラグ)
- NOTZEROERR (非ゼロワード値書き込みエラーフラグ)

このエラーフラグをリセットするには、ソフトウェアでは該当のフラグを 1 に書き込む必要があります。

表 15. フラッシュ割り込みリクエスト

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|----------|--|------------|
| 操作終了 | EOP | EOPIE |
| 誤差 | RDERR WRPERR PGAERR OPTVERR SIZERR FWWERR NOTZEROERR | ERRIE |

3.5.1 バスエラー (ハードフォールト)

バスエラーは、次のように生成されます。

- RDP のセット時に読み出しアクセスを試みた場合に、メモリバスに生成されます。
- データとしての読み出しを受信した場合に、メモリバスに生成されます。その後、メモリインタフェースはハーフページ書き込み中にデータ/アドレスを待ちます (最初のアドレス後、16 番目のアドレス前)。
- PEKEYR、PRGKEYR、または OPTKEYR に誤った値が書き込まれた場合に、レジスタバスに生成されます。

3.6 メモリインタフェース管理

このセクションでは、1 つの動作の進行中に別の動作を要求した場合に何が起こるか明らかにすることを目的としています。どのように異なる動作が協調して動作し、メモリインタフェースによって管理されるかについて示します。

3.6.1 動作の優先順位と展開

動作には 3 つのタイプがあり、それぞれに異なる流れがあります。

読み出し

- 進行中の動作がない場合で読み出しアドレスが保護されていない場合、読み出しは遅延なく実際の設定で実行されます。
- 読み出しアドレスが保護されている場合、動作はフィルタリングされ（要求された読み出しがいつまでもメモリに送信されない）、エラーが発生します。
- 読み出しアドレスが保護されておらず、メモリインタフェースがビジーで動作を実行できない場合、読み出しはホールド状態になり、可能になり次第実行されます。

書き込み／消去

- 進行中の動作がなく、書き込みアドレスが保護されていない場合、書き込み／消去はただちに開始します。バスとマスタがブロックされるいくつかのクロックパルス後（表 10 を参照）、メモリインタフェースは動作を続行し、バスとマスタも解放されます。
- アドレスが保護されている場合、書き込み／消去はフィルタリングされ（要求された書き込み／消去がいつまでもメモリに送信されない）、エラーが発生します。
- アドレスが保護されておらず、1 つまたは複数の条件を満たしている場合、動作はアボートされ（アボートには、NVM および データ EEPROM をデフォルトの設定に戻す必要があるため、実行するまで時間がかかる）、エラーが発生します。
- 書き込み／消去アドレスが保護されておらず、すべての規則を順守している場合、またメモリインタフェースがビジーである場合、動作はホールド状態になり、可能になり次第実行されます。

オプションバイトローディング

- 書き込み／消去が進行中の場合、オプションバイトローディングは動作が終了するまで待つて実行されます。待ち状態であっても、その他の書き込み／消去は受け付けません。
- 進行中の書き込み／消去がない場合、オプションバイトは直接実行されます（読み出し動作は、オプションバイトの要求の結果、システムリセットが 0 になるまで実行される）。

これは、オプションバイトローディングの優先順位が読み出し動作や書き込み／消去動作よりも高いことを意味しています。その他の動作はすべて要求順に実行されます。

3.6.2 動作シーケンス

書き込み中のデータとしての読み出し

マスタが書き込み動作の進行中にデータとして読み出しを要求する場合には（[データとしての読み出しおよび事前読み出し](#)を参照）、3つの異なるケースがあります。

1. 読み出しが保護された領域にある場合、RDERR フラグが立てられ、書き込み動作が続行します。
2. 書き込み動作で [シングルプログラミング動作](#)または [マルチプログラミング動作（ハーフページ）](#)を使用する場合で、アドレス／データがすべてメモリインタフェースに送信されている場合、読み出しはホールド状態になり、書き込み動作が完了し次第、実行されます。特に強調すべき点は、読み出しが実行を待つ間、マスタはブロックされ、書き込みおよび読み出し動作が完了するまで他の動作を実行することができないということです。
3. 書き込み動作で [マルチプログラミング動作（ハーフページ）](#)を使用する場合で、すべてのアドレス／データがメモリインタフェースに送信されていない場合、読み出しは受け付けられず、バスエラーが生成され、メモリインタフェースでは書き込み動作を完了するために欠落しているアドレス／データを待ち続けます。

書き込み中にフェッチ

書き込みの進行中にマスタが命令をフェッチする場合、この状況はデータとしての読み出しに似ていますが（[1.](#) および [2.](#) を参照）、後者の場合は以下のとおりです。

- 書き込み動作で [マルチプログラミング動作（ハーフページ）](#)を使用する場合で、すべてのアドレス／データがメモリインタフェースに送信されていない場合、書き込みはアボートされ、初めからなかったかのように扱われます。読み出しが受け付けられ、値がマスタに送信されます。

他の書き込み動作の進行中に書き込み

マスタが別の書き込み動作の進行中に書き込み動作を要求する場合には、さまざまなケースがあります。

- 前回の書き込み動作で [シングルプログラミング動作](#)または [マルチプログラミング動作（ハーフページ）](#)を使用し、アドレス／データがすべてメモリインタフェースに送信されている場合、また新しい書き込みが保護された領域内にある場合、WRPERR フラグが立てられ、前回の書き込みが続行し、新しい書き込みが削除されます。
- 前回の書き込み動作で [シングルプログラミング動作](#)または [マルチプログラミング動作（ハーフページ）](#)を使用し、アドレス／データがすべてメモリインタフェースに送信されている場合、また新しい [シングルプログラミング動作](#)または [マルチプログラミング動作（ハーフページ）](#)が保護された領域内にない場合、新しい書き込みはホールド状態となり、最初の書き込み動作が完了し次第、実行されます。特に強調すべき点は、2番目の書き込みを要求したマスタが、最初の書き込みが完了し、2番目が内部的にアドレスとデータを格納するまでブロックされるということです。
- 全体消去の進行中に、新しい書き込みを要求することは禁止されています。全体消去の全段階においてデータは内部的に格納されていないため、格納された保護値が新しいデータによって変更され、不要な保護が追加される可能性があります。
- 最初の動作で [シングルプログラミング動作](#)または [マルチプログラミング動作（ハーフページ）](#)が使用されており、すべてのアドレス／データがメモリインタフェースに送信されている場合は、新しい書き込み動作を準備するために設定を変更することができます。

3.6.3 読み出し中にウェイトステート数を変更

ウェイトステート数を変更するには、FLASH_ACR レジスタに書き込む必要があります。レジスタの読み出し／書き込みでは、メモリの読み出し／書き込みとは異なるインタフェースを使用します。メモリインタフェースの読み出し中にウェイトステート数を変更することはできず、レジスタインタフェースに要求が送信された場合にメモリインタフェースを停止することはできません。このため、マスタがメモリを読み出しているときに別のマスタがウェイトステート数を変更した場合、レジスタインタフェースは変更が有効になるまで（読み出しが停止するまで）ロックされます。ウェイトステート数を変更するマスタを停止するには、FLASH_ACR レジスタの内容を読み戻すことが重要です。ウェイトステート数の変更に必要なクロックのサイクル数は、顧客コードに依存するため、知ることはできません。

3.6.4 パワーダウン

NVM をパワーダウン状態にするには、アンロックシーケンスを実行する必要があります。

次のシーケンスは、FLASH_ACR レジスタの RUN_PD ビットをアンロックするために使用します。

- FLASH_PDKEYR レジスタに PDKEY1 = 0x04152637 を書き込みます。
- FLASH_PDKEYR レジスタに PEKEY2 = 0xFAFBFCFD を書き込みます。

2 つのキーを、他の読み出しや書き込みの制約なしに書き込む必要があります。誤ったキーを使用してもエラーは生成されません。両方の書き込み後、RUN_PD ビットはアンロックされて 1 に書き込むことができ、NVM はパワーダウンモードになります。

RUN_PD フラグを 0 にリセットすると（NVM が使用可能な状態になる）自動的にシーケンスがリセットされ、RUN_PD を再度有効化するために 2 つのキーが要求されます。

3.7 フラッシュレジスタの説明

レジスタの読み出し

メモリーインタフェースの内部レジスタをすべて読み出すには、ユーザーはレジスタアドレスで読み出さなければなりません。内容はただちに使用可能です（レジスタの読み出しにウェイトステートは不要）。ウェイトステート数の修正後にユーザが FLASH_ACR レジスタの読み出しを試みた場合、内容は変更が有効になり次第使用可能になります（NVM メモリで読み出しが行われない場合、ウェイトステート数が変更される）。

レジスタが選択されていない場合、または誤ったアドレスがメモリーインタフェースに送信された場合、応答としてゼロ値が送信されます。エラーは生成されません。

マスタが 8 個または 16 個のビットを読み出す要求を送信した場合、メモリーインタフェースではデータ出力バス上のレジスタの対応する部分を返します。たとえば、レジスタの内容が 0x12345678 で、マスタが 2 番目のバイトを読み出す要求を送信した場合、出力は 0x34343434 となります（ゼロからバイトをカウントし始めた場合、2 番目のレジスタバイトの内容は 0x34 であるため）。同様に、前回のレジスタのハーフワードゼロを読み出す要求をマスタが送信した場合、出力は 0x56785678 となります。

レジスタへの書き込み

メモリーインタフェースの設定レジスタには、2 つのタイプのビットがあります。

- 直接書き込めるビット
- アンロックする際に特殊なシーケンスが必要なビット

どちらのカテゴリにビットが属しているかを判断するには、次のセクションで各ビットについて詳細に説明しているので参照してください。

レジスタまたはキーレジスタに直接書き込める場合、ユーザはレジスタアドレスに期待値を書き込む必要があります。アドレスが正しくない場合にエラーは生成されません。ユーザが読み出し専用のレジスタを修正しようとしても、エラーは生成されず、修正動作による影響もありません。レジスタを、バイト、ハーフワード、およびワードで書き込むことができます。

アンロックシーケンスが必要な場合は、使用する正しい値が与えられます。

3.7.1 アクセス制御レジスタ (FLASH_ACR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|----------|-----------|--------|----------|------|--------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PRE_READ | DISAB_BUF | RUN_PD | SLEEP_PD | Res. | PRFTEN | LATENCY |
| | | | | | | | | | r/w | r/w | r/w | r/w | | r/w | r/w |

ビット 31:7 予約済み

ビット 6 **PRE_READ**

このビットは、事前読み出しを有効にします。

0 : 事前読み出しを無効にします。

1 : 事前読み出しを有効にします。メモリインタフェースでは、読み出した最後のアドレスをデータとして格納し、他に進行中の読み出しまたは書き込みまたはプリフェッチ動作がない場合に、次のアドレスの読み出しを試みます。

注： これは、DISAB_BUF ビット (このレジスタに含まれる) が 1 にセットされるたびに自動的にリセットされます。

ビット 5 **DISAB_BUF**

このビットは、読み出し中にキャッシュとして使用されるバッファを無効にします。これは、すでに読み出されたアドレス (たとえば、前回のアドレス) を含むすべての読み出しが、NVM にアクセスすることを意味しています。このビットがリセットされると、PRFTEN および PRE_READ ビットも自動的にリセットされます。

0 : バッファを有効にします。

1 : バッファを無効にします。NVM 値が 1 つ必要になるたびに、1 つの新しいメモリ読み出しシーケンスを行う必要があります。

ビット 4 **RUN_PD**

このビットは、デバイスが実行モードである場合に、NVM をパワーダウンモードにするか、アイドルモードにするかを決定します。FLASH_PDKEYR レジスタにアンロックされた書き込みがある場合にのみ、このビットを書き込むことができます。

正しいシーケンスについては、[セクション 3.6.4 : パワーダウン](#)で説明しています。このビットに 0 を書き込む場合、キーは自動的に失われ、1 を再度書き込む場合は新しいアンロックシーケンスが必要です。

0 : デバイスが実行モードのとき、NVM はアイドルモードです。

1 : デバイスが実行モードのとき、NVM はパワーダウンモードです。

ビット 3 **SLEEP_PD**

このビットは、デバイスが SLEEP モードである場合に、フラッシュプログラムメモリとデータ EEPROM をパワーダウンモードか、アイドルモードにすることができます。

0 : デバイスが SLEEP モードのとき、NVM はアイドルモードです。

1 : デバイスが SLEEP モードのとき、NVM はパワーダウンモードです。

ビット 2 予約済み

ビット 1 **PRFTEN**

このビットは、プリフェッチを有効にします。これは、DISAB_BUF ビット（このレジスタに含まれる）が 1 にセットされるたびに自動的にリセットされます。プリフェッチの動作方法については、[フェッチとプリフェッチ](#)セクションを参照してください。

0：プリフェッチを無効にします。

1：プリフェッチを有効にします。メモリインタフェースでは、フェッチした最後のアドレスを格納し、他に進行中の読み出しまたは書き込み動作がない場合に、次のアドレスの読み出しを試みます。

ビット 0 **LATENCY**

このビットの値によって、NVM の読み出しに必要なウェイトステートが 0 であるか 1 であるかを指定します。ユーザは、コア周波数および動作モード（電源）に対し、正しい値を書き込む必要があります。使用する正しい値については、[表 6](#) を参照してください。設定が正しいかどうかを検証するための確認は行われません。

クロック周波数を増加するには、ユーザは、このビットを「1」に変更してから周波数を増加する必要があります。クロック周波数を低減するには、周波数を下げてからこのビットを「0」に変更します。

0：NVM にワードを読み出すために、0 ウェイトステートを使用します。

1：NVM にワードを読み出すために、1 ウェイトステートを使用します。

3.7.2 プログラミングと消去制御レジスタ (FLASH_PECR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0007

このレジスタは、FLASH_PKEYR に適切な書き込みシーケンスが行われて PELOCK ビットがリセットされた場合にのみ書き込まれます。

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|------|------|------|------|-------|------|------|------|------|------|------|------------|----------|---------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OBL_LAUNCH | ERRIE | EOPIE |
| | | | | | | | | | | | | | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | FPRG | ERASE | FIX | Res. | Res. | Res. | DATA | PROG | OPT_LOCK | PRG_LOCK | PE_LOCK |
| | | | | | rw | rw | rw | | | | rw | rw | rs | rs | rs |

ビット 31:19 予約済み

ビット 18 **OBL_LAUNCH**

このビットをセットすると、ソフトウェアからオプションバイトの再ロードを要求されます。オプションバイトの再ロードによって進行中の修正動作が停止されることはありませんが、新しい動作はブロックされます。オプションバイトの再ロードによって、システムリセットが生成されます。

0 : オプションバイトローディングが完了します。

1 : オプションバイトローディングを行います。

注 : このビットは、OPTLOCK が 0 である場合にのみ修正できます。OPTLOCK (またはその他のロックビット) をロックしても、このビットはリセットされません。

ビット 17 **ERRIE** : エラー割り込みイネーブル

0 : エラー割り込みは無効です。

1 : エラー割り込みは有効です。

注 : このビットは、PELOCK が 0 である場合にのみ修正できます。PELOCK をロックしてもこのビットはリセットされず、割り込みは有効なままです。

ビット 16 **EOPIE** : プログラミング終了割り込みイネーブル

0 : プログラミング終了割り込みは無効です。

1 : プログラミング終了割り込みは有効です。

注 : このビットは、PELOCK が 0 である場合にのみ修正できます。PELOCK をロックしてもこのビットはリセットされず、割り込みは有効なままです。

ビット 15:11 予約済み

ビット 10 **FPRG** : ハーフページプログラミングモード

0 : ハーフページプログラミングモードは無効です。

1 : ハーフページプログラミングモードは有効です。

注 : このビットは、PELOCK が 0 である場合に修正できます。PELOCK がセットされるとリセットされます。

ビット 9 **ERASE**

0 : 消去動作は要求されません。

1 : 消去動作が要求されます。

注 : このビットは、PELOCK が 0 である場合に修正できます。PELOCK がセットされるとリセットされます。

ビット 8 FIX

0 : 消去フェーズは、データ EEPROM とオプションバイト領域でのプログラム動作の前に、必要に応じて自動的に実行されます。プログラミング時間は、Tprog (プログラム動作) または 2 * Tprog (消去 + プログラム動作) です。

1 : プログラム動作は常に予備消去とともに実行され、プログラミング時間は 2 * Tprog です。

注： このビットは、PELOCK が 0 である場合に修正できます。PELOCK がセットされるとリセットされます。

ビット 5:7 予約済み

ビット 4 DATA

0 : データ EEPROM は選択されません。

1 : データメモリが選択されます。

注： このビットは、PELOCK が 0 である場合に修正できます。PELOCK がセットされるとリセットされます。ページとワードがデータ EEPROM で同じサイズを持つため、このビットはあまり有用ではありませんが、ワード動作から消去動作 (ページごと) を識別するために使用されます。

ビット 3 PROG

このビットは、ハーフページプログラム動作と、フラッシュプログラムメモリのページ消去動作に使用されます。

0 : フラッシュプログラムメモリは選択されません。

1 : フラッシュプログラムメモリが選択されます。

注： このビットは、PELOCK が 0 である場合に修正できます。PELOCK がセットされるとリセットされます。

ビット 2 OPTLOCK : オプションバイトロック

このビットでは、ユーザのオプションバイト領域と OBL_LAUNCH ビット (このレジスタに含まれる) への書き込み/消去動作をブロックします。再度ロックするには、1 を書き込む必要があります。0 にリセットするには、PELOCK ビットを 0 にセットし、OPTKEYR レジスタを使用して正しいアンロックシーケンスを行う必要があります (オプションバイト領域のアンロックを参照)。シーケンスが正しくない場合、ビットは次のシステムリセットまでロックされ、バスエラーが 1 つ生成されます。PELOCK = 1 のときにシーケンスが実行されると、ビットはロックされたままとなり、バスエラーも生成されません。アンロックに必要なキーは次のとおりです。

– 1 つ目のキー : 0xFBEAD9C8

– 2 つ目のキー : 0x24252627

0 : オプションバイト領域での書き込み/消去動作は無効です。

1 : オプションバイト領域での書き込み/消去動作は有効です。

注： このビットは、PELOCK がセットされるとセットされます。

ビット 1 PRGLOCK : プログラムメモリのロック

このビットでは、フラッシュプログラムメモリへの書き込み/消去動作をブロックします。再度ロックするには、1 を書き込む必要があります。0 にリセットするには、PELOCK ビットを 0 にセットし、PRGKEYR レジスタを使用して正しいアンロックシーケンスを行う必要があります (フラッシュプログラムメモリのアンロックを参照)。シーケンスが正しくない場合、ビットは次のシステムリセットまでロックされ、バスエラーが 1 つ生成されます。PELOCK = 1 のときにシーケンスが実行されると、ビットはロックされたままとなり、バスエラーも生成されません。アンロックに必要なキーは次のとおりです。

– 1 つ目のキー : 0x8C9DAEBF

– 2 つ目のキー : 0x13141516

0 : フラッシュプログラムメモリでの書き込み/消去動作は無効です。

1 : フラッシュプログラムメモリでの書き込み/消去動作は有効です。

注： このビットは、PELOCK がセットされるとセットされます。

ビット 0 **PELOCK** : FLASH_PECR ロック

このビットでは、FLASH_PECR をロックします。再度ロックするには、1 を書き込む必要があります。0 にリセットするには、PEKEYR レジスタを使用して正しいアンロックシーケンスを行う必要があります (データ EEPROM および FLASH_PECR レジスタのアンロックを参照)。シーケンスが正しくない場合、ビットは次のシステムリセットまでロックされ、バスエラーが 1 つ生成されます。アンロックに必要なキーは次のとおりです。

- 1 つ目のキー : 0x89ABCDEF
- 2 つ目のキー : 0x02030405

0 : FLASH_PECR レジスタをアンロックします。修正して他のビットをアンロックすることもできます。データの書き込み/消去動作は有効です。

1 : FLASH_PECR レジスタをロックします。書き込み/消去動作を開始することはできません。

3.7.3 パワーダウンキーレジスタ (FLASH_PDKEYR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

ビット31:0 書き込み専用レジスタです。2つの書き込み動作（1つ目は 0x04152637、2つ目は 0xFAFBFCFD）のシーケンスによって、書き込みサイズがワードと同じ場合、FLASH_ACR レジスタの RUN_PD ビットをアンロックすることができます。詳細については、[セクション 3.6.4: パワーダウン](#)を参照してください。

3.7.4 PECKR アンロックキーレジスタ (FLASH_PEKEYR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

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

ビット31:0 書き込み専用レジスタです。2つの書き込み動作（1つ目は 0x89ABCDEF、2つ目は 0x02030405）のシーケンスによって、書き込みサイズがワードと同じ場合、FLASH_PCKR レジスタをアンロックすることができます。詳細については、[データEEPROM およびFLASH_PCKR レジスタのアンロック](#)を参照してください。

3.7.5 キーレジスタのプログラミングと消去 (FLASH_PRGKEYR)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

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

ビット31:0 書き込み専用レジスタです。2つの書き込み動作（1つ目は 0x8C9DAEBF、2つ目は 0x13141516）のシーケンスによって、書き込みサイズがワードと同じ場合、フラッシュプログラムメモリをアンロックすることができます。このシーケンスは、PELOCK がすでにアンロックされている場合にのみ実行できます。詳細については、[フラッシュプログラムメモリのアンロック](#)を参照してください。



3.7.6 オプションバイトアンロックキーレジスタ (FLASH_OPTKEYR)

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

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

ビット31:0 書き込み専用レジスタです。2つの書き込み動作（1つ目は 0xFBEAD9C8、2つ目は 0x24252627）のシーケンスによって、書き込みサイズがワードと同じ場合、オプションバイト領域と OBL_LAUNCH ビットをアンロックすることができます。このシーケンスは、PELOCK がすでにアンロックされている場合にのみ実行できます。詳細については、[オプションバイト領域のアンロック](#)を参照してください。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

3.7.7 ステータスレジスタ (FLASH_SR)

アドレスオフセット : 0x018

リセット値 : 0x0000 000C

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|-------|------|---------|--------|--------|--------|------|------|------|------|-------|-------|--------|------------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FWWERR | NOTZEROERR |
| | | | | | | | | | | | | | | rc_w1 | rc_w1 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | RDERR | Res. | OPTVERR | SIZERR | PGAERR | WRPERR | Res. | Res. | Res. | Res. | READY | HWOFF | EOP | BSY |
| | | rc_w1 | | rc_w1 | rc_w1 | rc_w1 | rc_w1 | | | | | r | r | rc_w1 | r |

ビット 31:18 予約済み

ビット 17 FWWERR

このビットは、フェッチを実行するために書き込み／消去動作がアボートされた場合に、ハードウェアによってセットされます。これは実際のエラーではありませんが、書き込み／消去動作が実行されなかったことを通知するために使用されます。このフラグをリセットするには、1 を書き込みます。

0 : フェッチを実行するために書き込み／消去動作をアボートすることはありません。

1 : フェッチを実行するために書き込み／消去動作をアボートします。

ビット 16 NOTZEROERR

このビットは、フラッシュプログラムまたはシステムメモリのプログラムが非ゼロ領域に上書きしようとした場合に、ハードウェアによってセットされます。このフラグでプログラム動作が停止することはありません。読み戻しの際に検出した値は、ユーザが書いたものではない可能性があります。このフラグをリセットするには、1 を書き込みます。

0 : 書き込み動作を消去された領域で行います。もしくは書き込みを行う前にメモリインタフェースで消去を適用することもできます。

1 : 書き込み動作を消去されていない領域で行います。また、書き込みを行う前にメモリインタフェースで消去を適用することはできません。

ビット 15:14 予約済み

ビット 13 RDERR

このビットは、ユーザが PcROP で保護された領域を読み出そうとする場合に、ハードウェアによってセットされます。1 を書き込むことによってクリアされます。

0 : 読み出し保護エラーは発生しません。

1 : 読み出し保護エラーが 1 つ発生します。

ビット 12 予約済み

ビット 11 OPTVERR : オプション有効性エラー

このビットは、オプションバイトローディング中、1 つ以上の設定に不一致があった場合に、ハードウェアによってセットされます。これは、ロードした設定が、ユーザがメモリに書き込んだ設定と異なる可能性があることを意味します。1 を書き込むことによってクリアされます。

保護 (WPRMOD、RDPROT、WRPROT) のロード中にエラーが発生した場合、フラッシュプログラムメモリのソースコードが正しく実行されない場合があります。

0 : オプションバイトローディング中にエラーが発生しなかった場合

1 : オプションバイトローディング中に 1 つ以上のエラーが発生した場合

ビット 10 SIZERR : サイズエラー

このビットは、プログラムするデータのサイズが正しくない場合に、ハードウェアによってセットされます。1 を書き込むことによってクリアされます。

- 0 : サイズエラーは発生しません。
- 1 : サイズエラーが 1 つ発生します。

ビット 9 PGAERR : プログラミング配置エラー

このビットは、ハーフページ動作の最初のワードがハーフページに配置されていない、またはハーフページ動作の次のうちの 1 ワードが最初のワードと同じハーフページにないなど、アライメントエラーの発生時にハードウェアによってセットされます。このビットをセットすると、1 を書き込む前にクリアする必要があります、またハーフページ動作を受け付けることはできません。

- 0 : アライメントエラーは発生しません。
- 1 : アライメントエラーが 1 つ発生します。

ビット 8 WRPERR : 書き込み保護エラー

このビットは、プログラムまたは消去するアドレスが書き込み保護されている場合に、ハードウェアによってセットされます。1 を書き込むことによってクリアされます。

- 0 : 保護エラーは発生しません。
- 1 : 保護エラーが 1 つ発生します。

ビット 7:4 予約済み**ビット 3 READY**

このビットをセットすると、NVM の読み出しおよび書き込み／消去動作がレディ状態になります。

- 0 : NVM はレディ状態ではありません。読み出しおよび書き込み／消去動作を行うことができません。
- 1 : NVM はレディ状態ではありません。

ビット 2 HVOFF

このビットは、ハードウェアによってセット／リセットされます。

- 0 : 高電圧で NVM での書き込み／消去動作を実行します。
- 1 : 高電圧はオフです。進行中の NVM での書き込み／消去動作はありません。

ビット 1 EOP : プログラムの終了

このビットは、書き込み／消去動作がアボートされていない場合に、書き込み／消去動作の終了時にハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってリセットされます。

- 0 : EOP 動作は発生しません。
- 1 : EOP イベントが発生しました。EOPIE ビットがセットされると、割り込みが生成されます。

ビット 0 BSY: メモリインタフェースビジー

書き込み／消去動作が進行中です。

- 0 : 進行中の書き込み／消去動作はありません。
- 1 : 書き込み／消去動作が進行中です。

3.7.8 オプションバイトレジスタ (FLASH_OTPR)

アドレスオフセット : 0x1C

リセット値 : 0xX0XX 0XXX

| | | | | | | | | | | | | | | | |
|-------|------|------|------|------|------|------|--------|--------|------------|-----------|--------|--------------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| BOOT1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | nRST_STDBY | nRST_STOP | WDG_SW | BOR_LEV[3:0] | | | |
| r | | | | | | | | | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | WPRMOD | RDPROT | | | | | | | |
| | | | | | | | r | r | r | r | r | r | r | r | r |

ビット 31 **BOOT1**

入力パッド Boot0 とともに、このビットではブートソースを選択します。

- BOOT0 = 0 および BOOT1 = X の場合、ブートはフラッシュプログラムメモリにあります。
- BOOT0 = 1 および BOOT1 = 0 の場合、ブートは RAM にあります。
- BOOT0 = 1 および BOOT1 = 1 の場合、ブートはシステムメモリにあります。

ビットは読み出し専用です。ブートソースを変更するには、オプションバイトを再ロードする必要があります。オプションバイトローディング中に、この設定に不一致がある場合、1 がロードされます。

デバイスがレベル 2 で保護されている場合、Boot0 および Boot1 はそれぞれ意味を失います。ブートはフラッシュプログラムメモリに常に強制的に設定されます。

ビット 30:23 予約済み

ビット 22 **nRST_STDBY**

このビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、1 がロードされます。

- 0 : STANDBY モードに移行するときにリセットを生成します。
- 1 : リセットは生成されません。

ビット 21 **nRST_STOP**

このビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、1 がロードされます。

- 0 : STOP モードに移行するときにリセットを生成します。
- 1 : リセットは生成されません。

ビット 20 **WDG_SW**

このビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、1 がロードされます。

- 0 : ハードウェアウォッチドッグ。
- 1 : ソフトウェアウォッチドッグ。

ビット 19:16 BOR_LEV : ブラウンアウトリセット閾値レベル

これらのビットでは、閾値レベルを 1.45 V から 1.55 V の電圧範囲でリセットします (パワーダウンのみ)。この特殊な例では、BOR を無効にするために、デバイス OBL シーケンスの開始時に VDD33 は BOR LEVEL 0 以上である必要があります。パワーダウンは PDR によって監視されます。BOR が無効である場合、1.65 V と VPDR 閾値の間には「グレーゾーン」が存在します (これは、VDD33 が VPDR 閾値に達するまでリセットなしで最小動作電圧 (1.65 V) を下回る場合があることを意味する)。

これらのビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、0x8 がロードされます。

1000 : 1.69 V から 1.8 V 電圧範囲のリセット閾値レベルは BOR LEVEL 0 (パワーオン)。

1001 : 1.94 V から 2.1 V 電圧範囲のリセット閾値レベルは BOR LEVEL 1 (パワーオン)。

1010 : 2.3 V から 2.49 V 電圧範囲のリセット閾値レベルは BOR LEVEL 2 (パワーオン)。

1011 : 2.54 V から 2.74 V 電圧範囲のリセット閾値レベルは BOR LEVEL 3 (パワーオン)。

1100 : 2.77 V から 3.0 V 電圧範囲のリセット閾値レベルは BOR LEVEL 4 (パワーオン)。

ビット 15:9 予約済み**ビット 8 WPRMOD**

このビットは、フラッシュプログラムメモリセクタの書き込み／読み出し保護を選択します。このビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、1 がロードされます。

0 : PCROP は無効です。WRPROT ビットは、セクタの書き込み保護として使用されます。

1 : PCROP は有効です。WRPROT ビットは、セクタの読み出し保護として使用されます。

ビット 7:0 RDPROT : 読み出し保護

このビットは、オプションバイトローディング中にロードされた保護レベルを含みます。これらのビットは読み出し専用です。オプションバイトローディング中に、この設定に不一致がある場合、0x00 がロードされます。

0xAA : レベル 0

0xCC : レベル 2

その他 : レベル 1

3.7.9 書込み保護レジスタ (FLASH_WRPROT)

アドレスオフセット : 0x20

リセット値 : 0x0000 XXXX

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

ビット 31:16 予約済み

ビット 15:0 **WRPROT** : 書込み保護機能

- FLASH_OTPR レジスタの WPRMOD = 0 では、これらのビットはフラッシュメモリの書込み保護設定を含みます（1 つ目のビットでは 1 つ目のセクタを、2 つ目のビットでは 2 つ目のページを保護するなど、各ビットで 16 KB セクタを保護）。この場合、1 = 保護されたセクタ、0 = 保護なしを意味します。
- WPRMOD = 1 の場合、データとして読み出されること（[データとしての読み出しおよび事前読み出し](#)を参照）、また、書き込まれること（同じ粒度と同じ組み合わせのビットおよびセクタを持つため）から保護するために、これらのビットを使用します。読み出し保護でフェッチから保護することはできません。この場合、1 = 保護なし、0 = セクタ保護を意味します。

WPRMOD = 0 の場合、このビットを制限なくセットまたはリセットして、相対オプションバイトを変更できます。

WPRMOD = 1 の場合、保護の強化のみ可能です。これは、ユーザが 0 を追加でき、1 は追加できないことを意味します。

全体消去によって WPRMOD ビットは削除されますが、このレジスタの内容は削除されません。全体消去後に、相対オプションバイトに 0 を書き込んで書込み保護を完全に解除する必要があります。

オプションバイトローディング中に、この設定に不一致がある場合、FLASH_OTPR レジスタの WPRMOD の内容は次のとおりです。

- この設定には 0x0000 がロードされます。
- この設定には 0xFFFF がロードされます。

FLASH_OTPR レジスタで WPRMOD のロード時に不一致がある場合（そのため 1 がロードされる）、レジスタには 0x0000 がロードされます。

3.7.10 フラッシュレジスタマップ

表 16. フラッシュインタフェース - レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|---------------|---------------|------|------|------|------|------|------|------|------|-----------|-----------|--------|------------|------------|-------|--------------|------|------|-------|------|---------|--------|--------|--------|-------------|----------|------------|--------|----------|---------|---------|---------|
| 0x00 | FLASH_ACR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PRE_READ | DESAB_8BUF | RUN_PD | SLEEP_PD | Res. | PRFTEN | LATENCY |
| | 0x00000000 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x004 | FLASH_PECR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OBL_LAUNCH | ERRIE | EOPIE | Res. | Res. | Res. | Res. | Res. | FPRG | ERASE | FIX | Res. | Res. | Res. | DATA | PRG | OPTLOCK | PRGLOCK | PELOCK |
| | 0x00000007 | | | | | | | | | | | | | | 0 | 0 | 0 | | | | | | 0 | 0 | 0 | | | | 0 | 0 | 1 | 1 | 1 |
| 0x008 | FLASH_PDKEYR | PDKEYR[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0x00000000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x00C | FLASH_PKEYR | PKEYR[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0x00000000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x010 | FLASH_PRGKEYR | PRGKEYR[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0x00000000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x014 | FLASH_OPTKEYR | OPTKEYR[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 0x00000000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x018 | FLASH_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FWERR | NOTZEROERR | Res. | Res. | RDERR | Res. | OPTVERR | SIZERR | PGAERR | WRPERR | Res. | Res. | Res. | Res. | READY | HWOFF | EOP | BSY |
| | 0x0000000C | | | | | | | | | | | | | | | 0 | 0 | | | 0 | | 0 | 0 | 0 | 0 | | | | | 1 | 1 | 0 | 0 |
| 0x01C | FLASH_OPTR | BOOT1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | nRST_STBY | nRST_STOP | WDG_SW | BOR_LEV:0] | | | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WPRMOD | RDPROT[7:0] | | | | | | | |
| | 0x00XX0XXX | X | | | | | | | | | X | X | X | X | X | X | X | | | | | | | | X | X | X | X | X | X | X | X | X |
| 0x020 | FLASH_WRPROT | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WRPROT[15:0] | | | | | | | | | | | | | | | | |
| | 0x0000XXXX | | | | | | | | | | | | | | | | | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |

3.8 オプションバイト

NVM では、オプションバイトのセットを格納するために、一部が予約されています。この情報は製品の設定に使用されます。この部分は工場で書き込まれており、別の部分はエンドユーザの管理下にあります。

エンドユーザに管理されている設定は、オプションバイト領域（32 バイト）にあります。最初の 12 バイト（3 ワード）は、ブート中に自動的にロードされ、FLASH_OTPR レジスタと FLASH_WRPOT レジスタの内容のセットに使用されます。

ブート中の読み出しにおける各ワードは、表 17 に示すように解釈されます。下位の 16 ビットにはメモリインタフェースレジスタにコピーするデータが含まれ、上位の 16 ビットには読み出した値が正しいことを確認するための補数が含まれます。ローディング中にエラーが発生した場合（上位は下位の補数ではない）、デフォルト値はレジスタに格納されます。この確認は設定で行います。セクション 3.8.2 では、保護設定に不一致がある場合に何が生じるかについて説明します。

書き込み中、ワードの上部が下部の補数であるかどうかを確認するための制御は行われません。ユーザが制御する必要があります。

表 17. オプションバイトのフォーマット

| 31-24 | 23-16 | 15-8 | 7-0 |
|------------------|------------------|------------|------------|
| 補数 オプションバイト 1 | 補数 オプションバイト 0 | オプションバイト 1 | オプションバイト 0 |

3.8.1 オプションバイトの説明

オプションバイトは、表 18 にリストされたメモリ位置から読み出すことができます。

表 18. オプションバイトの構成

| アドレス | [31:16] | [15:0] |
|-------------|--------------------|-------------------|
| 0x1FF8 0000 | nFLASH_OTPR[15:0] | FLASH_OTPR[15:0] |
| 0x1FF8 0004 | nFLASH_OTPR[31:16] | FLASH_OTPR[31:16] |
| 0x1FF8 0008 | nFLASH_WRPOT[15:0] | FLASH_WRPOT[15:0] |

3.8.2 保護フラグのロード中の不一致

オプションバイトローディング中に不一致がある場合は、メモリインタフェースではレジスタにデフォルト値をセットします。

オプションバイト領域には、3 種類の保護情報があります。

- **RDPROT**

この設定では保護レベルをセットします。次のセクションで説明するように、このレベルを変更すると NVM や製品にアクセスできる可能性が変わります。デフォルト値はレベル 1 です。レベル 1 からレベル 0 に戻ることはできますが、データ EEPROM とフラッシュプログラムメモリのすべての内容は削除されます（全体消去）。レベル 2 に移ることも常に可能ですが、レベル 2 がロードされているときに保護レベルを変更することはできません（ユーザがオプションバイトをレベル 2 を書き込んで再ロードしなかった場合、メモリインタフェースは引き続き前回のレベルで動作し、オプションバイト領域には再度異なる保護レベルを書き込めるようになる）。

- **WPRMOD**

フラグは RDPROT から独立しており、フラッシュプログラムメモリが読み出し／書き込み保護されている場合にセットされます。このフラグが 1（読み出し保護）の場合、リセットする唯一の方法は、全体消去を要求することです（保護レベルはレベル 0 に戻る）。これは、デバイスがレベル 2 にある場合、読み出し保護を解除する方法はないことを意味しています。デフォルト値は 1（読み出し保護）です。このビットにおける不一致からは、WRPROT 設定のデフォルト値も生成されます。

- **WRPROT**

この設定では、読み出し／書き込み保護するフラッシュプログラムメモリのセクタをセットします。読み出し保護が無効（WPRMOD = 0）の場合、セクタを保護するために、該当のビットに 1 をセットする必要があります。読み出し保護が有効（WPRMOD = 1）の場合、セクタを保護するために、該当のビットに 0 をセットする必要があります。ブート中に WPRMOD に不一致がある場合は、この設定には 0 がロードされ、フラッシュプログラムメモリのセクタはすべて読み出しから保護されます。WPRMOD が正しく読み出されている場合で WRPROT の読み出し時に不一致があった場合、WPRMOD = 1 のときはレジスタには 0 がロードされ、WPRMOD = 0 のときは 1 がロードされます。

そのため、保護の不一致は、通常のコードの実行に重大な影響を与えることがあります（フラッシュプログラムメモリに含まれる場合）。読み出し保護されている場合、フェッチのみを実行できます。フラッシュプログラムメモリでは、一部の値はコードの実行中にデータとして読み出されます（定数など）。読み出しからすべてのセクタを保護することは、フラッシュプログラムメモリからアプリケーションコードを実行する際に妨げになります。

3.8.3 ソフトウェアによるオプションバイトの再ロード

FLASH_PECR レジスタの OBL_LAUNCH フラグを 1 にセットすることで、オプションバイトの再ロードを要求できます。このビットは OPTLOCK = 0（および PELOCK = 0）のときのみセットできます。このビットをセットすると、進行中の書き込み／消去は完了しますが、新しい書き込み／消去または読み出し動作は実行されません。

オプションバイトの再ロードによってデバイスのリセットが生成されます（ただし、パワーダウンなし）。変更が適用されるよう、NVM でオプションバイトを変更するたびに、必ず再ロードする必要があります。OBL_LAUNCH をセット、または、V18 ドメインをパワーオン（つまり、パワーオンリセット後、またはスタンバイ後）して再ロードすることもできます。

4 巡回冗長検査計算ユニット (CRC)

4.1 概要

CRC (Cyclic Redundancy Check : 巡回冗長検査) 計算ユニットは、8、16、または 32 ビットデータワードと、ある生成多項式から、CRC コードを得るために使用されます。

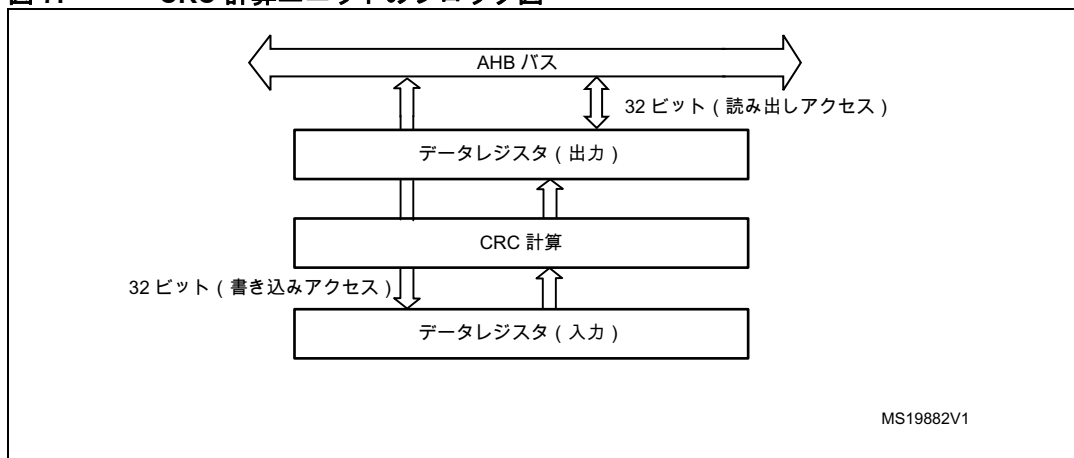
他のアプリケーションの中でも、CRC ベースのテクニックは、データ転送やストレージの整合性を確認するために使用されます。機能安全規格の範囲内では、このテクニックがフラッシュメモリの整合性を確認するひとつの手段となっています。CRC 計算ユニットは、実行時にソフトウェアのシグネチャ計算を支援します。リンク時に生成されて、特定のメモリ領域に保存されたりファレンスシグネチャと計算されたソフトウェアシグネチャが比較されます。

4.2 CRC の主な機能

- 以下のCRC-32 (イーサネット) 多項式を使用します。0x4C11DB7
$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$
- あるいは、プログラム可能なサイズ (7、8、16、32 ビット) の完全にプログラム可能な多項式を使用します。
- サイズが 8、16、32 ビットのデータを取り扱います。
- プログラム可能な CRC の初期値
- シングル入力/出力 32 ビットデータレジスタ
- 計算時のバスのストールを避けるための入力バッファ
- データサイズが 32 ビットの場合、CRC の計算は AHB クロック 4 サイクル (HCLK) 以内に行われます。
- 汎用 8 ビットレジスタ (一時ストレージとして使用可能)
- I/O データの可逆性のオプション

4.3 CRC の機能説明

図 7. CRC 計算ユニットのブロック図



CRC 計算ユニットは、1 つの 32 ビット読み出し／書き込みデータレジスタ (CRC_DR) を持っています。このレジスタを使用して、新しいデータを入力し (書き込みアクセス)、前回の CRC 計算結果を保持します (読み出しアクセス)。

データレジスタへの書き込み操作のたびに、前回の CRC 値 (CRC_DR に格納) と新しい値の組み合わせが作成されます。CRC 計算は、書き込まれるデータのフォーマットに応じ、32 ビットデータワード全体に対して、またはバイト単位で行われます。

CRC_DR レジスタは、ワード、右詰め、ハーフワード、右詰め、バイトによってアクセスできます。他のレジスタについては、32 ビットアクセスのみ可能です。

計算の時間はデータ幅に依存します。

- 32 ビットの場合、4 AHB クロックサイクル
- 16 ビットの場合、2 AHB クロックサイクル
- 8 ビットの場合、1 AHB クロックサイクル

入力バッファを使うと、前回の CRC 計算によるウェイトステートを待つことなく、すぐに第 2 のデータを書き込むことができます。

与えられたバイト数に対する書き込みアクセス数を最小限に抑えるために、データサイズを動的に調節することができます。たとえば、5 バイトの CRC は、1 ワードの書き込みと、それに続く 1 バイトの書き込みで計算することができます。

入力データを逆にして、さまざまなエンディアンネス方式を管理することができます。逆転操作は、CRC_CR レジスタの REV_IN[1:0] ビットに応じて、8、16、および 32 ビットで行うことができます。

たとえば、入力データ 0x1A2B3C4D は、CRC 計算では以下のように使用されます。

- 0x58D43CB2 (バイト単位でビットが逆転)
- 0xD458B23C (ハーフワード単位でビットが逆転)
- 0xB23CD458 (フルワードでビットが逆転)

また、出力データも、CRC_CR レジスタの REV_OUT ビットをセットすることによって逆にすることができます。

操作はビットレベルで行われます。たとえば、出力データ 0x11223344 は 0x22CC4488 に変換されます。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットを使用して、プログラム可能な値に初期化することができます（デフォルト値は 0xFFFFFFFF）。

CRC の初期値は、CRC_INIT レジスタでプログラムすることができます。CRC_DR レジスタは、CRC_INIT レジスタの書き込みアクセス時に自動的に初期化されます。

CRC_IDR レジスタを使用して、CRC 計算に関する一時的な値を保持することができます。このレジスタは、CRC_CR レジスタの RESET ビットによる影響を受けません。

多項式のプログラミング可能性

多項式係数は CRC_POL レジスタを介して完全にプログラム可能であり、多項式のサイズは CRC_CR レジスタの POLYSIZE[1:0] ビットをプログラムすることにより、7、8、16、または 32 ビットに設定できます。

CRC データが 32 ビット未満であれば、その値は CRC_DR レジスタの下位ビットから読み出すことができます。

信頼できる CRC 計算結果を得るために、CRC 計算の実行中に多項式の値やサイズを変更することはできません。そのため、CRC 計算が進行中である場合、アプリケーションは計算をリセットするか、または CRC_DR の読み出しを行ってから、多項式を変更しなければなりません。

多項式のデフォルト値は、CRC-32（イーサネット）多項式、0x4C11DB7 です。

4.4 CRC レジスタ

4.4.1 データレジスタ（CRC_DR）

アドレスオフセット：0x00

リセット値：0xFFFF FFFF

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

ビット 31:0 **DR[31:0]**：データレジスタビット

このレジスタを使用して、CRC 計算機に新しいデータを書き込みます。

読み出し時には、前回の CRC 計算結果を保持します。

データサイズが 32 ビット未満であれば、下位ビットを使用して正しい値の書き込み／読み出しを行います。

4.4.2 独立型データレジスタ (CRC_IDR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

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

- ビット 31:8 予約済み、クリア状態を保つ必要があります。
- ビット 7:0 **IDR[7:0]** : 汎用 8 ビットデータレジスタビット
これらのビットは、1 バイトの一時的なストレージ領域として使用できます。
このレジスタは、CRC_CR レジスタの RESET ビットによって生成される CRC リセットの影響を受けません。

4.4.3 制御レジスタ (CRC_CR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

- ビット 31:8 予約済み、クリア状態を保つ必要があります。
- ビット 7 **REV_OUT** : 出力データを逆にします。
このビットは、出力データビット順序の反転を制御します。
0 : ビットの順序は変わりません。
1 : 出力フォーマットのビットが反転しています。
- ビット 6:5 **REV_IN[1:0]** : 入力データを逆にします。
これらのビットは、入力データビット順序の反転を制御します。
00 : ビットの順序は変わりません。
01 : バイト単位でビットが反転しています。
10 : ハーフワード単位でビットが反転しています。
11 : ワード単位でビットが反転しています。

ビット 4:3 **POLY****SIZE**[1:0] : 多項式のサイズ
これらのビットは、多項式のサイズを制御します。
00 : 32 ビットの多項式
01 : 16 ビットの多項式
10 : 8 ビットの多項式
11 : 7 ビットの多項式

ビット 2:1 予約済み、クリア状態を保つ必要があります。

ビット 0 **RESET** : RESET ビット
このビットは、CRC 計算ユニットをリセットし、CRC_INIT レジスタに格納された値にデータレジスタをセットするために、ソフトウェアによってセットされます。このビットはセットのみが可能で、ハードウェアによって自動的にクリアされます。

4.4.4 CRC の初期値（CRC_INIT）

アドレスオフセット : 0x10
リセット値 : 0xFFFF FFFF

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

ビット 31:0 **CRC_INIT** : プログラム可能な CRC の初期値
このレジスタを使用して、CRC の初期値を書き込みます。

4.4.5 CRC 多項式（CRC_POL）

アドレスオフセット : 0x14
リセット値 : 0x04C11DB7

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

ビット 31:0 **POL**[31:0] : プログラム可能な多項式
このレジスタを使用して、CRC 計算に使用される多項式の係数を書き込みます。
多項式のサイズが 32 ビット未満であれば、下位ビットを使用して正しい値をプログラムする必要があります。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにもかかわらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

4.4.6 CRC レジスタマップ

表 19. CRC レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
|-----------|----------|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|----------|---------|-------------|---|---------------|---|---|---|------|------|-------|
| 0x00 | CRC_DR | DR[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | |
| 0x04 | CRC_IDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IDR[7:0] | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x08 | CRC_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | REV_OUT | REV_IN[1:0] | | POLYSIZE[1:0] | | | | Res. | Res. | RESET |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | | 0 | | |
| 0x10 | CRC_INIT | CRC_INIT[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| 0x14 | CRC_POL | 多項式係数 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0x04C11DB7 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

5 ファイアウォール (FW)

5.1 概要

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

5.2 ファイアウォールの主な機能

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

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

各セグメントの開始アドレスとそれぞれの長さは、ファイアウォールを有効にする前に設定する必要があります（[セクション 5.3.5：ファイアウォールの初期化](#)を参照）。

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

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

5.3 ファイアウォールの機能詳細

5.3.1 ファイアウォールの AMBA バススヌープ

ファイアウォールのペリフェラルでは、メモリ（揮発性および不揮発性）が接続されている AMBA バスをスヌープしています。アーキテクチャの全体図を [図 8](#) に示します。

Figure 1-1: Block diagram of the internal peripheral connection. The diagram shows a central vertical bus labeled "バスマトリックス" (Bus Matrix). On the left, "CORTEX M0+" and "DMA" are connected to the bus via "AHB マスタ 1" and "AHB マスタ 2" respectively. On the right, "EEPROM 64 KB", "フラッシュ プログラムメモリ 2 KB データ", and "SRAM" are connected. A "ファイアウォール" (Firewall) block is connected to the bus and the SRAM. The connection between the bus and the EEPROM/Flash/Data is labeled "AHB スレーブ" (AHB Slave). The connection between the bus and the SRAM is also labeled "AHB スレーブ".

割り込み管理

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

割り込みベクタとフラッシュプログラムメモリの最初のユーザセクタについて

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

ファイアウォールによって生成される割り込みはありません。

5.3.3 ファイアウォールセグメント

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

コードセグメント

このセグメントは、フラッシュプログラムメモリに配置されています。これには、ファイアウォールによる保護が必要な、実行対象のコードが含まれます。ファイアウォールを開くためには、「コールゲート」エントリシーケンスを使用してセグメントに達する必要があります。ファイアウォールがシステム設定レジスタの FWDIS ビットを使用して有効になっていて、「コールゲート」エントリシーケンスが順守されない場合、システムリセットが生成されます（[ファイアウォールを開く](#)を参照）。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスを設定する必要があります（[セクション 5.3.5：ファイアウォールの初期化](#)を参照）。

不揮発性データセグメント

このセグメントには、保護されたコードで使用する不揮発性データが含まれており、ファイアウォールで保護する必要があります。このセグメントへのアクセスについては、[セクション 5.3.4：セグメントへのアクセスと特性](#)で定義しています。この領域のデータにアクセスする前に、ファイアウォールを開く必要があります。不揮発性データセグメントは、フラッシュプログラムまたは 2 KB のデータ EEPROM メモリに配置する必要があります。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスを設定する必要があります（[セクション 5.3.5：ファイアウォールの初期化](#)を参照）。

揮発性データセグメント

コードセグメントに配置された保護されたコードで使用する揮発性データについて、SRAM メモリで定義する必要があります。このセグメントへのアクセスについては、[セクション 5.3.4：セグメントへのアクセスと特性](#)で定義しています。揮発性データセグメントの設定に応じて、このセグメント領域にアクセスする前にファイアウォールを開くかどうかを決める必要があります。ファイアウォールを有効にする前に、セグメントの長さやセグメントのベースアドレスおよびセグメントオプションを設定する必要があります（[セクション 5.3.5：ファイアウォールの初期化](#)を参照）。

また、揮発性データセグメントをファイアウォール設定レジスタの 2 つのビット（揮発性データ共有オプション用のビット VDS と、揮発性データ実行機能用のビット VDE）を使用して実行可能（コードの実行）セグメントまたは共有セグメントとして定義することもできます。詳細については、[表 20](#)を参照してください。

5.3.4 セグメントへのアクセスと特性

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

ファイアウォールの状態に応じたセグメントへのアクセス

3 つのセグメントにはそれぞれ固有の特性があり、これらについて [表 20](#) に示します。

表 20. ファイアウォールの状態に応じたセグメントへのアクセス

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

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

- 共有 (レジスタの VDS ビット)
これは、このセグメントに配置された領域とデータを、保護されたコードと保護されていない領域で実行されたユーザコードで共有できるということを意味しています。ファイアウォールが開いているか、閉じているか、無効であるかにかかわらず、アクセスできます。
VDS ビットは VDE ビットに優先され、その場合、最後のビット値は無視されます。これは、配置されたコードの一部を揮発性データセグメントで実行することができ、コードの実行前にファイアウォールを開く必要がないということを意味しています。

注： **ファイアウォールが閉じており (有効な状態)、コードセグメントに「コールゲート」エントリが適用されている場合で、コードセグメントから実行したコードが、共有に指定されている揮発性データセグメントにジャンプした場合 (新しい「コールゲート」エントリシーケンスの実行なし)、コードは揮発性データセグメントから実行されますが、ファイアウォールは分岐命令に対して閉じたままとなります。揮発性データセグメントからのコードの実行の終了時に、コードがコードセグメントに戻る (たとえば、関数に戻る) 場合、システムリセットが生成されます。**

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

セグメントの特性

各セグメントには、ファイアウォールで保護するセグメントサイズを定義する、固有のセグメント長レジスタがあります。コードセグメント長レジスタの CSL レジスタ、不揮発性データセグメント長レジスタの NVDSL レジスタ、および揮発性データセグメント長レジスタの VDSL レジスタです。各セグメントの粒度と領域の範囲について、表 21 に示します。

表 21. セグメントの粒度と領域の範囲

| セグメント | 粒度 | 領域の範囲 |
|--------------|---------|--------------------|
| コードセグメント | 256 バイト | 最大 64 KB - 256 バイト |
| 不揮発性データセグメント | 256 バイト | 最大 64 KB - 256 バイト |
| 揮発性データセグメント | 64 バイト | 8 KB - 64 バイト |

5.3.5 ファイアウォールの初期化

初期化フェーズは、ユーザコードの実行の最初の段階で行います（[書込み保護機能](#)を参照）。

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

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

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

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

次に初期化手順について示します。

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

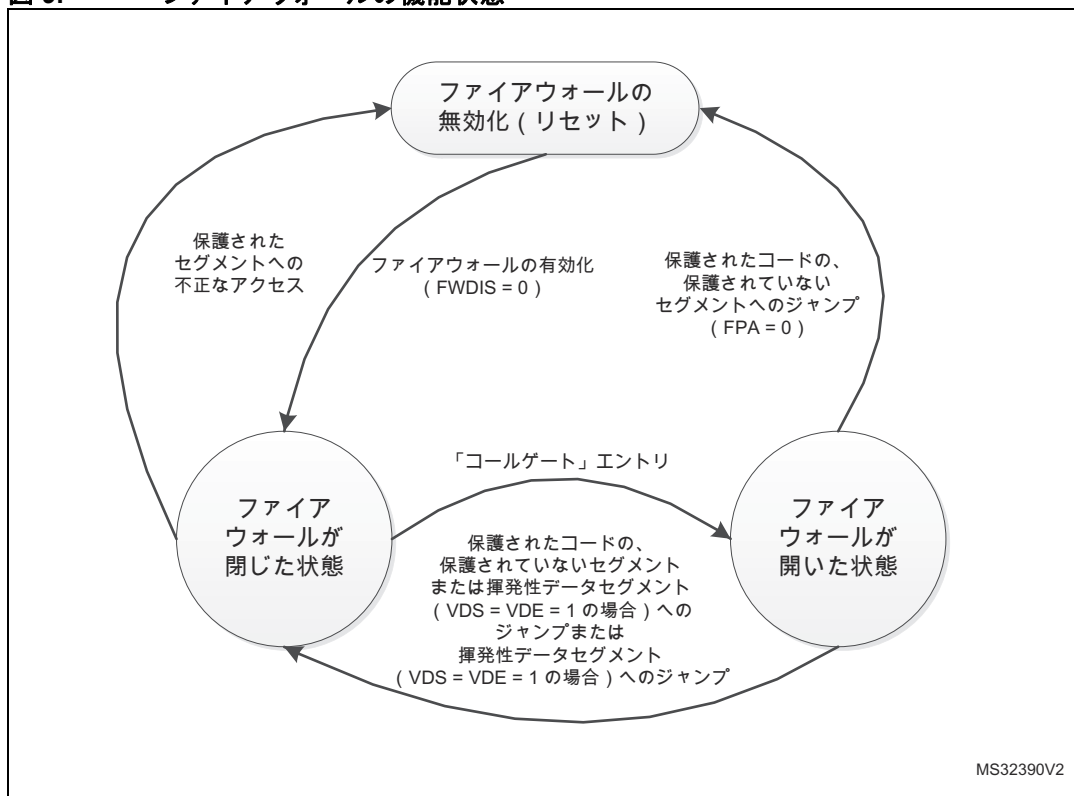
ファイアウォール設定レジスタ（FW_CR レジスタ）は、ファイアウォールが有効な場合でも動的に管理することができる唯一のレジスタです。ただし、このレジスタは不揮発性データセグメントと同じ方法で保護されます。このレジスタへのアクセスは、ファイアウォールが開いている場合で、不揮発性データセグメントが定義されている場合にのみ可能です（NVDSL レジスタは 0 以外の値）。

5.3.6 ファイアウォールの状態

図 9 に示すように、ファイアウォールには、3 つの異なる状態があります。

- 無効：FWDIS ビットはリセット後にデフォルトでセットされています。ファイアウォールはアクティブではありません。
- 閉じている：ファイアウォールでは、3 つのセグメント（コード、不揮発性データ、および揮発性データ）へのアクセスを保護します。
- 開いている：セクション 5.3.4：セグメントへのアクセスと特性で定義されているとおり、ファイアウォールでは保護されたセグメントへのアクセスを許可します。

図 9. ファイアウォールの機能状態



ファイアウォールを開く

ファイアウォールを有効にすると、すぐに閉じられます。これは、保護された領域へのほとんどのアクセスが禁止されていることを意味します（[セクション 5.3.4：セグメントへのアクセスと特性](#)を参照）。ファイアウォールを開いて保護されたセグメントと相互作用するには、この後説明する「コールゲート」シーケンスを適用する必要があります。

「コールゲート」シーケンス

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

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

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

ファイアウォールが開くと、[セクション 5.3.4：セグメントへのアクセスと特性](#)に示すように、ただちに保護された領域にアクセスできるようになります。

ファイアウォールを閉じる

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

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

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

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

注： VDS = VDE = 1 の場合、保護されたコードが揮発性データセグメントにジャンプすると、ファイアウォールは閉じます。

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

5.4 ファイアウォールレジスタ

5.4.1 コードセグメントの開始アドレス (FW_CSSA)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

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

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

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

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

注: これらのビットは、ファイアウォールを有効にする前にもみ書き込み可能です。[セクション 5.3.5: ファイアウォールの初期化](#)を参照してください。

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

5.4.2 コードセグメント長 (FW_CSL)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

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

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

ビット 21:8 LENG[21:8] : コードセグメント長

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

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

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

これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 5.3.5: ファイアウォールの初期化を参照してください。

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

5.4.3 不揮発性データセグメントの開始アドレス (FW_NVDSSA)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

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

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

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

注： これらのビットは、ファイアウォールを有効にする前にのみ書き込み可能です。セクション 5.3.5: ファイアウォールの初期化を参照してください。

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

5.4.4 不揮発性データセグメント長 (FW_NVDSL)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

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

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

ビット 21:8 **LENG[21:8]** : 不揮発性データセグメント長

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

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

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

これらのビットは、ファイアウォールを有効にする前のみ書き込み可能です。セクション 5.3.5: ファイアウォールの初期化を参照してください。

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

5.4.5 揮発性データセグメントの開始アドレス (FW_VDSSA)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:6 **ADD[15:6]** : 揮発性データセグメントの開始アドレス

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

注 : これらのビットは、ファイアウォールを有効にする前のみ書き込み可能です。セクション 5.3.5: ファイアウォールの初期化を参照してください。

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



5.4.6 揮発性データセグメント長 (FW_VDSL)

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:6 **LENG[15:6]** : 不揮発性データセグメント長

LENG[15:6] では、64 バイトの倍数であるバイトで表される、不揮発性データセグメントのサイズを選択します。

セグメント領域は、{ADD[15:6], 0x00} から {ADD[15:6]+LENG[15:6], 0x00} - 0x01 まで定義されています。

注 : ファイアウォールを有効にした後でLENG[15:6] = 0 である場合、このセグメントは定義されないため、ファイアウォールで保護されません。

これらのビットは、ファイアウォールを有効にする前のみ書き込み可能です。[セクション 5.3.5: ファイアウォールの初期化](#)を参照してください。

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

5.4.7 設定レジスタ (FW_CR)

アドレスオフセット : 0x20

リセット値 : 0x0000 0000

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

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

ビット 2 VDE : 揮発性データ実行

0 : VDS = 0 の場合、揮発性データセグメントは実行できません。

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

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

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

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

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

ビット 1 VDS : 揮発性データ共有

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

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

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

ビット 0 FPA : ファイアウォールプリアーム

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

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

[ファイアウォールを閉じる](#)を参照してください。

このレジスタは不揮発性データセグメントと同じ方法で保護されます（[セクション 5.3.5 : ファイアウォールの初期化](#)を参照）。

5.4.8 ファイアウォールレジスタマップ

次の表に、ファイアウォールレジスタマップとリセット値を示します。

表 22. ファイアウォールレジスタマップとリセット値

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



6 電源制御 (PWR)

6.1 電源

BOR が使用可能である場合、このデバイスでは、1.8 から 3.6 V V_{DD} の動作電源（パワーダウン時は 1.65 V まで）が必要です。BOR が使用不可である場合、このデバイスでは、1.65 から 3.6 V V_{DD} の動作電源が必要です。

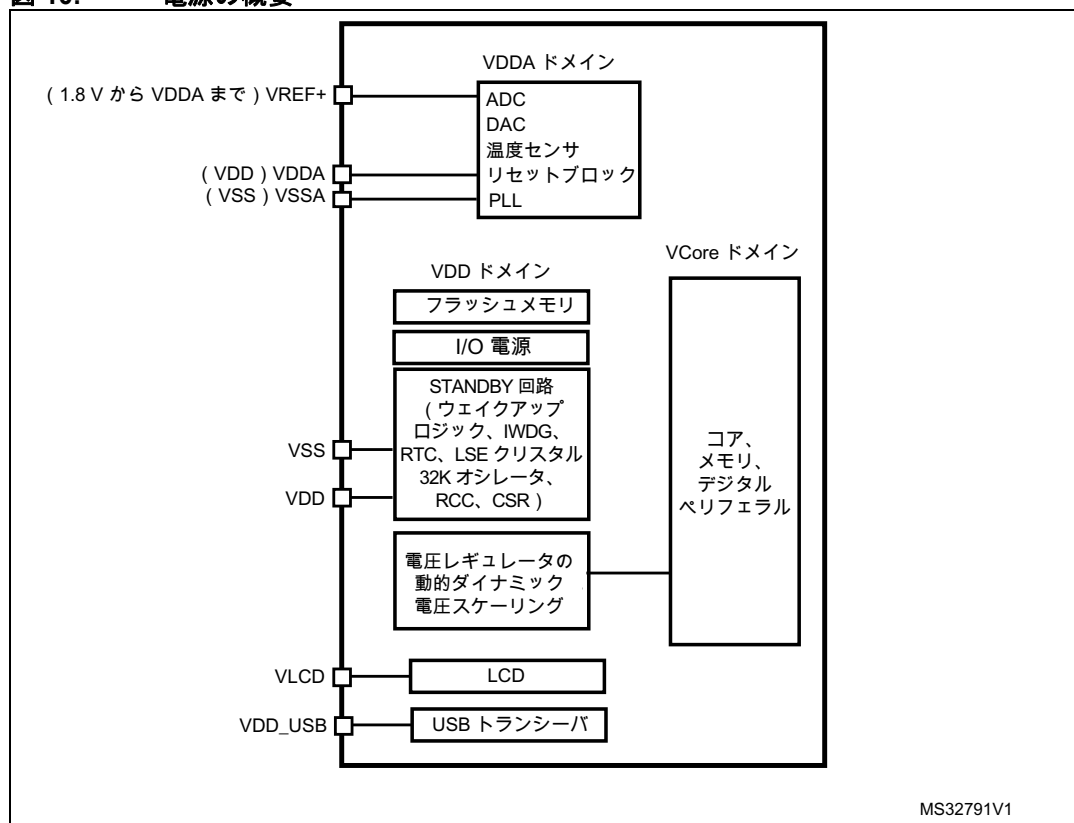
1.2 から 1.8 V の範囲の内部のデジタル電源を供給するために、リニア電圧レギュレータが組み込まれています。

- BOR が使用可能である場合は、 $V_{DD} = 1.8$ V（パワーオン時）または 1.65 V（パワーダウン時）から 3.6 V。BOR が使用不可である場合は、 $V_{DD} = 1.65$ V から 3.6 V。
 V_{DD} は I/O および内蔵レギュレータの外部電源。 V_{DD} ピンから外部より供給されます。
- $V_{CORE} = 1.2$ から 1.8 V
 V_{CORE} は、デジタルペリフェラル、SRAM、およびフラッシュメモリの電源。内蔵電圧レギュレータによって生成されます。 V_{DD} に応じて、3 つの V_{CORE} の範囲をソフトウェアによって選択可能（図 11 を参照）。
- BOR が使用可能である場合は、 V_{SSA} 、 $V_{DDA} = 1.8$ V（パワーオン時）または 1.65 V（パワーダウン時）から 3.6 V、および BOR が使用不可である場合は、 V_{SSA} 、 $V_{DDA} = 1.65$ から 3.6 V。
 V_{DDA} は、ADC、DAC、リセットブロック、RC オシレータ、および PLL の外部アナログ電源。ADC を使用している場合、 V_{DDA} に適用される最小電圧は 1.8 V。
- V_{REF+}
 V_{REF+} は入力基準電圧。
 V_{REF+} は TFBGA64 パッケージでのみ外部ピンとして使用可能。それ以外の場合は、 V_{DDA} に結合されています。
- $V_{LCD} = 2.5$ から 3.6 V
LCD コントローラの電源は、 V_{LCD} ピンから外部的に、または内蔵ステップアップコンバータによって生成される内蔵電圧から内部的に供給されます。
- $V_{DD_USB} = 3.0$ から 3.6 V
 V_{DD_USB} は、フルスピードトランシーバ専用の独立した USB 電源。USB オルタネート機能として設定されている場合に、PA11 および PA12 ピンで使用可能。

注： V_{DD_USB} 値は、 V_{DD} および V_{DDA} に依存しません。ただし、 V_{DD_USB} は、最後にデバイスに供給され、かつ最初に切断される電源である必要があります。3 つの電源がシャットダウンしても、 V_{DD_USB} が短時間アクティブなままであり、 V_{DDA}/V_{DDIO} が機能範囲を下回る場合は、デバイスは損傷を受けません。

V_{DD_USB} が切断されてもデバイスは機能します。

図 10. 電源の概要



1. V_{DDA} と V_{SSA} はそれぞれ V_{DD} と V_{SS} に接続する必要があります。
2. 使用する動作電源範囲に応じて、一部のペリフェラルは機能または性能が限定されます。
3. V_{REF+} はTFBGA64 パッケージでのみ使用可能です。

6.1.1 独立した A/D および DAC コンバータ用電源と基準電圧

変換の精度を向上させるため、ADC および DAC には独立した電源が供給されます。この電源は、PCB のノイズを防ぐために、個別にフィルタシールドすることができます。

- ADC 用の電源入力、独立した V_{DDA} ピンから供給されます。
- ADC 電源のグラウンドは、独立した V_{SSA} ピンとして用意されています。

TFBGA64 について

低電圧入力および出力時の精度を確保するため、 V_{DD} より低い独立した外部基準電圧を V_{REF+} に接続できます。 V_{REF+} は、アナログ入力 (ADC) または出力 (DAC) 信号のフルスケール値に該当する最高電圧です。

ADC および DAC について次のように表すことができます。

$$1.8\text{ V} \leq V_{REF+} < V_{DDA}$$

64 ピンまたはそれ以下のパッケージ (BGA パッケージを除く) について

V_{REF+} ピンは使用できません。このピンは、ADC 用の電源 (V_{DDA}) に内部接続されています。

6.1.2 独立した LCD 電源

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

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

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

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

6.1.3 RTC および RTC バックアップレジスタ

本リアルタイムクロック (RTC) は、独立した BCD タイマ/カウンタです。この RTC には、時刻クロック/カレンダー、プログラム可能なアラーム割り込みを 2 本、割り込み機能を備えたプログラム可能な周期的ウェイクアップフラグを搭載しています。この RTC には、5 個のバックアップデータレジスタ (20 バイト) が搭載されています。これらのバックアップレジスタは、タンパ検出イベントが発生するとリセットされます。詳細については、[リアルタイムクロック \(RTC\)](#) セクションを参照してください。

RTC レジスタアクセス

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

1. RCC_APB1ENR レジスタの PWREN ビットをセットして、電源インタフェースクロックを有効にします。
2. PWR_CR レジスタの DBP ビットをセットします ([セクション 6.4.1](#) を参照)。
3. RCC_CSR レジスタの RTCSEL[1:0] ビットから、RTC クロックソースを選択します。
4. RCC_CSR レジスタの RTCEN ビットをプログラミングすることによって、RTC クロックを有効にします。

6.1.4 電圧レギュレータ

STANDBY 回路以外のすべてのデジタル回路に電圧を供給するリニア電圧レギュレータが組み込まれています。レギュレータ出力電圧 (V_{CORE}) は、ソフトウェアによって、1.2 から 1.8 V (typ) の 3 つの異なる範囲にプログラムすることができます ([セクション 6.1.5](#) を参照)。

リセット後、電圧レギュレータは常に使用可能になります。アプリケーションモードに応じて、メイン (MR)、低電力 (LPR)、パワーダウンの 3 つの異なるモードで動作します。

- RUN モードでは、レギュレータはメイン (MR) モードで、 V_{CORE} ドメイン (コア、メモリ、デジタルペリフェラル) にフル電力を供給します。
- 低電力 RUN モードでは、レギュレータは低電力 (LPR) モードで、レジスタと内蔵 SRAM の内容を保持したまま、 V_{CORE} ドメインに低電力を供給します。
- SLEEP モードでは、レギュレータはメイン (MR) モードで、レジスタと内蔵 SRAM の内容を保持したまま、 V_{CORE} ドメインにフル電力を供給します。
- 低電力 SLEEP モードでは、レギュレータは低電力 (LPR) モードで、レジスタと内蔵 SRAM の内容を保持したまま、 V_{CORE} ドメインに低電力を供給します。
- STOP モードでは、レジスタと内蔵 SRAM の内容を保持したまま、レギュレータは V_{CORE} ドメインに低電力を供給します。
- STANDBY モードでは、レギュレータの電源がオフになります。STANDBY 回路を除き、レジスタと SRAM の内容は失われます。

6.1.5 ダイナミック電圧スケーリングの管理

ダイナミック電圧スケーリングは、状況に応じて、デジタルペリフェラル (V_{CORE}) で使用する電圧の増減を行う電源管理技術です。

V_{CORE} を増加するダイナミック電圧スケーリングをオーバーボルティングと言います。これにより、デバイスの性能を向上させることができます。STM32L0x3 の動作条件と性能の関係については、[図 11](#) を参照してください。

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

レンジ1

レンジ 1 は、「高性能」範囲です。

電圧レギュレータからは、 V_{DD} 入力電圧が 1.71 V より上である限り、1.8 V 電圧 (typ) を出力します。このレンジでは、フラッシュプログラムと消去動作を実行できます。

CPU の周波数変化は、変更前と変更後の間で、以下の条件に従う必要があります。

- $f_{\text{CPU}} (\text{変更後}) < 4f_{\text{CPU}} (\text{変更前})$ 。
- さらに、この 2 回の変更の間に、5 μs の遅延を入れる必要があります。たとえば、4.2 から 32 MHz に切り替えるには、4.2 から 16 MHz に切り替えて 5 μs 待ち、その後、16 から 32 MHz に切り替えます。

レンジ 2 および 3

レギュレータは、 V_{DD} (1.65 から 3.6 V) の制限なく、制御された 1.5 V (typ、レンジ 2) または 1.2 V (typ、レンジ 3) を出力するようにプログラムできます。

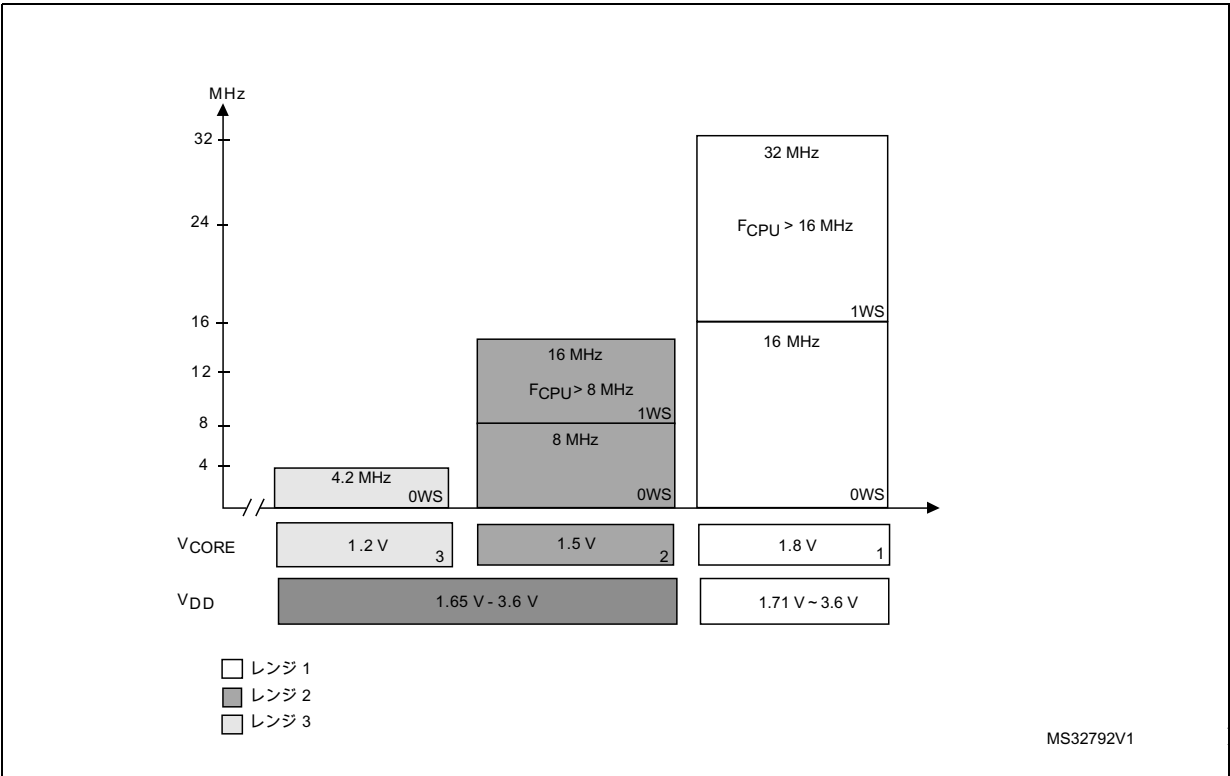
- 1.5 V で、フラッシュメモリは機能しますが、読み出しアクセス時間は中程度です。これは「中程度の性能」レンジです。フラッシュメモリへのプログラムおよび消去動作は可能です。
- 1.2 V で、フラッシュメモリは機能しますが、読み出しアクセス時間は遅くなります。これは「低性能」レンジです。この条件下では、フラッシュメモリへのプログラムおよび消去動作はできません。

各レンジの性能の詳細については、表 23 を参照してください。

表 23. V_{CORE} のレンジにおける性能

| CPU 性能 | 電源性能 | V_{CORE} レンジ | Typical 値 (V) | 最大周波数 (MHz) | | V_{DD} 範囲 |
|--------|------|----------------|---------------|-------------|------|-------------|
| | | | | 1 WS | 0 WS | |
| 高 | 低 | 1 | 1.8 | 32 | 16 | 1.71~3.6 |
| 中 | 中 | 2 | 1.5 | 16 | 8 | 1.65~3.6 |
| 低 | 高 | 3 | 1.2 | 4.2 | 4.2 | |

図 11. V_{DD} と V_{CORE} のレンジにおける性能



6.1.6 ダイナミック電圧スケーリングの設定

電圧レギュレータのレンジをプログラムするには、次のシーケンスが必要です。

1. V_{DD} を確認し、使用可能なレンジを識別します (図 11: V_{DD} と V_{CORE} のレンジにおける性能を参照)。
2. PWR_CSR の VOSF ビットをポーリングします。0 にリセットされるまで待ちます。
3. PWR_CR レジスタに VOS[12:11] ビットをセットして、電圧のスケーリングレンジを設定します。
4. PWR_CSR レジスタの VOSF ビットをポーリングします。0 にリセットされるまで待ちます。

注： 電圧スケーリングの設定中は、レギュレータが安定するまで、システムクロックは停止します (VOSF=0)。割り込みに重要な反応時間が必要になる場合に備え、使用するペリフェラル (タイマーや通信など) に応じて、アプリケーションの開発時にこれを考慮する必要があります。

6.1.7 V_{DD} が 1.71 V を下回った時の電圧レギュレータとクロックの管理

V_{CORE} のレンジ 1 が選択され、 V_{DD} が 1.71 V を下回った場合、アプリケーションはシステムを再設定する必要があります。

システムを再設定するには、次の 3 ステップのシーケンスを行う必要があります。

1. V_{DD} が 1.71 V を下回ったことを検出します。
PVD を使用して V_{DD} 電圧を監視し、電圧が選択したレベルを下回ったときに割り込みを生成します。1.71 V の電圧制限を検出するには、アプリケーションはソフトウェアで PVD 閾値 2 (typ: 2.26 V) で選択できます。PVD の詳細については、[セクション 6.2.3](#) を参照してください。
2. クロック周波数を、次のステップで選択する電圧レンジに合わせます。
1.71 V を下回る場合、システムクロック周波数はレンジ 2 の場合は 16 MHz に、レンジ 3 の場合は 4.2 MHz に制限されます。
3. 必要な電圧レンジを選択します。
 V_{DD} が 1.71 V を下回る場合、レンジ 2 またはレンジ 3 のみが選択可能である点に注意してください。

注： V_{CORE} のレンジ 2 またはレンジ 3 が選択され、 V_{DD} が 1.71 V を下回る場合、システムの再設定は不要です。

6.1.8 V_{CORE} のレンジ修正時の電圧レギュレータとクロックの管理

V_{DD} が 1.71 V を上回る場合は、3 つの電圧レンジのいずれかを選択できます。

- 電圧レンジが目標の電圧範囲を上回る場合 (例：レンジ 1 から 2)
 - a) クロック周波数を、次のステップで選択する低い電圧レンジに合わせます。
 - b) 必要な電圧レンジを選択します。
- 電圧レンジが目標の電圧範囲を下回る場合 (例：レンジ 3 から 1)
 - a) 必要な電圧レンジを選択します。
 - b) 必要に応じてクロック周波数を調整します。

V_{DD} が 1.71 V を下回る場合、レンジ 2 および 3 のみを選択できます。

- レンジ 2 から レンジ 3
 - a) クロック周波数を電圧レンジ 3 に合わせます。
 - b) 電圧レンジ 3 を選択します。
- レンジ 3 から レンジ 2
 - a) 電圧レンジ 2 を選択します。
 - b) 必要に応じてクロック周波数を調整します。

6.1.9 V_{DD} の範囲が 1.71 V から 2.0 V である場合の電圧レンジと制限

STM32L0x3 電圧レギュレータは、超低電力用に設計されたアーキテクチャに基づきます。外部コンデンサは使用しません。このようなレギュレータは、負荷の速い変化の影響を受けます。この場合、出力電圧は短時間の間、低下します。32 MHz 動作を確実にを行うために 1.65 V よりも高いコア電圧が必要であることを考慮すると、非常に低い V_{DD} 電圧 (例： V_{DD} の最小値の 1.71 V) にとって、この現象は重要です。

$V_{DD}=1.8\text{ V}\pm 5\%$ 、接合部温度 $105\text{ }^{\circ}\text{C}$ 、1 ウェイトステート、および V_{CORE} のレンジ 1 で 32 MHz 動作を確実にを行うには、RUN モードの CPU 周波数を 1 回で 4 の比率を超える変更を防ぐよう管理する必要があります。2 回の変更の間に、 $5\text{ }\mu\text{s}$ の遅延を入れる必要があります。低電力モードからのウェイクアップの際に、制限はありません。

6.2 電源供給スーパバイザ

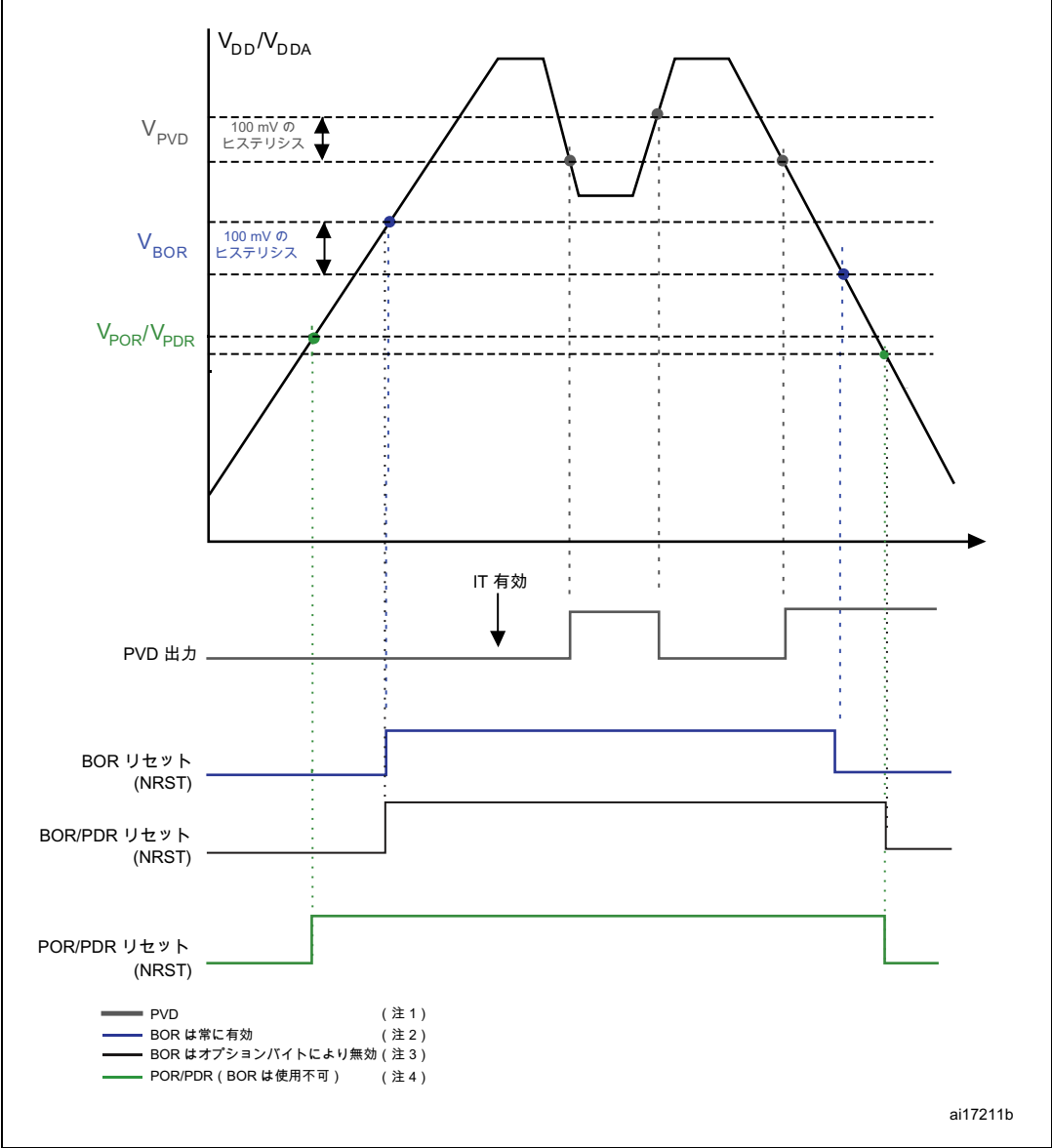
このデバイスには、統合ゼロパワー パワーオンリセット (POR) / パワーダウンリセット (PDR) に加え、ブラウンアウトリセット (BOR) 回路が搭載されています。1.8 と 3.6 V の間で動作するデバイスでは、BOR はパワーオン時に常にアクティブであるため、1.8 V から適切な動作を確実に行うことができます。1.8 V の BOR 閾値に達すると、オプションバイトローディング処理が開始され、デフォルトの閾値の確認／修正、または BOR を永久的に無効（この場合、パワーダウン時の V_{DD} の最小値は 1.65 V）にします。1.65 V と 3.6 V の間で動作するデバイスでは、BOR は永久的に無効です。したがって、パワーオン時の起動時間は $1\text{ ms}(\text{typ})$ まで短縮されます。

オプションバイトによって、1.65 から 3 V の間の 5 つの BOR 閾値を設定できます。STOP モードで消費電力を削減するために、内部基準電圧 (V_{REFINT}) を自動的にオフにすることができます。デバイスは、 V_{DD} が規定の閾値 (V_{POR} 、 V_{PDR} 、または V_{BOR}) を下回っている間は、外部のリセット回路を必要とせずに、リセットモードを維持することができます。

このデバイスには、 V_{DD}/V_{DDA} 電源を監視し、それを V_{PVD} 閾値と比較する内蔵のプログラム可能な電圧検出器 (PVD) があります。1.85 と 3.05 V の間にある 7 つの異なる PVD レベルを 200 mV ステップで、ソフトウェアによって選択できます。 V_{DD}/V_{DDA} が V_{PVD} 閾値を下回った場合、および／または V_{DD}/V_{DDA} が V_{PVD} 閾値を上回った場合に、割り込みを生成できます。割り込みサービスルーチンにて警告メッセージを生成、および／または MCU を安全な状態に設定します。PVD はソフトウェアによって有効にされます。

異なる電源供給スーパバイザ (POR、PDR、BOR、PVD) について、[図 12](#) に示します。

図 12. 電源供給スーパバイザ



1. PVD は、すべてのデバイス上で使用可能で、ソフトウェアで有効または無効にすることができます。
2. BOR は、1.8 から 3.6 V で動作するデバイスでのみ使用可能で、オプションバイトで無効にしない限り、POR/PDR 閾値をマスクします。
3. BOR がオプションバイトで無効にされた場合、 V_{DD} が PDR レベルを下回るとリセットがアサートされます。
4. 1.65 から 3.6 V で動作するデバイスでは、BOR は無効です。また、リセットは V_{DD} が POR レベルを上回ると解放され、 V_{DD} が PDR レベルを下回るとアサートされます。

6.2.1 パワーオンリセット (POR) / パワーダウリセット (PDR)

このデバイスには、1.5 V までの動作を可能にする POR/PDR 統合回路が搭載されています。

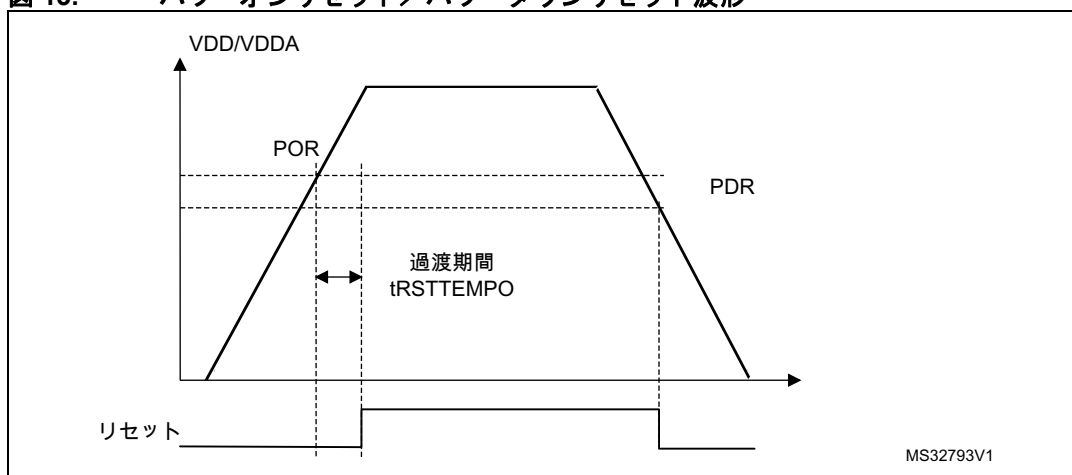
パワーオン時、デバイスは、 V_{DD}/V_{DDA} が規定の閾値 V_{POR} を下回っている間は、外部のリセット回路を必要とせず、リセットモードを維持します。POR 機能は常に有効です。POR 閾値は 1.5 V です。

パワーダウン時、PDR は、電源電圧 (V_{DD}) が V_{PDR} 閾値を下回ると、デバイスをリセット状態に保持します。PDR 機能は常に有効です。PDR 閾値は 1.5 V です。

POR と PDR は、BOR が無効な場合にのみ使用されます (セクション 6.2.2: ブラウンアウトリセット (BOR) を参照)。最小動作電圧 (1.65 V) を保証するには、BOR を BOR レベル 0 に設定する必要があります。BOR が無効の場合、最小動作電圧 (1.65 V) と V_{POR}/V_{PDR} 閾値の間には「グレーゾーン」が存在します。これは、 V_{DD} が V_{PDR} 閾値に達するまでにデバイスのリセットなしで 1.65 V を下回る場合があることを意味しています。

パワーオンリセット/パワーダウリセットの 閾値の詳細については、データシートの電気的特性の項を参照してください。

図 13. パワーオンリセット/パワーダウリセット波形



6.2.2 ブラウンアウトリセット (BOR)

パワーオン時、電源電圧が規定の V_{BOR} の閾値に達するまで、ブラウンアウトリセット (BOR) によってデバイスはリセット状態に保持されます。

1.65 から 3.6 V で動作するデバイスでは、BOR オプションは使用できず、電源は POR/PDR によって監視されます。POR/PDR 閾値が 1.5 V であるため、 V_{POR}/V_{PDR} 閾値と最小動作電圧 1.65 V の間には「グレーゾーン」が存在します。

1.8 から 3.6 V で動作するデバイスでは、BOR はパワーオン時に常にアクティブであり、閾値は 1.8 V です。

システムリセットが解放されると、オプションバイトローディングによって BOR レベルを再設定または無効化できます。

BOR レベルが最低レベルに保持されている場合 (パワーオン時 1.8 V、パワーダウン時 1.65 V)、システムリセットは BOR によって完全に管理され、動作電圧は安全な範囲に入ります。

また、BOR オプションがオプションバイトによって無効化されると、パワーダウリセットは PDR によって制御され、1.65 V と V_{PDR} の間には「グレーゾーン」が存在します。

V_{BOR} は、デバイスオプションバイトを介して設定されます。デフォルトで、レベル 4 閾値がアクティブになります。プログラム可能な V_{BOR} の閾値は 5 種類から選択できます。

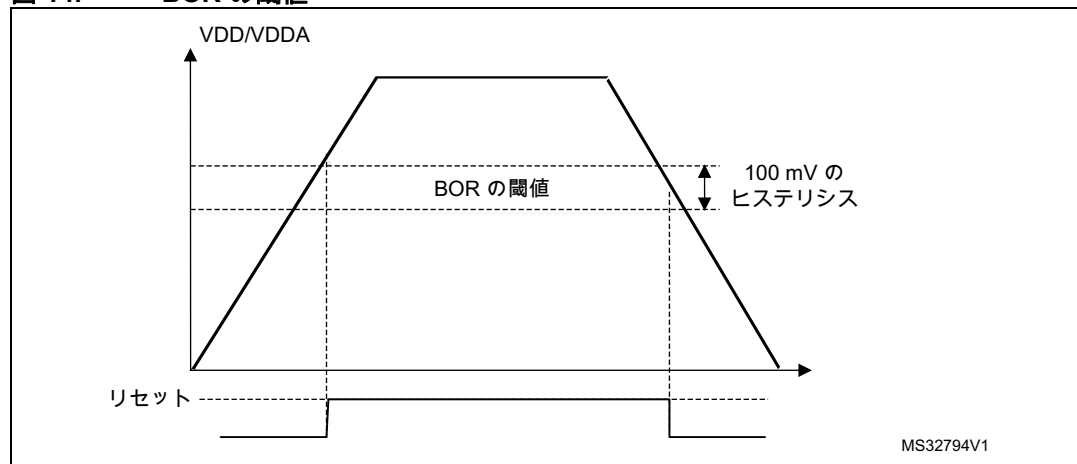
- BOR レベル 0 (V_{BOR0}) : 1.69 から 1.80 V 電圧範囲のリセット閾値レベル
- BOR レベル 1 (V_{BOR1}) : 1.94 から 2.1 V 電圧範囲のリセット閾値レベル
- BOR レベル 2 (V_{BOR2}) : 2.3 から 2.49 V 電圧範囲のリセット閾値レベル
- BOR レベル 3 (V_{BOR3}) : 2.54 から 2.74 V 電圧範囲のリセット閾値レベル
- BOR レベル 4 (V_{BOR4}) : 2.77 から 3.0 V 電圧範囲のリセット閾値レベル

電源電圧 (V_{DD}) が、選択された V_{BOR} の閾値を下回ると、デバイスリセットが生成されます。 V_{DD} が V_{BOR} の上限を上回る場合、デバイスリセットは解放され、システムを開始できます。

BOR は、デバイスのオプションバイトをプログラミングすることによって、無効にすることができます。BOR 機能を無効にするには、デバイスオプションバイトのプログラミングシーケンスを開始するために、 V_{DD} を V_{BOR0} よりも高く設定する必要があります。パワーオンとパワーダウンは、POR と PDR によって監視されます ([セクション 6.2.1 : パワーオンリセット \(POR\) / パワーダウンリセット \(PDR\)](#) を参照)。

BOR 閾値のヒステリシスは約 100 mV です (電源電圧の立ち上がりエッジと立ち下がりエッジの間)。

図 14. BOR の閾値



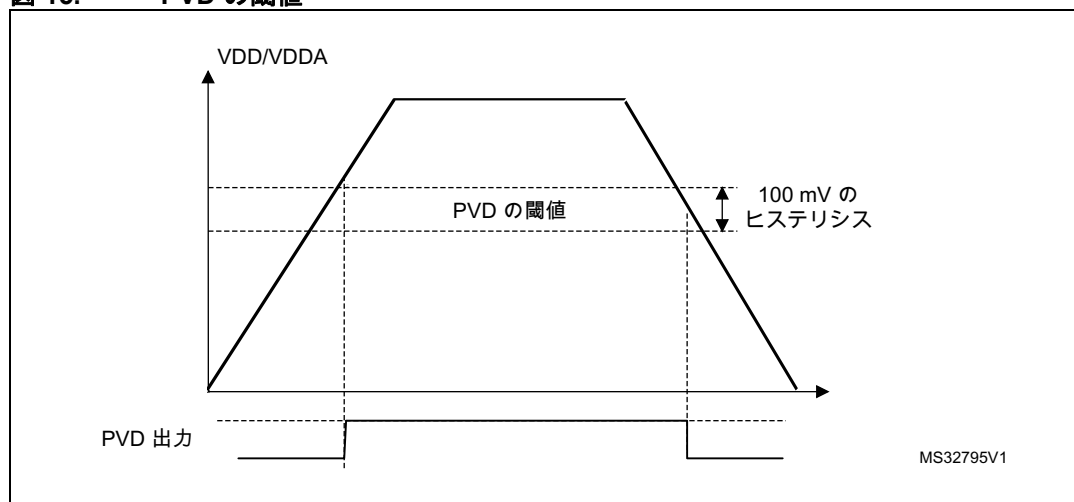
6.2.3 プログラム可能な電圧検出器 (PVD)

PVD を使用して、PWR_CR の PLS[2:0] ビットで選択した閾値と比較することで V_{DD} 電源を監視できます ([セクション 6.4.1](#) を参照)。

PVD では、内部的に VREFINT と比較される外部入力アナログ電圧 (PVD_IN) を使用できます。PVD_IN (PB7) は、PLS[2:0] = 111 の場合、アナログモードに設定する必要があります。PVD は PVDE ビットをセットすることで有効になります。

V_{DD} が PVD の閾値よりも高いか低いかを示すために、PWR_CSR の PVDO フラグが使用できます ([セクション 6.4.2](#) を参照)。このイベントは EXTI のライン 16 に内部接続され、EXTI レジスタで有効な場合は割り込みリクエストを生成させることができます。EXTI ライン 16 の立ち上がり/立ち下がりエッジの設定に応じて、 V_{DD} が PVD の閾値を下回るか、上回ったとき、あるいはその両方で、PVD 出力割り込みを生成させることができます。たとえば、サービスルーチンで、緊急停止処理を実行することなどが可能です。

図 15. PVD の閾値



6.2.4 内部基準電圧 (V_{REFINT})

内部基準電圧 (V_{REFINT}) によって、アナログペリフェラルに安定した電圧が提供されます。内部基準電圧 (V_{REFINT}) によって管理される機能は、BOR、PVD、ADC、HSI48、LCD、およびコンパレータです。これらの機能のいずれかが使用される場合、内部基準電圧 (V_{REFINT}) は常に有効です。

内部基準電圧の消費電力は、特に STOP モードおよび STANDBY モードで無視できません。消費電力を削減するために、PWR_CR レジスタの ULP ビット (超低電力) をセットし、内部基準電圧を無効にできます。ただし、この場合、STOP/STANDBY モードの終了時に、内部基準電圧で管理された機能は、内部基準電圧の起動時間中 (最大 3 ms) は信頼できなくなります。

デバイスは、ウェイクアップ時間を短縮するために、内部基準電圧の起動時間を待たずに STOP/STANDBY モードを終了することができます。これを実行するには、STOP/STANDBY モードに移行する前に PWR_CR レジスタで FWU ビット (早いウェイクアップ) をセットします。

ULP ビットがセットされている場合、STOP/STANDBY モードに移行する前に有効化された機能は、これらのモードが実行中の間は無効になり、FWU 値にかかわらず、内部基準電圧の起動時間終了後に再度有効になります。PWR_CSR レジスタの VREFINTRDYF フラグは、内部基準電圧がレディ状態であることを示します。

6.3 低電力モード

デフォルトでは、マイクロコントローラは、システムリセットまたはパワーオンリセット後は RUN モードです。RUN モードでは、CPU クロックとして HCLK が使用され、プログラムコードが実行されます。外部イベント待ちなど、CPU の連続実行が不要なときの節電のために、いくつかの低電力モードが用意されています。消費電力の節減、性能、起動時間の短縮、使用可能なウェイクアップソースを考慮した最適なモード選択はユーザに委ねられています。

このデバイスは、次の 5 つの低電力モードを備えています。

- 低電力 RUN モード：レギュレータは低電力モード、限定クロック周波数、限定数のペリフェラルが動作
- SLEEP モード：Cortex[®]-M0+ コアは停止、ペリフェラルは動作状態を維持
- 低電力 SLEEP モード：Cortex[®]-M0+ コアは停止、限定クロック周波数、限定数のペリフェラルが動作、レギュレータは低電力モード、RAMはパワーダウン状態、フラッシュは停止
- STOP モード：すべてのクロックを停止、レギュレータは低電力モードで動作
- STANDBY モード：V_{CORE} ドメイン電源オフ

さらに、次の手段のいずれかによって RUN モードの消費電力を節減できます。

- システムクロックの低速化
- 使用しない APBx および AHBx ペリフェラルへのクロックのゲーティング

表 24. 低電力モードの概要

| モード名 | エントリ | ウェイクアップ | V _{CORE} ドメイン クロックへの影響 | V _{DD} ドメイン クロック への影響 | 電圧レギュレータ |
|--|--|--|---|--|---------------------------|
| 低電力 RUN | LPSSDR および LPRUN ビット + クロックの設定 | レギュレータはメイン レギュレータ (1.8 V) が 強制的に適用 | なし | なし | 低電力モード |
| SLEEP (Sleep-now または Sleep-on-exit) | WFI | 割り込み | CPU クロックオフ 他のクロック およびアナログ クロックソース への影響なし | なし | オン |
| | WFE | ウェイクアップイベント | | | |
| 低電力 SLEEP (Sleep-now または Sleep-on-exit) | LPSSDR ビット + WFI | 割り込み | CPU クロックオフ 他のクロック およびアナログ クロックソース への影響なし フラッシュ クロックオフ | なし | 低電力モード |
| | LPSSDR ビット + WFE | ウェイクアップイベント | | | |
| STOP | PDDS ビット、 LPSSDR ビット + SLEEPDEEP ビット + WFI または WFE | 任意の EXTI ライン (EXTI レジスタで設定、 内部ラインおよび 外部ライン) | すべての V _{CORE} ドメインクロック オフ | HSI16 および HSE および MSI オシレータ オフ | 低電力モードでオン (PWR_CR による) |
| STANDBY | PDDS ビット + SLEEPDEEPビット + WFI または WFE | WKUP ピンの立ち上がり エッジ、RTC アラーム (アラーム A または アラーム B)、 RTCウェイクアップ イベント、RTC タンパ イベント、RTCタイム スタンプイベント、 NRST ピンによる外部 リセット、IWDG リセット | | | オフ |

6.3.1 低電力モードでのクロックの動作

APB ペリフェラルおよび DMA クロックは、ソフトウェアで無効にできます。

SLEEP および低電力 SLEEP モード

CPU クロックは、SLEEP および低電力 SLEEP モードで停止します。メモリアンタフェースクロック（フラッシュメモリおよび RAM インタフェース）およびすべてのペリフェラルクロックは、ソフトウェアによって SLEEP モード中に停止できます。低電力 SLEEP モードでは、メモリアンタフェースクロックは停止され、RAM はパワーダウン状態になります。AHB - APBブリッジのクロックは、SLEEP および低電力 SLEEP モード中にこれらに接続されたペリフェラルのクロックがすべて無効になった場合に、ハードウェアによって無効化されます。

STOP および STANDBY モード

STOP および STANDBY モードでは、システムクロックおよびすべての高速クロックは、停止します。

- PLL は無効化されます。
- 内部 RC 16 MHz (HSI16) オシレータは無効化されます。
- 外部 1-24 MHz (HSE) オシレータは無効化されます。
- 内部 65 kHz - 4.2 MHz (MSI) オシレータは無効化されます。

割り込みによってこのモードを終了する場合 (STOP モード)、内部 MSI または HSI16 をシステムクロックとして選択できます。両方のオシレータについて、それぞれの設定 (範囲およびトリミング) 値は STOP モードの終了時に保持されます。

リセットによってこのモードを終了する場合 (STANDBY モード)、システムクロックとして内部 MSI オシレータが選択されます。範囲とトリミング値は、デフォルトの 2.1 MHz にリセットされます。

フラッシュプログラム処理、または APB ドメインへのアクセスが進行中の場合、フラッシュメモリまたは APB へのアクセスが完了してから、STOP/STANDBY モードに移行します。

6.3.2 システムクロックの低速化

RUN モードでは、プリスケアラレジスタをプログラミングすることによって、システムクロック (SYSCLK、HCLK、PCLK1、PCLK2) の速度を下げるができます。SLEEP モードに移行する前にペリフェラルの速度を下げるため、これらのプリスケアラを使用することもできます。

詳細については、[セクション 7.3.4 : クロック設定レジスタ \(RCC_CFGR\)](#) を参照してください。

6.3.3 ペリフェラルクロックゲーティング

RUN モードでは、消費電力を低減するため、個々のペリフェラルとメモリへの HCLK および PCLKx をいつでも停止することができます。

SLEEP モードで消費電力をさらに低減するため、WFI または WFE 命令を実行する前に、ペリフェラルクロックを停止することができます。

ペリフェラルへのクロックゲーティングは、AHB ペリフェラルクロック有効レジスタ (RCC_AHBENR)、APB2 ペリフェラルクロック有効レジスタ (RCC_APB2ENR)、APB1 ペリフェラルクロック有効レジスタ (RCC_APB1ENR) によって制御されます ([セクション 7.3.13 : AHB ペリフェラルクロック有効レジスタ \(RCC_AHBENR\)](#)、[セクション 7.3.15 : APB1 ペリフェラルクロック有効レジスタ \(RCC_APB1ENR\)](#)、および [セクション 7.3.14 : APB2 ペリフェラルクロック有効レジスタ \(RCC_APB2ENR\)](#) を参照)。

RCC_AHBLPENR レジスタと RCC_APBxLPENR レジスタ (x は 1 または 2) の対応ビットをリセットすることで、SLEEP モード時のペリフェラルクロックを自動的に停止させることができます。

6.3.4 低電力 RUN モード (LP RUN)

システムが RUN モードの場合に消費電力を削減するために、レギュレータを低電力モードで設定することができます。このモードでは、システム周波数は f_{MSI} レンジ 1 を超えてはいけません。

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

注： *APB1 クロック周波数が RTC クロック周波数の 7 倍 ($7 \times \text{RTCCLK}$) 未満である場合に RTC カレンダーレジスタを読み出せるようにするには、カレンダー時間と日付レジスタをソフトウェアによって 2 回読み出す必要があります。*

RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。

V_{CORE} がレンジ 2 の場合のみ低電力 RUN モードに移行できます。さらに、低電力 RUN モードを選択しているときは、ダイナミック電圧スケーリングを使用してはいけません。低電力 RUN モードが選択されているときは、レギュレータが低電力モードに設定された STOP モードおよび STANDBY モードのみ使用可能です。

注： *低電力 RUN モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。*

低電力 RUN モードへの移行

低電力 RUN モードに移行するには、次の手順に従います。

- 各デジタル IP クロックは、RCC_APBxENR および RCC_AHBENR レジスタを使用して有効または無効にする必要があります。
- システムクロックの周波数は、 f_{MSI} レンジ 1 の周波数を超えないよう低減する必要があります。
- レギュレータは、ソフトウェアによって強制的に低電力モードに入ります。(LPRUN および LPDSR ビットがセットされる)。

低電力 RUN モードの終了

低電力 RUN モードを終了するには、次の手順に従います。

- レギュレータは、ソフトウェアによって強制的にメインレギュレータモードに入ります。
- 必要に応じてフラッシュメモリをオンにします。
- システムクロックの周波数を増加できます。

6.3.5 SLEEP モード

SLEEP モードへの移行

WFI (Wait For Interrupt) 命令または WFE (Wait for Event) 命令を実行することにより、SLEEP モードに移行します。SLEEP モードへの移行方法として、Cortex®-M0+ システム制御レジスタの SLEEPONEXIT ビットによって、2 つのオプションが選択できます。

- Sleep-now : SLEEPONEXIT ビットがクリアされている場合、MCU は、WFI または WFE 命令の実行直後に SLEEP モードに移行します。
- Sleep-on-exit : SLEEPONEXIT ビットがセットされている場合、MCU は、最も優先度の低い ISR の終了直後に SLEEP モードに移行します。

注： *SLEEP モードで、すべての I/O ピンは RUN モードと同じ状態を保持します。*

SLEEP モードへの移行方法の詳細については、表 25 : Sleep-now および表 26 : Sleep-on-exit を参照してください。



SLEEP モードの終了

WFI 命令によって SLEEP モードに移行した場合、ペリフェラルの割り込みがネスト化されたベクタ割り込みコントローラ (NVIC) によって認識されると、デバイスは SLEEP モードを終了することができます。

WFE 命令によって SLEEP モードに移行した場合、MCU はイベントの発生直後に SLEEP モードを終了します。ウェイクアップイベントは、次のいずれかによって生成できます。

- ペリフェラル制御レジスタで割り込みを有効にし、NVIC では有効にせず、Cortex®-M0+ システム制御レジスタの SEVONPEND ビットを有効にします。MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと (NVIC 割り込みクリアペンディングレジスタの) ペリフェラル NVIC IRQ チャンネルペンディングビットをクリアする必要があります。
- あるいは、外部または内部 EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、ペリフェラル割り込みペンディングビットや NVIC IRQ チャンネルペンディングビットをクリアする必要はありません。

このモードは、割り込みの入口と出口で余分な時間を必要としないため、ウェイクアップ時間が最も短くなります。

SLEEP モードの終了方法の詳細については、[表 25 : Sleep-now](#) および [表 26 : Sleep-on-exit](#) を参照してください。

表 25. Sleep-now

| Sleep-now モード | 説明 |
|---------------|---|
| モードへの移行 | 次の条件下での WFI (Wait for Interrupt) または WFE (Wait for Event) – SLEEPDEEP = 0 および – SLEEPONEXIT = 0 Cortex®-M0+ システム制御レジスタを参照してください。 |
| モードの終了 | WFI 命令を使用して移行した場合 : 割り込み : 参照先 : 表 45 : ベクタテーブル WFE 命令を使用して移行した場合 : ウェイクアップイベント : 参照先 : セクション 13.3.2 : ウェイクアップイベント管理 |
| ウェイクアップ遅延時間 | なし |

表 26. Sleep-on-exit

| Sleep-on-exit | 説明 |
|---------------|--|
| モードへの移行 | 次の条件下での WFI (Wait for Interrupt) – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 Cortex®-M0+ システム制御レジスタを参照してください。 |
| モードの終了 | 割り込み : 参照先 : 表 45 : ベクタテーブル |
| ウェイクアップ遅延時間 | なし |

6.3.6 低電力 SLEEP モード (LP SLEEP)

低電力 SLEEP モードへの移行

電圧レギュレータを低電力モードに設定し、WFI (Wait for Interrupt) またはWFE (Wait for Event) 命令を実行することにより、低電力 SLEEP モードに移行します。このモードでは、フラッシュメモリは使用できませんが、RAM メモリは使用できます。

このモードでは、システム周波数は f_{MSI} 範囲 1 を超えてはいけません。

電圧レギュレータとペリフェラルの動作条件の詳細については、製品データシートを参照してください。

V_{CORE} がレンジ 2 の場合のみ低電力 SLEEP モードに移行します。

注： **APB1 クロック周波数が RTC クロック周波数の 7 倍 ($7 \times RTCLK$) 未満である場合に RTC カレンダーレジスタを読み出せるようにするには、カレンダー時間と日付レジスタをソフトウェアによって 2 回読み出す必要があります。**

RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。

低電力 SLEEP モードへの移行方法として、Cortex[®]-M0+ システム制御レジスタの SLEEPONEXIT ビットによって、2 つのオプションが選択できます。

- Sleep-now : SLEEPONEXIT ビットがクリアされている場合、MCU は、WFI または WFE 命令の実行直後に SLEEP モードに移行します。
- Sleep-on-exit : SLEEPONEXIT ビットがセットされている場合、MCU は、最も優先度の低い ISR の終了直後に SLEEP モードに移行します。

低電力 SLEEP モードに移行するには、次の手順に従います。

- フラッシュメモリは、制御ビットを使用してオフにすることができます (FLASH_ACR レジスタの SLEEP_PD。詳細については、PM0062 を参照)。これにより、消費電力を削減できますがウェイクアップ時間は増えます。
- 各デジタル IP クロックは、RCC_APBxENR および RCC_AHBENR レジスタを使用して有効または無効にする必要があります。
- システムクロックの周波数を下げる必要があります。
- レギュレータは、ソフトウェアによって強制的に低電力モードに入ります (LPSSDR ビットがセットされる)。
- SLEEP モードに移行するには、WFI/WFE 命令を実行する必要があります。

注： **低電力 SLEEP モードで、I/O ピンはすべて RUN モードと同じ状態を保持します。**

低電力 SLEEP モードへの移行方法の詳細については、[表 27: Sleep-now](#) および [表 28: Sleep-on-exit](#) を参照してください。

低電力 SLEEP モードの終了

WFI 命令によって低電力 SLEEP モードに移行した場合、ペリフェラルの割り込みがネスト化されたベクタ割り込みコントローラ (NVIC) によって認識されると、デバイスは低電力 SLEEP モードを終了することができます。

WFE 命令によって低電力 SLEEP モードに移行した場合、MCU はイベントの発生直後に SLEEP モードを終了します。ウェイクアップイベントは、次のように生成できます。

- ペリフェラル制御レジスタで割り込みを有効にし、NVIC では有効にせず、Cortex®-M0+ システム制御レジスタの SEVONPEND ビットを有効にします。MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと NVIC 割り込みクリアペンディングレジスタのペリフェラル NVIC IRQ チャンネルペンディングビットをクリアする必要があります。
- あるいは、外部または内部 EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、ペリフェラル割り込みペンディングビットや NVIC IRQ チャンネルペンディングビットをクリアする必要はありません。

割り込みまたはウェイクアップイベントの発行によって低電力 SLEEP モードを終了する場合、レギュレータはメインレギュレータモードで設定し、フラッシュメモリを (必要に応じて) オンにし、システムクロックを増加することができます。

電圧レギュレータが低電力モードで動作している場合、低電力 SLEEP モードからのウェイクアップの際に、更なる起動時間が必要になります。

低電力 SLEEP モードの終了方法の詳細については、[表 27 : Sleep-now](#) および [表 28 : Sleep-on-exit](#) を参照してください。

表 27. Sleep-now

| Sleep-now モード | 説明 |
|---------------|--|
| モードへの移行 | 電圧レギュレータは低電力モード、フラッシュメモリはオフ 次の条件下での WFI (Wait for Interrupt) または WFE (Wait for Event) – SLEEPDEEP = 0 および – SLEEPONEXIT = 0 参照先 - Cortex®-M0+ システム制御レジスタ |
| モードの終了 | 電圧レギュレータはメインレギュレータモード、フラッシュメモリはオン WFI 命令を使用して移行した場合： 割り込み：参照先 - 表 45 : ベクタテーブル WFE 命令を使用して移行した場合： ウェイクアップイベント：参照先 - セクション 13.3.2 : ウェイクアップイベント管理 |
| ウェイクアップ遅延時間 | 低電力モードからのレギュレータのウェイクアップ時間 |

表 28. Sleep-on-exit

| Sleep-on-exit | 説明 |
|---------------|---|
| モードへの移行 | 電圧レギュレータは低電力モード、フラッシュメモリはオフ 次の条件下での WFI (Wait for Interrupt) – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 参照先 - Cortex®-M0+ システム制御レジスタ |

表 28. Sleep-on-exit (続き)

| Sleep-on-exit | 説明 |
|---------------|---------------------------|
| モードの終了 | 割り込み：参照先 - 表 45：ベクタテーブル |
| ウェイクアップ遅延時間 | 低電力モードからのレギュレータのウェイクアップ時間 |

6.3.7 STOP モード

STOP モードは、ペリフェラルクロックゲーティングと組み合わせられた Cortex®-M0+ のディープスリープ (deepsleep) モードに準拠しています。電圧レギュレータは、通常モードまたは低電力モードのいずれかに設定できます。STOP モードでは、 V_{CORE} ドメインのすべてのクロックが停止し、PLL、MSI、HSI16、および HSE RC オシレータが停止します。内蔵 SRAM とレジスタの内容は保持されます。

STOP モードで最小消費電力を達成するために、内部フラッシュメモリも低電力モードに移行します。フラッシュメモリが低電力モードで動作している場合、STOP モードからのウェイクアップの際に、さらなる起動時間が必要になります。

STOP モードの消費電力を最小化するために、STOP モードに移行する前に V_{REFINT} 、BOR、PVD、および温度センサをオフにすることができます。PWR_CR レジスタの ULP ビットを使用して STOP モードを終了した後で、ソフトウェアによって再度オンにすることができます。

注： STOP モードでは、すべての I/O ピンは RUN モードと同じ状態を保持します。

STOP モードへの移行

STOP モードへの移行方法の詳細については、表 29 を参照してください。

STOP モードに移行する前にアプリケーションが外部クロックを無効にする必要がある場合、最初に HSEON ビットを無効にし、次いでシステムクロックを HSI16 に切り替える必要があります。

そうしなければ、STOP モードに移行する前に外部クロック (外部オシレータ) が取り外される可能性があるにもかかわらず HSEON ビットが有効のままであった場合、STOP モードに移行するときに外部オシレータ障害を検出して誤動作を回避するべく、クロックセキュリティシステム (CSS) 機能を有効にする必要があります。

STOP モードで消費電力をさらに低減するには、内蔵電圧レギュレータを低電力モードにします。この設定は PWR_CR レジスタの LPSSDR ビットで行います (セクション 6.4.1 を参照)。

フラッシュメモリプログラムまたは APB ドメインへのアクセスが進行中の場合、メモリまたは APB アクセスが完了するまで、STOP モードへの移行は遅延します。

STOP モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ (IWDG)：IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。セクション 25.3：IWDG の機能説明のセクション 25：独立型ウォッチドッグ (IWDG) を参照してください。
- リアルタイムクロック (RTC)：この設定は RCC_CSR レジスタの RTCEN ビットで行います (セクション 7.3.21 を参照)。
- 内部 RC オシレータ (LSI RC)：この設定は RCC_CSR レジスタの LSION ビットで行います。
- 外部 32.768 kHz オシレータ (LSE OSC)：この設定は RCC_CSR レジスタの LSEON ビットで行います。

ADC、DAC、または LCD は、STOP モードに移行する前に停止させない限り、STOP モードでも電力を消費します。ADC および DAC を停止させるには、ADC_CR2 レジスタの ADON ビットと DAC_CR レジスタの ENx ビットの両方に 0 を書き込む必要があります。



STOP モードの終了

STOP モードの終了方法の詳細については、[表 29](#) を参照してください。

割り込みまたはウェイクアップイベントの発行によって STOP モードを終了する場合、RCC_CFGR レジスタの STOPWUCK ビットによって、システムクロックとして MSI または HSI16 RC オシレータが選択されます。

電圧レギュレータが低電力モードで動作している場合、STOP モードからのウェイクアップの際に、さらなる起動時間が必要になります。STOP モードの間も内蔵レギュレータをオン状態に保つことによって、消費電力は増加しますが、起動時間は短縮されます。

表 29. STOP モード

| STOP モード | 説明 |
|-------------|--|
| モードへの移行 | 次の条件下での WFI (Wait for Interrupt) または WFE (Wait for Event) <ul style="list-style-type: none">– Cortex®-M0+ システム制御レジスタの SLEEPDEEP ビットをセット– 電源制御レジスタ (PWR_CR) の PDDS ビットをクリア– 電源制御/ステータスレジスタ (PWR_CSR) の WUF ビットをクリア– RCC_CFGR レジスタの STOPWUCK ビットを設定して、STOP モードの終了のシステムクロックとして MSI または HSI16 RC オシレータを選択 注： STOP モードに移行するには、すべての EXTI ラインのペンディングビット (セクション 13.5.6 : EXTI ペンディングレジスタ (EXTI_PR) 内)、すべてのペリフェラル割り込みペンディングビット、RTC アラーム (アラーム A およびアラーム B)、RTC ウェイクアップ、RTC タンパ、RTC タイムスタンプフラグをリセットする必要があります。そうしないと、TOP モード移行手順が無視され、プログラムが実行され続けます。 |
| モードの終了 | WFI 命令を使用して移行した場合： 割り込みモードに設定されている任意の EXTI ライン (対応する EXTI 割り込みベクタが NVIC で有効になっている必要があります)。表 45 : ベクタテーブルを参照してください。 WFE 命令を使用して移行した場合： イベントモードに設定されている任意の EXTI ライン参照先: セクション 13.3.2: ウェイクアップイベント管理 (254 ページ) |
| ウェイクアップ遅延時間 | MSI または HSI16 RC ウェイクアップ時間 + 低電力モードからのレギュレータウェイクアップ時間 + FLASH ウェイクアップ時間 |

6.3.8 STANDBY モード

STANDBY モードでは、消費電力を最も少なくできます。このモードは、電圧レギュレータを無効にした状態の Cortex®-M0+ のデープスリープ (deepsleep) モードに基づきます。結果として、V_{CORE} ドメインの電源がオフになります。PLL、MSI、HSI16 オシレータ、HSE オシレータもオフになります。RTC レジスタ、RTC バックアップレジスタ、および STANDBY 回路を除いて、SRAM とレジスタの内容は失われます ([図 10](#) を参照)。

STANDBY モードへの移行

STANDBY モードへの移行方法の詳細については、表 30 を参照してください。

STANDBY モードでは、個別の制御ビットをプログラミングすることによって、次の機能を選択できます。

- 独立型ウォッチドッグ (IWDG) : IWDG は、キーレジスタへの書き込みによって、またはハードウェアオプションによって起動します。ウォッチドッグの動作がいったん開始されると、リセット以外では停止できません。[セクション 26.3 : WWDG の機能説明 \(568 ページ\)](#) を参照してください。
- リアルタイムクロック (RTC) : この設定は RCC_CSR レジスタの RTCEN ビットで行います ([セクション 7.3.21](#) を参照)。
- 内部 RC オシレータ (LSI RC) : この設定は RCC_CSR レジスタの LSION ビットで行います。
- 外部 32.768 kHz オシレータ (LSE OSC) : この設定は RCC_CSR レジスタの LSEON ビットで行います。

STANDBY モードの終了

マイクロコントローラは 外部リセット (NRST ピン)、IWDG リセット、WKUP ピン (WKUP1、WKUP2、または WKUP3) の立ち上がりエッジ、RTC アラーム、タンパイイベント、タイムスタンプイベントのいずれかが検出されると、STANDBY モードを終了します。STANDBY モードからのウェイクアップ後、[PWR 電力制御/ステータスレジスタ \(PWR_CSR\)](#) を除くすべてのレジスタがリセットされます。

STANDBY モードからのウェイクアップ後、プログラム実行はリセット後と同様にリスタートされます (ブートピン信号のサンプリング、ベクタリセットのフェッチなど)。PWR_CSR レジスタの SBF ステータスフラグ ([セクション 6.4.2](#) を参照) は、MCU が STANDBY モードにあったことを示します。

STANDBY モードの終了方法の詳細については、表 30 を参照してください。

表 30. STANDBY モード

| STANDBY モード | 説明 |
|-------------|---|
| モードへの移行 | 次の条件下での WFI (Wait for Interrupt) または WFE (Wait for Event) <ul style="list-style-type: none">– Cortex®-M0+ システム制御レジスタの SLEEPDEEP をセット– 電源制御レジスタ (PWR_CR) の PDDS ビットをセット– 電源制御/ステータスレジスタ (PWR_CSR) の WUF ビットをクリア– 選択されたウェイクアップソース (RTC アラーム A、RTC アラーム B、RTC ウェイクアップ、タンパ、タイムスタンプフラグ) に対応する RTC フラグをクリア |
| モードの終了 | WKUP ピンの立ち上がりエッジ、RTC アラーム (アラーム A およびアラーム B)、RTC ウェイクアップイベント、タンパイイベント、タイムスタンプイベント、NRST ピンによる外部リセット、IWDG リセット |
| ウェイクアップ遅延時間 | リセットフェーズ |

STANDBY モードにおける I/O の状態

STANDBY モードでは、以下のピンを除き、すべての I/O ピンがハイインピーダンス状態となります。

- リセットパッド (有効)
- ウェイクアップピン 2 (WKUP2)、タンパ、タイムスタンプ、RTC アラーム出力、または RTC クロック較正出力として設定されている場合のピン (PC13)
- WKUP ピン 1 (PA0) (有効な場合)

デバッグモード

デフォルトでは、デバッグ機能が使用されているときにアプリケーションが MCU を STOP または STANDBY モードにすると、デバッグ接続は失われます。これは、Cortex®-M0+ コアにクロックが供給されなくなるためです。

ただし、DBG_CR レジスタの設定ビットをセットすることで、低電力モードを多用しているときでも、ソフトウェアのデバッグを行うことができます。詳細については、[セクション 33.9.1：低電力モードのデバッグサポート](#)を参照してください。

6.3.9 RTC およびコンパレータを使用した STOP モードおよび STANDBY モードからデバイスのウェイクアップ

MCU は、外部割り込み（自動ウェイクアップモード）に頼ることなく、RTC アラームイベント、RTC ウェイクアップイベント、タンパイイベント、タイムスタンプイベント、コンパレータイイベントのいずれかによって、低電力モードからウェイクアップさせることができます。

これらの RTC オルタネート機能によって、システムを低電力の STOP モードおよび STANDBY モードからウェイクアップさせることができます。一方コンパレータイイベントではシステムを STOP モードからのみウェイクアップさせることができます。

システムはまた、外部割り込み（自動ウェイクアップモード）に頼ることなく、RTC アラームや RTC ウェイクアップイベントを使用して低電力モードからウェイクアップさせることもできます。

RTC を、一定の時間間隔で STOP モードや STANDBY モードからウェイクアップさせるためのプログラム可能なタイムベースとすることができます。この目的のため、RCC_CSR レジスタの RTCSEL[1:0] ビットをプログラミングすることによって、3 つの代替 RTC クロックソースのうちから 2 つを選択できます（[セクション 7.3.21](#) を参照）。

- 低電力 32.768 kHz 外部クリスタルオシレータ（LSE OSC）
このクロックソースは、非常に少ない消費電力（標準的な条件下で追加消費電力 1 μ A 未満）で高精度のタイムベースとなります。
- 低電力内部 RC オシレータ（LSI RC）
このクロックソースには、32.768 kHz クリスタルのコストを節約できるという利点があります。この内部 RC オシレータは、最小限の消費電力で動作するように設計されています。

STOP モードからの RTC 自動ウェイクアップ (AWU)

- RTC アラームイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) EXTI ライン 17 を立ち上がりエッジを検知するように設定します (割り込みモードまたはイベントモード)。
 - b) RTC_CR レジスタ内の RTC アラーム割り込みを有効にします。
 - c) RTC が RTC アラームを生成するように設定します。
- RTC タンパまたはタイムスタンプイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) EXTI ライン 19 を立ち上がりエッジを検知するように設定します (割り込みモードまたはイベントモード)。
 - b) RTC_CR レジスタの RTC タイムスタンプ割り込みまたは RTC_TCR レジスタの RTC タンパ割り込みを有効にします。
 - c) タンパまたはタイムスタンプイベントを検出するよう RTC を設定します。
- RTC ウェイクアップイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) EXTI ライン 20 を立ち上がりエッジを検知するように設定します (割り込みモードまたはイベントモード)。
 - b) RTC_CR レジスタ内の RTC ウェイクアップ割り込みを有効にします。
 - c) RTC ウェイクアップイベントを生成するように RTC を設定します。

STANDBY モードからの RTC 自動ウェイクアップ (AWU)

- RTC アラームイベントによって STANDBY モードからウェイクアップさせるには、次の設定が必要です。
 - a) RTC_CR レジスタ内の RTC アラーム割り込みを有効にします。
 - b) RTC が RTC アラームを生成するように設定します。
- RTC タンパまたはタイムスタンプイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) RTC_CR レジスタの RTC タイムスタンプ割り込みまたは RTC_TCR レジスタの RTC タンパ割り込みを有効にします。
 - b) タンパまたはタイムスタンプイベントを検出するよう RTC を設定します。
- RTC ウェイクアップイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) RTC_CR レジスタ内の RTC ウェイクアップ割り込みを有効にします。
 - b) RTC ウェイクアップイベントを生成するように RTC を設定します。

STOP モードからのコンパレータ自動ウェイクアップ (AWU)

- コンパレータ 1 またはコンパレータ 2 のウェイクアップイベントによって STOP モードからウェイクアップさせるには、次の設定が必要です。
 - a) コンパレータ 1 の EXTI ライン 21 またはコンパレータ 2 の EXTI ライン 22 (割り込みまたはイベントモード) が、選択したエッジ (立ち上がり、立ち下がり、または両方) を検知するように設定します。
 - b) コンパレータがイベントを生成するように設定します。

6.4 電源制御レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

6.4.1 PWR 電源制御レジスタ (PWR_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 1000 (STANDBY モードからのウェイクアップによってリセット)

| | | | | | | | | | | | | | | | |
|------|-------|------------|----------|------|------|------|------|----------|------|------|------|-------|-------|------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | LPRUN | DS_EE_KOFF | VOS[1:0] | | FWU | ULP | DBP | PLS[2:0] | | | PVDE | CSBF | CWUF | PDDS | LPSDSR |
| | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rC_w1 | rC_w1 | rW | rW |

ビット 31:15 予約済み、常に 0 として読み出されます。

ビット 14 **LPRUN** : 低電力 RUN モード

LPRUN ビットが LPSDSR ビットと一緒にセットされると、レギュレータはメインモードから低電力モードに切り替わります。それ以外の場合は、メインモードが維持されます。LPRUN をリセットすると、レギュレータの動作はメインモードに戻ります。

MCU が低電力 RUN モードの場合は、LPSDSR をリセットすることは禁止されています。LPSDSR は、低電力モードへの移行に対する事前措置として使用され、これによって低電力モードへの移行時に選択されるレギュレータの設定をシステムに示します。LPRUN ビットの前に LPSDSR ビットをセットする必要があります。LPSDSR は、LPRUN ビット = 0 の場合のみリセットできます。

0 : 低電力 RUN モードにて電圧レギュレータはメインモード

1 : 低電力 RUN モードにて電圧レギュレータは低電力モード

ビット 13 : **DS_EE_KOFF** : ディープスリープモードではフラッシュメモリはオフ状態に維持されます。

低電力モード (STOP または STANDBY) に移行する場合、DS_EE_KOFF および FLASH_ACR レジスタの RUN_PD を両方ともセットすると ([セクション 3.7.1 : アクセス制御レジスタ \(FLASH_ACR\)](#) を参照)、ディープスリープモードの終了時にフラッシュメモリはウェイクアップされません。

0 : RUN_PD がセットされている場合でも、ディープスリープモードの終了時にフラッシュメモリはウェイクアップされます。

1 : 低電力モードの終了時に EEPROM はウェイクアップされません (RUN_PD ビットがセットされている場合)。

ビット 12:11 **VOS[1:0]** : 電圧スケーリングレンジの選択

このビットは、内部レギュレータの電圧範囲を選択するために使用します。

RCC_APB1RSTR レジスタの PWRRST ビットをリセットして電源インタフェースをリセットする前に、このビットを「10」にセットし、それに応じてシステムの周波数を設定する必要があります。

00 : 使用禁止 (ビットは変更されずに前の値が維持され、電圧は変更されません)

01 : 1.8 V (レンジ 1)

10 : 1.5 V (レンジ 2)

11 : 1.2 V (レンジ 3)

ビット 10 **FWU** : 高速ウェイクアップ

このビットは、ULP ビットとともに動作します。

ULP = 0 の場合、FWU は無視されます。

ULP = 1 および FWU = 1 の場合：低電力モードの終了時に V_{REFINT} の起動時間は無視されます。PWR_CSR レジスタの VREFINTRDYF フラグは、再度 V_{REFINT} がレディ状態になったことを示します

ULP = 1 および FWU = 0 の場合：低電力モードは、 V_{REFINT} がレディ状態になった場合にのみ終了します（起動時間後）。このビットは、RCC_APB1RSTR レジスタの PWRRST ビットをリセットしてもしリセットされません。

0 : 低電力モードは、 V_{REFINT} がレディ状態になった場合にのみ終了します。

1 : 低電力モードの終了時に V_{REFINT} の起動時間は無視されます。

ビット 9 **ULP** : 超低電力モード

セットすると、 V_{REFINT} は低電力モードでオフになります。このビットは、RCC_APB1RSTR レジスタの PWRRST ビットをリセットしてもしリセットされません。

0 : V_{REFINT} は低電力モードでオンになります。

1 : V_{REFINT} は低電力モードでオフになります。

ビット 8 **DBP** : バックアップ書き込み保護の無効化

リセット状態で、RTC、RTC バックアップレジスタ、および RCC_CSR レジスタは不要な書き込みアクセスから保護されます。これらのレジスタへの書き込みアクセスを可能にするには、このビットをセットする必要があります。

0 : RTC、RTC バックアップ、および RCC_CSR レジスタへのアクセスは無効です。

1 : RTC、RTC バックアップ、および RCC_CSR レジスタへのアクセスは有効です。

注： 2、4、8、または 16 で分周した HSE が RTC クロックとして使用される場合、このビットは 1 にセットした状態を維持する必要があります。

ビット 7:5 **PLS[2:0]** : PVD レベル選択

これらのビットは、電源電圧検出器によって検出される電圧閾値を選択するために、ソフトウェアで書き込みます。

000 : 1.9 V

001 : 2.1 V

010 : 2.3 V

011 : 2.5 V

100 : 2.7 V

101 : 2.9 V

110 : 3.1 V

111 : 外部入力アナログ電圧（内部的に V_{REFINT} と比較されます）

PVD_IN 入力 (PB7) は、PLS[2:0] = 111 の場合、アナログ入力として設定する必要があります。

注： 詳細については、データシートの電気的特性を参照してください。

ビット 4 **PVDE** : 電源電圧検出器有効化

このビットは、ソフトウェアによってセット/クリアされます。

0 : PVD は無効です。

1 : PVD は有効です。

ビット 3 **CSBF** : STANDBY フラグのクリア

このビットは常に 0 として読み出されます。

0 : 影響なし

1 : SBF STANDBY フラグのクリア（書き込み）

ビット 2 **CWUF** : ウェイクアップフラグのクリア

このビットは常に 0 として読み出されます。

0 : 影響なし

1 : WUF ウェイクアップフラグをシステムクロックの 2 サイクル後にクリア

ビット 1 **PDDS** : パワーダウディープスリープ

このビットは、ソフトウェアによってセット／クリアされます。

0 : CPU がディープスリープに移行すると、STOP モードに移行します。レギュレータは低電力モードです。

1 : CPU がディープスリープに移行すると、STANDBY モードに移行します。

ビット 0 **LPSDSR** : 低電力ディープスリープ／SLEEP／低電力 RUN

– ディープスリープ／SLEEP モード

このビットをセットすると、CPU が SLEEP またはディープスリープモードへ移行する時に、レギュレータは低電力モードに切り替わります。CPU がこれらのモードを終了すると、レギュレータはメインモードに戻ります。

– 低電力 RUN モード

このビットをセットすると、LPRUN ビットのセット時に、レギュレータは低電力モードに切り替わります。LPRUN をリセットすると、レギュレータはメインモードに戻ります。

このビットは、ソフトウェアによってセット／クリアされます。

0 : 電圧レギュレータは、ディープスリープ／SLEEP／低電力 RUN モードの間、オン状態を保ちます。

1 : 電圧レギュレータは、ディープスリープ／SLEEP／低電力 RUN モードの間、低電力モードを保ちます。

6.4.2 PWR 電源制御/ステータスレジスタ (PWR_PWR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0008 (STANDBY モードからのウェイクアップによってリセットされない)

このレジスタを読み出すには、通常の APB 読み出しに対して、追加の APB サイクルが必要です。

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|------|------|------|------|-------|-------|------|------|---------|------|-------------|------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | EWUP2 | EWUP1 | Res. | Res. | REG LPF | VOSF | VREINT RDYF | PVDO | SBF | WUF |
| | | | | | | rw | rw | | | r | r | r | r | r | r |

ビット 31:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **EWUP2** : WKUP ピン 2 を有効にします。

このビットは、ソフトウェアによってセット/クリアされます。

0 : WKUP ピン 2 は汎用 I/O として使用されます。この WKUP ピン 2 でイベントが発生しても、デバイスは STANDBY モードからウェイクアップしません。

1 : WKUP ピン2は STANDBY モードからのウェイクアップに使用され、強制的にプルダウン入力モードに設定されます (WKUP ピン 2 の立ち上がりエッジで、システムは STANDBY モードからウェイクアップします)。

注: このビットは、システムリセットによってリセットされます。

ビット 8 **EWUP1** : WKUP ピン 1 を有効にします。

このビットは、ソフトウェアによってセット/クリアされます。

0 : WKUP ピン 1 は汎用 I/O として使用されます。この WKUP ピン 1 でイベントが発生しても、デバイスは STANDBY モードからウェイクアップしません。

1 : WKUP ピン1は STANDBY モードからのウェイクアップに使用され、強制的にプルダウン入力モードに設定されます (WKUP ピン 1 の立ち上がりエッジで、システムは STANDBY モードからウェイクアップします)。

注: このビットは、システムリセットによってリセットされます。

ビット 7:6 予約済みであり、リセット値のままにしておかなければなりません。

ビット 5 **REGLPF** : レギュレータ LP フラグ

このビットは、MCU が低電力 RUN モードである場合に、ハードウェアによってセットされます。

MCU が低電力 RUN モードを終了すると、レギュレータがメインモードでレディ状態になるまで、このビットは 1 の状態を維持します。レギュレータがメインモードになるのを待つために、このビットのポーリングを推奨します。このビットは、レギュレータがレディ状態になると、ハードウェアによってリセットされます。

0 : レギュレータはメインモードでレディ状態です。

1 : レギュレータ電圧は低電力モードです。

ビット 4 **VOSF** : 電圧スケーリング選択フラグ

電圧レンジ変更後に内蔵レギュレータがレディ状態になるには、遅延が必要です。VOSF ビットは、レギュレータが PWR_CR レジスタの VOS ビットで定義された電圧レベルに達したことを示します。

このビットは、PWR_CR レジスタの VOS[1:0] が変更されるとリセットされます。

レギュレータがレディ状態になると、セットされます。

0 : レギュレータは選択された電圧レンジでレディ状態になります。

1 : レギュレータ電圧出力は、必要な VOS レベルに変化します。

- ビット 3 **VREFINTRDYF** : 内部基準電圧 (V_{REFINT}) レディフラグ
このビットは、内部基準電圧 (V_{REFINT}) の状態を示します。
0 : V_{REFINT} はオフです。
1 : V_{REFINT} はレディ状態です。
- ビット 2 **PVDO** : PVD 出力
このビットは、ハードウェアによってセット/クリアされます。PVDE ビットによって PVD が有効化されている場合のみ有効です。
0 : V_{DD} は PLS[2:0] ビットで選択された PVD 閾値より高いです。
1 : V_{DD} は PLS[2:0] ビットで選択された PVD 閾値より低いです。
注: **PVD は STANDBY モードによって停止します。したがって、このビットは STANDBY またはリセット後、PVDE ビットがセットされるまで 0 になります。**
- ビット 1 **SBF** : STANDBY フラグ
このビットは、ハードウェアによってセットされ、POR/PDR (パワーオンリセット/パワーダウンリセット)、または **PWR 電源制御レジスタ (PWR_CR)** の CSBF ビットをセットすることでのみクリアされます。
0 : デバイスは STANDBY モードではありません。
1 : デバイスは STANDBY モードです。
- ビット 0 **WUF** : ウェイクアップフラグ
このビットは、ハードウェアによってセットされ、システムリセットまたは **PWR 電源制御レジスタ (PWR_CR)** の CWUF ビットをセットすることでクリアされます。
0 : ウェイクアップイベントは発生していません。
1 : WKUP ピン、RTC アラーム (アラーム A またはアラーム B)、RTC タンパイイベント、RTC タイムスタンプイベント、RTC ウェイクアップのいずれかよりウェイクアップイベントを受信しています。
注: **WKUP ピンのレベルがすでにハイであっても、(EWUPx (x=1, 2, 3) ビットをセットすることで) WKUP ピンが有効化されている場合、追加のウェイクアップイベントが検出されます。**

6.4.3 PWR レジスタマップ

次の表に PWR レジスタの一覧を示します。

表 31. PWR - レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|---------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|------------|--------------|------|-------|-------|----------|------|--------|------|-------------|------|------|-------|---|
| 0x000 | PWR_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | LPRUN | DS_EE_KOFF | VOS [1:0] | FWU | ULP | DBP | PLS[2:0] | | | PVDE | CSBF | CWUF | PDDS | LPDSR | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 1 | | | | | | | | | | | | | | 0 |
| 0x004 | PWR_CSR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EWUP2 | EWUP1 | Res. | Res. | REGLPF | VOSF | VREFINTRDYF | PVDO | SBF | WUF | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | 0 | 0 | 1 | 0 | 0 | 0 | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。



7 リセットおよびクロック制御 (RCC)

7.1 リセット

リセットには、システムリセット、電源リセット、RTC ドメインリセットの 3 種類があります。

7.1.1 システムリセット

システムリセットでは、RTC、RTC バックアップレジスタ、および制御/ステータスレジスタ (RCC_CR および RCC_CSR) を除く、すべてのレジスタをリセット値にセットします。

システムリセットは、次のイベントのいずれかの発生時に生成されます。

- NRST ピンのローレベル (外部リセット)
- ウィンドウ型ウォッチドッグのカウント終了条件 (WWDG リセット)
- 独立型ウォッチドッグのカウント終了条件 (IWDG リセット)
- ソフトウェアリセット (SW リセット) ([ソフトウェアリセット](#)を参照)
- 低電力管理リセット ([低電力管理リセット](#)を参照)
- オプションバイトローダリセット ([オプションバイトローダリセット](#)を参照)
- STANDBY モードの終了
- ファイアウォールによる保護 ([セクション 5 : ファイアウォール \(FW\)](#) を参照)

リセットソースは、クロック制御/ステータスレジスタ (RCC_CSR) のリセットフラグを確認することによって識別できます ([セクション 7.3.21](#)を参照)。

ソフトウェアリセット

デバイス上でソフトウェアリセットを実行するには、Cortex®-M0+ AIRCR レジスタ (アプリケーション割り込みおよびリセット制御レジスタ) の SYSRESETREQ ビットをセットする必要があります。詳細については、ARM Cortex®-M0+ テクニカルリファレンスマニュアルを参照してください。

低電力管理リセット

低電力管理リセットを生成するには、2 つの方法があります。

- STANDBY モードに移行するときにリセットを生成します :
この種類のリセットは、ユーザオプションバイトの nRST_STDBY ビットをリセットすることで有効になります。この場合、STANDBY モードへの遷移シーケンスが正常に実行されるたびに、STANDBY モードに移行する代わりにデバイスがリセットされます。
- STOP モードに移行するときにリセットを生成します :
この種類のリセットは、ユーザオプションバイトの nRST_STOP ビットをリセットすることで有効になります。この場合、STOP モードへの遷移シーケンスが正常に実行されるたびに、STOP モードに移行する代わりにデバイスがリセットされます。

オプションバイトローダリセット

オプションバイトローダリセットは、FLASH_PECR レジスタの OBL_LAUNCH ビット (ビット 18) がセットされると生成されます。このビットは、ソフトウェアによってオプションバイトローディングを起動するために使用されます。

ユーザオプションバイトの詳細については、[セクション 3 : フラッシュプログラムメモリとデータ EEPROM \(FLASH\)](#) を参照してください。

7.1.2 電源リセット

電源リセットは、次のいずれかのイベントが発生したときに生成されます。

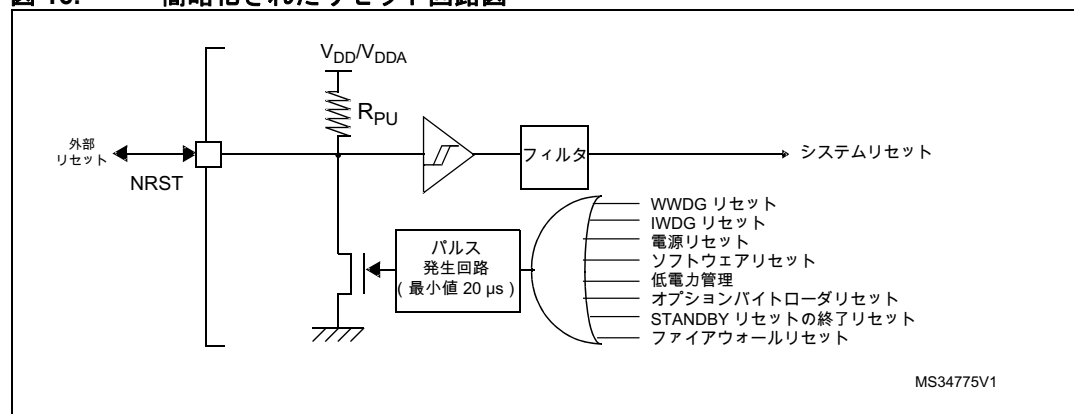
- パワーオン／パワーダウンリセット (POR/PDR リセット)
- BOR リセット

電源リセットは、RTC ドメインを含むすべてのレジスタをリセット値にセットします (図 16 を参照)。

これらのソースは NRST ピンに作用し、遅延フェーズの間、ピンをローに保ちます。リセットサービスルーチンベクタは、メモリマップのアドレス 0x0000_0004 に固定されています。詳細については、表 45 : ベクタテーブルを参照してください。

デバイスに与えられるシステムリセット信号は、NRST ピンに出力されます。パルス発生回路は、各内部リセットソースについて 20 μ s の最小リセットパルス期間を保証します。外部リセットの場合、リセットパルスは NRST ピンがローにアサートされているときに生成されます。

図 16. 簡略化されたリセット回路図



7.1.3 RTC およびバックアップレジスタリセット

RTC ペリフェラル、RTC クロックソース選択 (RCC_CSR)、およびバックアップレジスタは、次のいずれかのイベントが発生したときのみリセットされます。

- RCC_CSR レジスタの RTCRST ビットをセットすることでトリガされるソフトウェアリセット (セクション 7.3.21 を参照)
- 電源リセット (BOR/POR/PDR)

7.2 クロック

次の 4 つのクロックソースを使用して、システムクロック (SYSCLK) を駆動できます：

- HSI16 (高速内部) オシレータクロック
- HSE (高速外部) オシレータクロック
- PLL クロック
- MSI (マルチスピード内部) オシレータクロック

電源リセットの起動後、システムまたは RTC ドメインリセット、および STANDBY モードからのウェイクアップ後に、システムクロックソースとして 2.1 MHz の MSI が使用されます。

STOP モードからウェイクアップさせるために、HSI16、4 分周した HSI16、または使用可能な周波数に設定された MSI を使用できます。

デバイスには、2 つの 2 次クロックソースがあります。

- 37 kHz 低速内部 RC (LSI RC)。独立型ウォッチドッグを駆動し、オプションで、STOP/STANDBY モード、LCD、および LPTIMER からの自動ウェイクアップに使用される RTC を駆動します。
- 32.768 kHz 低速外部クリスタル (LSE クリスタル)。オプションで、リアルタイムクロック (RTCCLK)、LPTIMER、LCD、および UART を駆動します。

それぞれのクロックソースは、使用しないときに個別にオン/オフを切り替えて、電力消費を最適化可能です。

複数のプリスケアラを使用して AHB 周波数と 2 つの APB (APB1 および APB2) ドメインを設定できます。AHB、APB1、および APB2 ドメインの最大周波数は 32 MHz です。デバイスの電圧範囲に依存します。詳細については、[セクション 6.1.5：ダイナミック電圧スケーリングの管理](#) を参照してください。

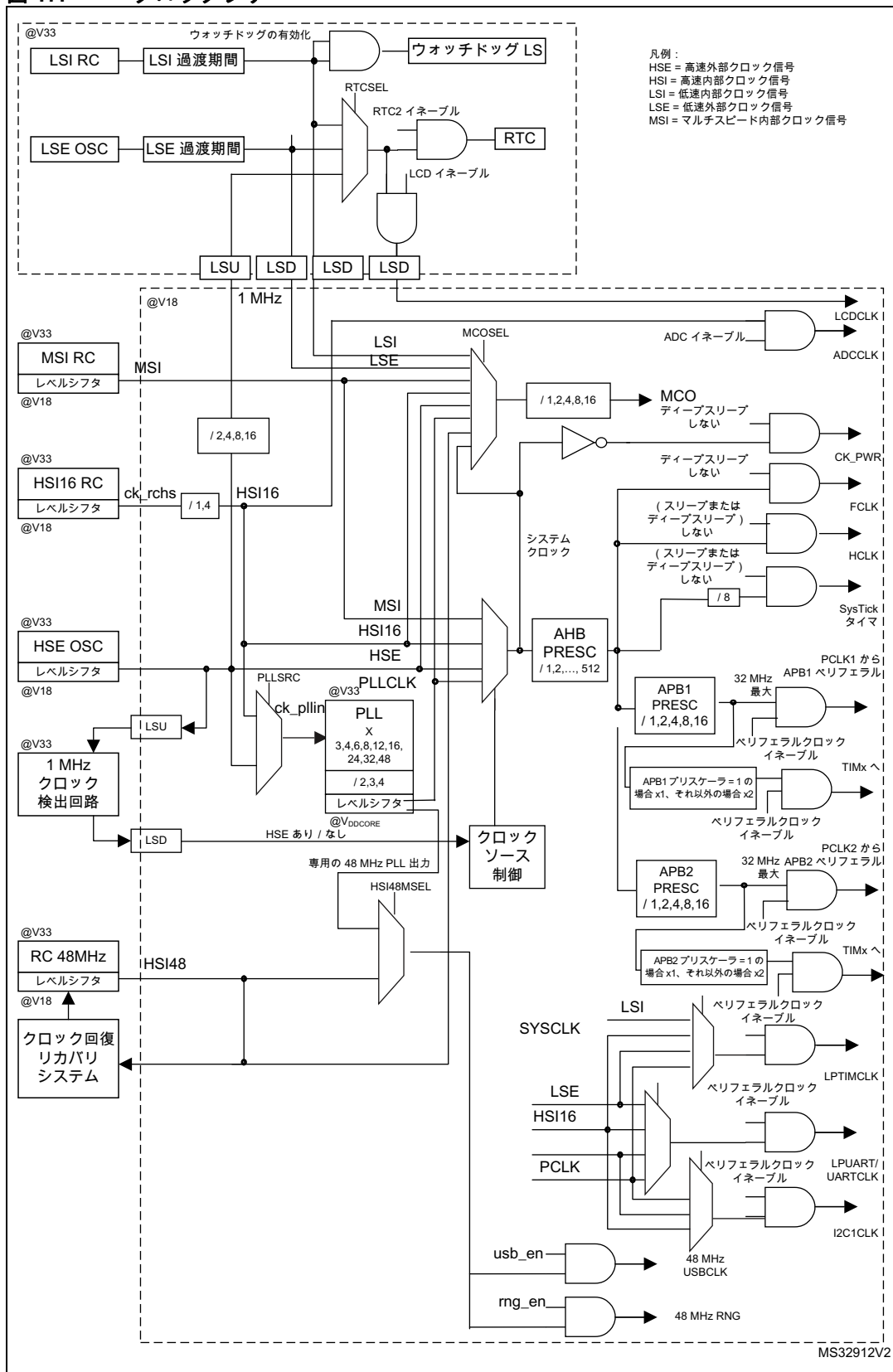
すべてのペリフェラルクロックは、システムクロック (SYSCLK) から生成されます。ただし、次のものを除きます。

- 次の2つのうちのいずれかのソースから生成される 48 MHz USB および RNG クロック
 - PLL VCO クロック
 - RC48 クロック (HSI48)
- クロック周波数をデバイスの動作条件に合わせるため、常に HSI16 クロックまたは 4 分周した HSI16 である ADC クロック詳細については、[セクション 6.1 : 電源](#) を参照してください。
- 次の4つのうちのいずれかのソースから生成される LPUART1 および USART1/2 クロック (ソフトウェアによって選択される)
 - システムクロック
 - HSI16 クロック
 - LSE クロック
 - APB クロック (PCLK)
- 次の3つのうちのいずれかのソースから生成される I2C1 クロック (ソフトウェアによって選択される)
 - システムクロック
 - HSI16 クロック
 - APB クロック (PCLK)
- 次の3つのうちのいずれかのソースから生成される LPTIMER クロック (ソフトウェアによって選択される)
 - HSI16 クロック
 - LSE クロック
 - LSI クロック
 - APB クロック (PCLK)
- 次のいずれかのソースから生成される RTC/LCD クロック
 - LSE クロック
 - LSI クロック
 - 1 MHz HSE_RTC (HSE をプログラム可能なプリスケアラで分周)
- 常に LSI クロックである IWDG クロック

システムクロック (SYSCLK) 周波数は、RTC/LCD クロック周波数より高いか、同じである必要があります。

RCC は Cortex システムタイマ (SysTick) の外部クロックに、8 分周した AHB クロック (HCLK) を供給します。SysTick は、この分周されたクロックか Cortex クロック (HCLK) のいずれかで動作でき、SysTick 制御およびステータスレジスタで設定可能です。

図 17. クロックツリー

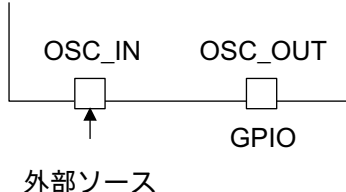
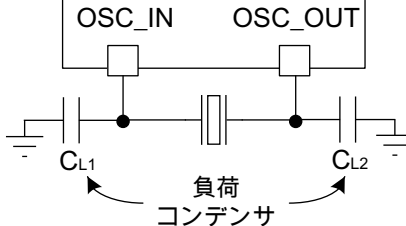


1. 内部および外部クロックソース特性の詳細については、デバイスのデータシートの「電気的特性」のセクションを参照してください。
- タイマクロック周波数は、ハードウェアによって自動的に固定されます。これには、2つのケースがあります。
1. APB プリスケーラが 1 の場合、タイマクロック周波数はタイマが接続されている APB ドメインのクロックと同じ周波数に設定されます。
2. そうでない場合は、タイマが接続されている APB ドメインの周波数の 2 倍に設定されます。
- f_{CLK} は、Cortex®-M0+ のフリーランニングクロックとして動作します。詳細については、[セクション 33：デバッグサポート \(DBG\)](#) を参照してください。

7.2.1 HSE クロック

- 高速外部クロック信号（HSE）は、次のどちらかのクロックソースから生成できます。
- HSE 外部クリスタル／セラミック発振子
 - HSE ユーザ外部クロック
- 波形ひずみと発振開始時の安定化までの時間を少なくするために、発振子と負荷コンデンサはオシレータのピンのできるだけ近くに配置する必要があります。負荷コンデンサの値は、選択したオシレータに応じて調整する必要があります。

図 18. HSE/LSE クロックソース

| クロックソース | ハードウェア構成 |
|----------------|--|
| 外部クロック |  <p>MSv31915V1</p> |
| クリスタル／セラミック発振子 |  <p>MSv31916V1</p> |

外部ソース (HSE バイパス)

このモードでは、外部クロックソースが必要です。最大 32 MHz までの周波数を使用できます。このモードを選択するには、RCC_CR レジスタの HSEBYP および HSEON ビットをセットします ([セクション 7.3.1: クロック制御レジスタ \(RCC_CR\) を参照](#))。約50%のデューティサイクルを持つ外部クロック信号で OSC_IN ピンを駆動する必要があり、その間、OSC_OUT ピンはハイインピーダンスに保たれます ([図 18](#) を参照)。外部クロック信号は、矩形波、正弦波、三角波のいずれかです。消費電力を最小化するには、矩形波信号を使用することを推奨します。

外部クリスタル／セラミック発振子 (HSE クリスタル)

1 から 24 MHz の外部オシレータには、メインクロックの周波数を非常に高い精度で生成できるという利点があります。

関連するハードウェア構成を [図 18](#) に示します。詳細については、データシートの電気的特性のセクションを参照してください。

RCC_CR レジスタの HSERDY フラグ ([セクション 7.3.1 を参照](#)) は、HSE オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSE クロック信号は出力されません。[RCC_CR レジスタ](#)で有効になっていれば、割り込みを生成することができます。

HSE クリスタルは、[RCC_CR レジスタ](#)の HSEON ビットを使用してオン／オフできます。

7.2.2 HSI16 クロック

HSI16 クロック信号は、内部 16 MHz RC オシレータから生成されます。直接システムクロックとして、または PLL 入力として使用できます。

HSI16 クロックは 低電力のSTOPモードからのウェイクアップ後に使用でき、これによって MSI クロックを使用したウェイクアップより短いウェイクアップ時間を実現できます。

HSI16 RC オシレータには、低コスト (外付部品なし) でクロックソースを供給できるという利点があります。同時に、HSE クリスタルオシレータよりも起動時間を短縮できますが、校正を実施しても、外部クリスタルオシレータやセラミック発振子よりも周波数の精度は劣ります。

校正

RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、周辺温度 (T_A) 25 °C で 1% の精度を確保するよう、製造時に ST においてデバイスごとの校正が行われています。

リセット後、工場較正値が内部クロックソース較正レジスタ (RCC_ICSCR) の HSI16CAL[7:0] ビットにロードされます ([セクション 7.3.2 を参照](#))。

アプリケーションが電圧または温度の変動を受ける場合、これが RC オシレータの速度に影響を与えることがあります。RCC_ICSCR レジスタの HSI16TRIM[4:0] ビットを使用して、アプリケーションで HSI16 周波数をトリミングできます。HSI16 周波数変動の測定方法の詳細については、[セクション 7.2.15: TIM21 を使用した内部／外部クロックの測定](#)を参照してください。

RCC_CR レジスタの HSI16RDY フラグは、HSI16 オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI16 RC 出力クロックは出力されません。

HSI16 RC オシレータは、RCC_CR レジスタの HSI16ON ビットを使用してオン／オフできます。

7.2.3 MSI クロック

MSI クロック信号は、内部 RC オシレータから生成されます。周波数範囲は、RCC_ICSCR レジスタの MSIRANGE[2:0] ビットを使用して、ソフトウェアによって調整することができます（[セクション 7.3.2：内部クロックソース校正レジスタ \(RCC_ICSCR\)](#) を参照）。7 種類の周波数範囲：65.536 kHz、131.072 kHz、262.144 kHz、524.288 kHz、1.048 MHz、2.097 MHz（デフォルト値）、および 4.194 MHz）を使用できます。

MSI クロックは、リセットからのリスタート後、および STANDBY モードからのウェイクアップ後に、常にシステムクロックとして使用されます。STOP モードからのウェイクアップ後に、HSI16（または HSI16/4）の代わりに MSI クロックをシステムクロックとして選択できます。

リセットまたは STANDBY モードからのウェイクアップ後にデバイスをリスタートした場合、MSI 周波数はデフォルト値にセットされます。MSI 周波数は、STOP モードからのウェイクアップ後に変更されることはありません。

MSI RC オシレータには、低コスト（外付部品なし）で低電力クロックソースを供給できるという利点があります。消費電力を低減するために、低電力モードのウェイクアップクロックとして使用されます。

RCC_CR レジスタの MSIRDY フラグは、MSI RC が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、MSI RC 出力クロックは出力されません。

MSI RC は、RCC_CR レジスタの MSION ビットを使用してオン/オフできます（[セクション 7.3.1](#) を参照）。

これは、HSE クリスタルオシレータに障害がある場合のバックアップクロックソース（補助クロック）としても使用できます。[セクション 7.2.10：HSE クロックセキュリティシステム \(CSS\) \(153 ページ\)](#) を参照してください。

校正

MSI RC オシレータの周波数は、製造工程でのばらつきのため、チップごとに異なります。このため、周辺温度 (T_A) 25 °C で 1% の精度を確保するよう、製造時に ST においてデバイスごとの校正が行われています。

リセット後、工場校正値が RCC_ICSCR レジスタの MSICAL[7:0] ビットにロードされます。アプリケーションが電圧または温度の変動を受ける場合、これが RC オシレータの速度に影響を与えることがあります。RCC_ICSCR レジスタの MSITRIM[7:0] ビットを使用して、アプリケーションで MSI 周波数をトリミングできます。MSI 周波数変動の測定方法の詳細については、[セクション 7.2.15：TIM21 を使用した内部／外部クロックの測定](#) を参照してください。

7.2.4 HSI48 クロック

HSI48 クロック信号は、内部 48 MHz RC オシレータから生成され、直接 USB、または乱数生成器 (RNG) に使用できます。

内部 48 MHz RC オシレータは、主に、特殊なクロックリカバリシステム (CRS) 回路によって USB ペリフェラルに高精度のクロックを提供します。CRS では、動作中のオシレータの周波数を自動的かつ迅速に調整するために、USB SOF 信号、LSE、または外部信号を使用できます。これは、システムが STOP または STANDBY モードに移行した直後に無効化されます。CRS が使用されない場合、HSI48 RC オシレータは製造工程でのばらつきの影響を受けるデフォルトの周波数で実行されます。

CRS ペリフェラルの設定および使用方法の詳細については、[セクション 8：クロックリカバリシステム \(CRS\)](#) を参照してください。

HSI48 を有効化するには、VREFINT および 48 MHz RC を含むバッファが必要です（[セクション 10.2.3：リファレンス制御およびステータスレジスタ \(REF_CFGR3\)](#) の ENREF_HSI48 および EN_VREFINT を参照）。

クロックリカバリ RC レジスタ (RCC_CRRCR) の HSI48RDY フラグは、HSI48 RC が安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、HSI48 RC 出力クロックはリリースされません。

HSI48 RC は、クロックリカバリ RC レジスタ (RCC_CRRCR) の HSI48ON ビットを使用してオン／オフできます。

7.2.5 PLL

内部 PLL クロックは、HSI16 RC または HSE クリスタルによって駆動されます。システムクロックを駆動し、USB ペリフェラル用の 48 MHz クロックを生成するために使用できます (図 17 および [セクション 7.3.1 : クロック制御レジスタ \(RCC_CR\)](#) を参照)。

PLL 入力クロックの周波数範囲は、2 から 24 MHz である必要があります。

必要な周波数は、PLL に埋め込まれた通倍数と出力の分周比を使用して取得します。

- USB がクロックソースとして PLL を使用する場合、PLL VCO クロック (PLL 通倍数によって定義される) は、96 MHz 周波数を出力するようにプログラムする必要があります (USBCLK = PLLVCO/2)。
- システムクロックは、出力の分周比で分周した PLL VCO から生成されます。

注： アプリケーションソフトウェアでは、PLL 通倍数を正しくセットして、PLLVCO として 96 MHz (製品がレンジ1にある場合)

48 MHz (製品がレンジ2にある場合)、

24 MHz (製品がレンジ3にある場合) を、それぞれ超えないようにする必要があります。

また、SYSCLK として 32 MHz を超えないように、出力の分周比を正しくセットする必要があります。

PLL の最小入力クロック周波数は 2 MHz (PLL ソースとして HSE を使用している場合) です。

PLL 設定 (クロックソースの選択、通倍数、出力の分周比) は、PLL を有効にする前に実行する必要があります。PLL が有効になると、これらのパラメータは変更できません。

PLL 設定を変更するには、次の手順に従います。

1. PLLON を 0 にセットして PLL を無効にします。
2. PLLRDY がクリアされるまで待ちます。PLL が完全に停止します。
3. 任意のパラメータを変更します。
4. PLLON を 1 にセットして PLL を再度有効にします。

RCC_CIER レジスタで有効になっていれば、PLL がレディ状態になると割り込みを生成することができます ([セクション 7.3.5](#) を参照)。

7.2.6 LSE クロック

LSE クリスタルは、32.768 kHz の低速外部クリスタルまたはセラミック発振子です。時計／カレンダー、その他のタイミング機能のためのリアルタイムクロックペリフェラル (RTC) に、低電力ながら高精度のクロックソースを供給できるという利点があります。

LSE クリスタルは、RCC_CSR レジスタの LSEON ビットを使用してオン／オフできます ([セクション 7.3.21](#) を参照)。

最適な堅牢性と、起動時間の短縮、および消費電力の低減を実現するために、クリスタルオシレータの駆動能力は、RCC_CSR レジスタの LSEDRV[1:0] ビットによって、ランタイム時に変更できます。駆動機能は、低ドライブモードに達した場合を除き、異なるドライブレベル間で、動的に変更できます。この場合、パワーオンリセットまたは RTC リセットによってのみ、別のモードに変更できます。



RCC_CSR レジスタの LSE RDY フラグは、LSE クリスタルが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、LSE クリスタル出力クロック信号はリリースされません。RCC_CIER レジスタで有効になっていれば、割り込みを生成することができます ([セクション 7.3.5](#) を参照)。

外部ソース (LSE バイパス)

このモードでは、外部クロックソースが必要です。最大 1 MHz までの周波数を使用できます。このモードを選択するには、RCC_CR の LSEBYP および LSEON ビットをセットします ([セクション 7.3.1](#) を参照)。約50%のデューティサイクルを持つ外部クロック信号 (矩形波、正弦波、または三角波) で OSC_32_IN ピンを駆動する必要があり、その間、OSC_32_OUT ピンはハイインピーダンスに保つ必要があります ([図 18](#) を参照)。

7.2.7 LSI クロック

LSI RC は、独立型ウォッチドッグ (IWDG) の STOP モードおよび STANDBY モードでの動作を可能にする低電力のクロックソースとして動作します。クロック周波数は約 37 kHz です。

LSI RC オシレータは、RCC_CSR レジスタの LSION ビットを使用してオン/オフできます ([セクション 7.3.21](#) を参照)。

RCC_CSR の LSIRDY フラグは、内部低速オシレータが安定しているかどうかを示します。起動時、このビットがハードウェアによってセットされるまで、クロックは出力されません。RCC_CIER で有効になっていれば、割り込みを生成することができます ([セクション 7.3.5](#) を参照)。

LSI の測定

LSI オシレータの周波数分散を測定して、正確な RTC のタイムベースおよび/または許容できる精度の IWDG タイムアウト時間 (LSI がこれらのペリフェラルのクロックソースとして使用される場合) を得ることができます。詳細については、データシートの電気的特性のセクションを参照してください。LSI 周波数の測定方法の詳細については、[セクション 7.2.15: TIM21 を使用した内部/外部クロックの測定](#) を参照してください。

7.2.8 システムクロック (SYSCLK) の選択

次の 4 つのクロックソースを使用して、システムクロック (SYSCLK) を駆動できます：

- HSI16 オシレータ
- HSE オシレータ
- PLL
- MSI オシレータクロック (リセット後のデフォルト)

クロックソースが直接、または PLL を経由してシステムクロックとして使用されているときに、このクロック信号を停止することはできません。

あるクロックソースから別のクロックソースへの切り替えは、切り替え後に使用するクロックソースの準備ができていない場合 (起動遅延時間を経てクロックが安定している状態、または、PLL がロックされている状態) にのみ行われます。準備ができていないクロックソースが選択された場合は、クロックソースの準備ができたときに切り替えが行われます。RCC_CR レジスタのステータスビットは、どちらのクロックの準備ができていないか、およびどちらのクロックがシステムクロックとして使用されているかを示します。

7.2.9 システムクロックソースの周波数と電圧レンジ

次の表に、製品電圧レンジに応じて異なるクロックソースの最大周波数を示します。

表 32. システムクロックソースの周波数

| 製品電圧レンジ | クロック周波数 | | | |
|---------------|---------|--------|---|--------------------------------|
| | MSI | HSI16 | HSE | PLL |
| レンジ 1 (1.8 V) | 4.2 MHz | 16 MHz | HSE 32 MHz (外部クロック) または 24 MHz (クリスタル) | 32 MHz (PLLVCO 最大 = 96 MHz) |
| レンジ 2 (1.5 V) | 4.2 MHz | 16 MHz | 16 MHz | 16 MHz (PLLVCO 最大 = 48 MHz) |
| レンジ 3 (1.2 V) | 4.2 MHz | 該当なし | 8 MHz | 4 MHz (PLLVCO 最大 = 24 MHz) |

7.2.10 HSE クロックセキュリティシステム (CSS)

クロックセキュリティシステムは、HSEに対してソフトウェアで有効にできます。この場合、HSE オシレータのスタートアップ遅延時間の後にクロック検出回路が有効になり、オシレータが停止すると検出回路も無効になります。

HSE クロック障害が検出された場合、このオシレータは自動的に無効になり、ソフトウェアに障害を通知するために CSSHSEI 割り込み (クロックセキュリティシステム割り込み) が生成されるため、MCU は対応処理を行うことができます。CSSHSEI は、Cortex®-M0+ NMI (ノンマスカブル割り込み) 例外ベクタにリンクされています。

注： *CSSHSE を有効にすると、HSE クロックに障害が発生した場合に CSSHSE 割り込みが発生し、NMI が自動的に生成されます。NMI は、CSSHSE 割り込みペンディングビットがクリアされない限り、無限に実行されます。そのため、NMI 割り込みサービスルーチン (ISR) では、RCC_CICR レジスタの CSSHSEC ビットをセットして、CSSHSE 割り込みをクリアする必要があります。*

HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合 (間接的とは、PLL 入力クロックとして使用され、その PLL クロックがシステムクロックとして使用されることを意味する)、障害が検出されると、システムクロックが MSI オシレータに切り替えられ、HSE オシレータは無効になります。障害発生時に HSE オシレータクロックがシステムクロックとして使用されている PLL のクロック入力であった場合には、PLL も無効になります。

7.2.11 LSE クロックセキュリティシステム

クロックセキュリティシステムは、LSEに対してソフトウェアで有効にできます。この設定は、RCC_CSR レジスタの CSSLSEON ビットを書き込むことによって行います。このビットは、ハードウェアリセット、RTC ソフトウェアリセット、または LSE クロック障害検出後に無効にできます。CSSLSEON ビットは、LSE クロックと LSI クロックを有効 (LSEON および LSION をセット) かつレディ状態 (ハードウェアで LSERDY および LSIRDY ビットをセット) にし、RTCSEL ビットで RTC クロックを選択した後で書き込む必要があります。

LSE CSS は、RUN、SLEEP、STOP、および STANDBY のすべてのモードで動作します。

外部 32 kHz オシレータで障害が検出された場合、LSE クロックは RTC に供給されなくなりますが、レジスタの内容は変更されません。

STANDBY モードでは、ウェイクアップが生成されます。その他のモードでは、ソフトウェアをウェイクアップさせるために割り込みを送信できます ([セクション 7.3.5](#) を参照)。

その後、ソフトウェアでは CSSLSEON ビットをリセットし、また LSEON ビットをリセットして障害のある 32 kHz オシレータを停止する必要があります。RTCSEL ビットを通して RTC クロックソース (LSI、HSE、またはクロックなし) を変更するか、アプリケーションを保護するために必要な操作を行うことができます。

7.2.12 RTC および LCD クロック

RTC および LCD には同じクロックソース (LSE、LSI、または HSE 1 MHz クロック (プログラム可能なプリスケアラで分周した HSE) のいずれか) があります。これは、RCC_CSR レジスタの RTCSEL[1:0] ビット ([セクション 7.3.21](#) を参照) および RCC_CR レジスタの RTCPRE[1:0] ビット ([セクション 7.3.1](#) を参照) をプログラミングすることによって、選択できます。

選択した RTC および LCD クロックソースを変更するには、RCC_CSR レジスタの RTCRST ビットをセットするか、POR を使用する必要があります。

LSE または LSI が RTC クロックソースとして使用されている場合、RTC は引き続き 低電力の STOP モード および STANDBY モード で動作し、ウェイクアップソースとして使用できます。ただし、HSE が RTC クロックソースである場合は、RTC を 低電力の STOP および STANDBY モード で使用することはできません。LCD は、LSE または LSI が RTC クロックソースとして使用されている場合は、低電力の STOP モード で使用できます。

注 : *APB1 クロック周波数が RTC クロック周波数の 7 倍 ($7 \times \text{RTCCLK}$) 未満である場合に RTC カレンダーレジスタを読み出せるようにするには、カレンダー時間と日付レジスタをソフトウェアによって 2 回読み出す必要があります。*

RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。

7.2.13 ウォッチドッグクロック

独立型ウォッチドッグ (IWDG) がハードウェアのオプションまたはソフトウェアアクセスによって開始された場合、LSI オシレータは強制的にオンになり、オフにすることはできません。LSI オシレータの起動時の過渡期間が終わると、このクロックが IWDG に供給されます。

7.2.14 クロック信号出力

マイクロコントローラクロック出力 (MCO) 機能では、設定可能なプリスケアラ (1、2、4、8、または 16) を使用して、外部 MCO ピン (PA8 または PA9) にクロックを出力することができます。対応する GPIO ポートの設定レジスタは、オルタネート機能モードに設定されている必要があります。7 つのクロッキングシグナルのうちの 1 つを MCO クロックとして選択できます。

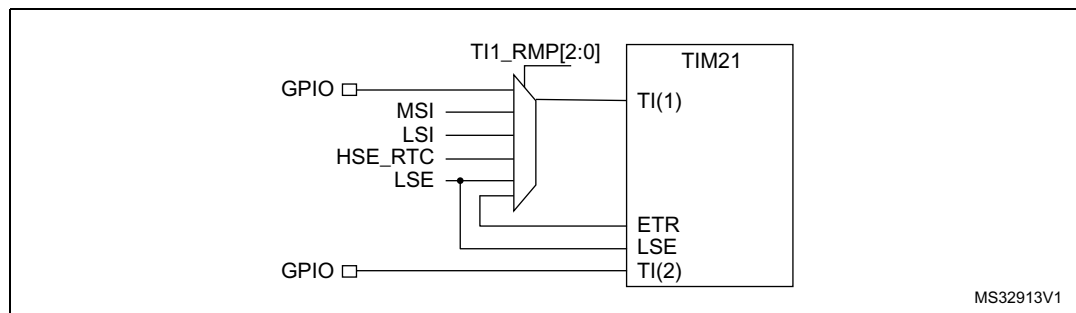
- SYSCLK
- HSI16
- HSI48
- MSI
- HSE
- PLL
- LSI
- LSE

選択は RCC_CFGR レジスタの MCOSEL[3:0] ビットによって制御されます ([セクション 7.3.20](#) を参照)。

7.2.15 TIM21 を使用した内部／外部クロックの測定

図 19 に示すように、TIM21 チャンネル 1 の入力キャプチャを使用することにより、ボード上のすべてのクロックソース発生回路の周波数を間接的に測定することができます。

図 19. 周波数を測定するための TIM21 チャンネル 1 の入力キャプチャの使用



TIM21 には、入力キャプチャのトリガに I/O または内部クロックのどちらを使用するかを選択する、入力マルチプレクサがあります。この選択は、TIM21_OR レジスタの TI1_RMP [2:0] ビットで行います。

LSE をチャンネル 1 の入力キャプチャに接続する主な目的は、HSI16 システムクロックと MSI システムクロックを正確に測定できるようにすることです（これには、HSI16 または MSI のどちらかがシステムクロックソースとして使用されている必要があります）。LSE 信号の連続したエッジ間の HSI16（または MSI）クロックのカウント数により、内部クロックの周期が測定されます。LSE クリスタルの高い精度（通常は数十 ppm）を利用することにより、同じ分解能で内部クロック周波数を判断でき、製造工程、温度、電圧に関連した周波数偏差に対してクロックソースを調整して補正することができます。

MSI および HSI16 の両方のオシレータには、上記の目的のためにユーザがアクセスできる専用の較正ビットがあります。

この基本概念は、相対測定（HSI16/LSE 比など）ができることにあり、そのため、精度は 2 つのクロックソース間の比に密接にかかわっています。この比が大きいほど、測定精度は向上します。

ただし、MSI クロック周波数が低い場合（通常は 1 MHz 未満）、十分な解像度を得ることはできません。この場合、以下を推奨します。

- 連続した複数のキャプチャの結果を累積します。
- タイマの入力キャプチャプリスケアラを使用します（8 周期ごとに最大 1 キャプチャ）。
- 512 Hz の RTC_OUT 信号（RTC クロックが LSE によって駆動される場合）を、チャンネル 1 入力キャプチャの入力として使います。これにより測定精度が向上します。

TIM21 は、LSI、MSI、または HSE_RTC の測定にも使用できます。これは、クリスタルを使用しないアプリケーションで有用です。超低電力 LSI オシレータには、大きな製造工程偏差があります。HSI16 クロックソースの関数として測定することで、その周波数を HSI16 の精度で決定できます。HSE_RTC 周波数（プログラム可能なプリスケアラで分周した HSE）が比較的高いため（1 MHz）、相対周波数測定はあまり正確ではありません。主な目的は、結果的におよそ外部クリスタル周波数を取得することです。これは、たとえば高調波周波数または低調波周波数（-50/+100% の偏差）を判断できることが求められる IEC 60730/IEC 61335 規格の要件を満たすのに有用です。

7.2.16 TIM2/TIM21/TIM22 のクロック独立型システムクロックソース

RTC のタイムベースとして 32.768 kHz クロックを使用する多くのアプリケーションでは、システムクロックから完全に独立して動作するタイムベースの使用が望ましいことがあります。これにより、プロセッサの状態（プロセッサは、停止か、低速／中速／高速動作）を考慮せずにタスクをスケジューリングすることができます。

この目的のため、LSE クロックは内部で、追加のクロックソースとして使用される 3 つのタイマの ETR 入力に接続されます。これにより、フラクショナルイベントに対して 1 つまたは 2 つの追加のコンペアチャネルを持つ最大 3 つの独立したタイムベース（自動再ロード機能を使用）が得られます。たとえば、TIM21 自動再ロード割り込みは 1 秒間のティック割り込み用にプログラムできます（メインティックの 250 ms 後に追加の割り込みが発生）。

注： *この設定では、外部クロック (LSE) と APB クロック間の比率が少なくとも 2 である必要があります。アプリケーションで LSE クロック周波数の 2 倍以下の APB クロック周波数を使用している場合は（通常、LSE = 32.768 kHz であるため、2 倍の LSE = 65.536 kHz）、タイマの外部トリガブリスケーラ機能を使用する必要があります。これにより、ETR クロックは最大 8 で分周できます。*

7.3 RCC レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1](#) を参照してください。

7.3.1 クロック制御レジスタ (RCC_CR)

アドレスオフセット : 0x00

システムリセット値 : 0b0000 0000 00XX 0X00 0000 0011 0000 0000 (X は未定義)

パワーオンリセット値 : 0x0000 0300

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|---------|-------|------|------|-------------|-----------|------------|-----------|-----------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | PLL RDY | PLLON | Res. | Res. | RTCPRE[1:0] | | CSSHSEON. | HSE BYP | HSE RDY | HSE ON |
| | | | | | | r | rw | | | rw | rw | rw | rw | r | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | MSI RDY | MSION | Res. | Res. | Res. | HSI16DIVF | HSI16DIVEN | HSI16RDYF | HSI16KEON | HSI16ON |
| | | | | | | r | rw | | | | r | rw | | r | rw |

ビット 31:26 予約済みであり、リセット値のままにしておかなければなりません。

ビット 25 **PLL RDY** : PLL クロックレディフラグ

このビットは、PLL がロック状態であることを示すために、ハードウェアによってセットされます。

0 : PLL アンロック

1 : PLL ロック

ビット 24 **PLLON** : PLL イネーブルビット

このビットは、PLL を有効にするために、ソフトウェアによってセット/クリアされます。

STOP または STANDBY モードに移行するときに、ハードウェアによってクリアされます。このビットは、PLL クロックがシステムクロックとして使用されている場合、またはシステムクロックとして選択されている場合は、リセットできません。

0 : PLL オフ

1 : PLL オン

ビット 23:22 予約済みであり、リセット値のままにしておかなければなりません。

ビット 21:20 **RTCPRE[1:0]** RTC/LCD プリスケアラ

これらのビットは、1 MHz クロックを HSE から取得するために、ソフトウェアによってセット/リセットされます。このプリスケアラは、HSE が有効 (HSEON = 1) な場合は変更できません。これらのビットはパワーオンリセット時にリセットされます。これらの値は、システムリセットによって変更されません。

00 : RTC/LCD クロック用に HSE を 2 分周

01 : RTC/LCD クロック用に HSE を 4 分周

10 : RTC/LCD クロック用に HSE を 8 分周

11 : RTC/LCD クロック用に HSE を 16 分周

ビット 19 **CSSHSEON** : HSEのクロックセキュリティシステムのイネーブルビット

このビットは、HSE のクロックセキュリティシステム (CSS) を有効にするために、ソフトウェアによってセット/クリアされます。CSSHSEON がセットされているとき、HSE オシレータがレディになるとクロック検出回路がハードウェアによって有効にされ、オシレータの障害が検出された場合にハードウェアによって無効にされます。

0 : クロックセキュリティシステムオフ (クロック検出回路オフ)

1 : クロックセキュリティシステムオン (HSE オシレータが安定していればクロック検出回路オン、そうでなければオフ)



ビット 18 HSEBYP : HSE クロックバイパスビット

このビットは、外部クロックでオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。デバイスで使用するには、HSEON ビットで外部クロックを有効にする必要があります。HSEBYP ビットは、HSE オシレータが無効のときのみ、書き込みができます。このビットは、パワーオンリセットによってリセットされます。この値は、システムリセットによって変更されません。

0 : HSE オシレータはバイパスされません。

1 : HSE オシレータはバイパスされ、外部クロックが使用されます。

ビット 17 HSERDY : HSE クロックレディフラグ

このビットは、HSE オシレータが安定状態であることを示すために、ハードウェアによってセットされます。HSEON ビットがクリアされた後、HSERDY は、HSE オシレータクロックの 6 サイクル後にローになります。

0 : HSE オシレータはレディ状態ではありません。

1 : HSE オシレータはレディ状態です。

ビット 16 HSEON : HSE クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

STOP または STANDBY モードに移行するときに、HSE オシレータを停止するためにハードウェアによってクリアされます。HSE オシレータが直接的または間接的にシステムクロックとして使用されている場合は、このビットをリセットできません。

0 : HSE オシレータオフ

1 : HSE オシレータオン

ビット 15:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 MSIRDY : MSI クロックレディフラグ

このビットは、MSI オシレータが安定状態であることを示すために、ハードウェアによってセットされます。

0 : MSI オシレータはレディ状態ではありません。

1 : MSI オシレータはレディ状態です。

注 : *MSION ビットがクリアされると、MSIRDY は、MSI オシレータクロックの 6 サイクル後にローになります。*

ビット 8 MSION : MSI クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

STOP または STANDBY モードを終了するとき、または、直接的または間接的にシステムクロックとして使用されている HSE オシレータに障害が発生した場合、MSI オシレータを強制的にオンにするために、ハードウェアによってセットされます。MSI オシレータがシステムクロックとして使用されている場合は、このビットをクリアできません。

0 : MSI オシレータオフ

1 : MSI オシレータオン

ビット 7:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 HSI16DIVF HSI16 分周器フラグ

このビットは、ハードウェアによってセット／リセットされます。HSI16DIVEN への書き込みは、周波数にすぐに影響を及ぼすことはないため、このフラグは HSI16 分周器の現在のステータスを示します。

0 : 16 MHz HSI クロックは分周されません。

1 : 16 MHz HSI クロックは4 分周されます。

ビット 3 HSI16DIVEN HSI16 分周器イネーブルビット

このビットは、16 MHz HSIの4分周を有効／無効にするために、ソフトウェアによってセット／リセットされます。いつでも書き込み可能です。

0 : 16 MHz HSI の分周は要求されていません。

1 : 16 MHz HSI の4分周が要求されています。

ビット 2 HSI16RDYF : 内部高速クロックレディフラグ

このビットは、HSI 16 MHz オシレータが安定状態であることを示すために、ハードウェアによってセットされます。HSI16ON ビットがクリアされた後、HSI16RDY は、HSI16 オシレータクロックの 6 サイクル後にローになります。

0 : HSI 16 MHz オシレータはレディ状態ではありません。

1 : HSI 16 MHz オシレータはレディ状態です。

ビット 1 HSI16KERON : いくつかの IP カーネル用の高速内部クロックイネーブルビット

このビットは、HSI 16 MHz RC を強制的にオンにして (STOP モードの場合も含む)、USART または I2C1 のカーネルクロックとしてすばやく使用できるようにするために、ソフトウェアによってセット／リセットされます。このビットは、HSI16ON の値への影響はありません。

0 : HSI 16 MHz オシレータは、強制的にオンにはなりません。

1 : HSI 16 MHz オシレータは、強制的にオンになります (STOP モードの場合も含む)。

ビット 0 HSI16ON : 16 MHz 高速内部クロックイネーブル

このビットは、ソフトウェアによってセット／クリアされます。16 MHz HSI オシレータが直接的または間接的にシステムクロックとして使用されている場合は、このビットをクリアできません。

0 : HSI16 オシレータオフ

1 : HSI16 オシレータオン

7.3.2 内部クロックソース較正レジスタ (RCC_ICSCR)

アドレスオフセット : 0x04

リセット値 : 0x00XX B0XX (X は未定義)。

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|---------------|----|----|----------------|----|----|----|----|---------------|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| MSITRIM[7:0] | | | | | | | | MSICAL[7:0] | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MSIRANGE[2:0] | | | HSI16TRIM[4:0] | | | | | HSI16CAL[7:0] | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | r | r | r | r | r | r | r | r |

ビット 31:24 **MSITRIM[7:0]** : MSI クロックトリミング

これらのビットは、MSI 較正を調整するために、ソフトウェアによってセットされます。
これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は MSICAL[7:0] ビットに加算されます。内部 MSI RC の周波数に影響する電圧や温度の変化に対応できるようにプログラムできます。

ビット 23:16 **MSICAL[7:0]** : MSI クロック較正

これらのビットは、起動時に自動的に初期化されます。

ビット 15:13 **MSIRANGE[2:0]** : MSI クロックレンジ

これらのビットは、MSI の周波数レンジを選択するために、ソフトウェアによってセットされます。7 つの周波数レンジを使用できます。
000 : レンジ 0 (約 65.536 kHz)
001 : レンジ 1 (約 131.072 kHz)
010 : レンジ 2 (約 262.144 kHz)
011 : レンジ 3 (約 524.288 kHz)
100 : レンジ 4 (約 1.048 MHz)
101 : レンジ 5 (約 2.097 MHz、リセット値)
110 : レンジ 6 (約 4.194 MHz)
111 : 設定禁止

ビット 12:8 **HSI16TRIM[4:0]** : 内部高速クロックトリミング

これらのビットにより、ユーザによるプログラムが可能なトリミング値が使用でき、この値は HSI16CAL[7:0] ビットに加算されます。内部 HSI16 RC の周波数に影響する電圧や温度の変化に対応できるようにプログラムできます。

ビット 7:0 **HSI16CAL[7:0]** 内部高速クロック較正

これらのビットは、スタートアップ時に自動的に初期化されます。

7.3.3 クロックリカバリ RC レジスタ (RCC_CRRCR)

アドレス : 0x08

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|---------------|------|------|------|------|------|------|------|------|------|------|------|------|------|----------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| HSI48CAL[7:0] | | | | | | | | Res. | Res. | Res. | Res. | Res. | Res. | HSI48RDY | HSI48ON |
| r | r | r | r | r | r | r | r | | | | | | | r | rw |

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:8 **HSI48CAL[7:0]** : 48 MHz HSI クロック較正

これらのビットは読み出し専用です。これらは、オプションバイトをシステムリセット中にロードすることで、ハードウェアによってセットされます。

ビット 7:2 予約済みであり、リセット値のままにしておかなければなりません。

ビット 1 **HSI48RDY** : 48MHz HSI クロックレディフラグ

このビットは、48 MHz RC オシレータが安定状態であることを示すために、ハードウェアによってセットされます。HSION リセット後に低下させるには、48 MHz RC オシレータクロックが 6 サイクル必要です。

0 : 48 MHz HSI クロックはレディ状態ではありません。

1 : 48 MHz HSI クロックはレディ状態です。

ビット 0 **HSI48ON** : 48 MHz HSI クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 48 MHz HSI クロックオフ

1 : 48 MHz HSI クロックオン

7.3.4 クロック設定レジスタ (RCC_CFGR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

アクセス : $0 \leq$ ウェイトステート ≤ 2 、ワード、ハーフワード、およびバイトアクセス

クロックソースの切り替え中にアクセスが発生した場合に限り、1 または 2ウェイトステートが挿入されます。

| | | | | | | | | | | | | | | | |
|------------|------------|------------|----|-------------|------------|----|----|-------------|----|-------------|----|----------|----|---------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | MCPRE[2:0] | | | MCOSEL[3:0] | | | | PLLDIV[1:0] | | PLLMUL[3:0] | | | | Res. | PLL SRC |
| | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| STOPW UCK. | Res. | PPRE2[2:0] | | | PPRE1[2:0] | | | HPRE[3:0] | | | | SWS[1:0] | | SW[1:0] | |
| rw | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | rw |



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

ビット 30:28 **MCOPRE[2:0]**: マイクロコントローラクロック出力プリスケアラ

これらのビットは、ソフトウェアによってセット／クリアされます。
MCO 出力を有効にする前に、このプリスケアラを変更することを強く推奨します。

- 000: MCO を 1 分周
- 001: MCO を 2 分周
- 010: MCO を 4 分周
- 011: MCO を 8 分周
- 100: MCO を 16 分周
- その他: 設定禁止

ビット 27:24 **MCOSSEL[3:0]**: マイクロコントローラクロック出力選択

これらのビットは、ソフトウェアによってセット／クリアされます。

- 0000: MCO 出力無効、MCO にクロックなし
- 0001: SYSCLK クロックの選択
- 0010: HSI16 オシレータクロックの選択
- 0011: MSI オシレータクロックの選択
- 0100: HSE オシレータクロックの選択
- 0101: PLL クロックの選択
- 0110: LSI オシレータクロックの選択
- 0111: LSE オシレータクロックの選択
- 1000: HSI48 オシレータクロックの選択
- その他: 予約済み

注: このクロック出力では、起動時または MCO クロックソースの切り替え時に、切り捨てのサイクルがある場合があります。

ビット 23:22 **PLLDIV[1:0]**: PLL 出力の分周比

これらのビットは、PLL VCO クロックの PLL 出力クロック分周比を制御するために、ソフトウェアによってセット／クリアされます。これらのビットに書き込めるのは、PLL が無効なときだけです。

- 00: 設定禁止
- 01: PLL クロック出力 = PLLVCO / 2
- 10: PLL クロック出力 = PLLVCO / 3
- 11: PLL クロック出力 = PLLVCO / 4

ビット 21:18 **PLLMUL[3:0]**: PLL 通倍数

これらのビットは、PLL VCO クロックを生成する PLL 通倍数を定義するために、ソフトウェアによって書き込まれます。これらのビットに書き込めるのは、PLL が無効なときだけです。

- 0000: PLLVCO = PLL クロック入力 × 3
- 0001: PLLVCO = PLL クロック入力 × 4
- 0010: PLLVCO = PLL クロック入力 × 6
- 0011: PLLVCO = PLL クロック入力 × 8
- 0100: PLLVCO = PLL クロック入力 × 12
- 0101: PLLVCO = PLL クロック入力 × 16
- 0110: PLLVCO = PLL クロック入力 × 24
- 0111: PLLVCO = PLL クロック入力 × 32
- 1000: PLLVCO = PLL クロック入力 × 48
- その他: 設定禁止

注意: PLL VCO クロック周波数は、96 MHz (製品がレンジ 1 にある場合)、48 MHz (レンジ 2 にある場合)、および 24 MHz (レンジ 3 にある場合) を、超えないようにしてください。

ビット 17 予約済みであり、リセット値のままにしておかなければなりません。

ビット 16 PLLSRC : PLL 入力クロックソース

このビットは、PLL クロックソースを選択するために、ソフトウェアによってセット/クリアされます。このビットに書き込めるのは、PLL が無効なときだけです。

0 : HSI16 オシレータクロックが PLL 入力クロックとして選択されます。

1 : HSE オシレータクロックが PLL 入力クロックとして選択されます。

注： PLL の最小入力クロック周波数は、2 MHz です。

ビット 15 STOPWUCK : STOPモードのウェイクアップクロック選択

このビットは、STOPモードのウェイクアップクロックを選択するために、ソフトウェアによってセット/クリアされます。

0 : STOPモードのウェイクアップクロックとして内部 64 KHz - 4 MHz (MSI) オシレータを選択

1 : STOPモードのウェイクアップクロックとして内部 16 MHz (HSI16) オシレータ (または、HSI16DIVEN = 1 の場合、HSI16/4) を選択

ビット 14 予約済みであり、リセット値のままにしておかなければなりません。**ビット 13:11 PPRE2[2:0]** : APB 高速プリスケアラ (APB2)

これらのビットは、APB 高速クロック (PCLK2) の分周比を制御するために、ソフトウェアによってセット/クリアされます。

0xx : HCLK は分周されません。

100 : HCLK は 2 分周されます。

101 : HCLK は 4 分周されます。

110 : HCLK は 8 分周されます。

111 : HCLK は 16 分周されます。

ビット 10:8 PPRE1[2:0] : APB低速プリスケアラ (APB1)

これらのビットは、APB ロースピードクロック (PCLK1) の分周比を制御するために、ソフトウェアによってセット/クリアされます。

0xx : HCLK は分周されません。

100 : HCLK は 2 分周されます。

101 : HCLK は 4 分周されます。

110 : HCLK は 8 分周されます。

111 : HCLK は 16 分周されます。

ビット 7:4 **HPRE[3:0]** : AHB プリスケアラ

これらのビットは、AHB クロックの分周比を制御するために、ソフトウェアによってセット／クリアされます。

注意： デバイスの電圧レンジに応じて、ソフトウェアではこれらのビットを正しくセットし、システム周波数が最大許容周波数を超えないようにします（詳細については、PWR の章の「ダイナミック電圧スケーリングの管理」セクションを参照）。これらのビットの書き込み動作後、電圧レンジを下げる前に、このレジスタを読み出して新しい値が正しく反映されていることを確認する必要があります。

- 0xxx : SYSCLK は分周されません。
- 1000 : SYSCLK は 2 分周されます。
- 1001 : SYSCLK は 4 分周されます。
- 1010 : SYSCLK は 8 分周されます。
- 1011 : SYSCLK は 16 分周されます。
- 1100 : SYSCLK は 64 分周されます。
- 1101 : SYSCLK は 128 分周されます。
- 1110 : SYSCLK は 256 分周されます。
- 1111 : SYSCLK は 512 分周されます。

ビット 3:2 **SWS[1:0]** : システムクロックスイッチステータス

これらのビットは、どのクロックソースがシステムクロックとして使用されているかを示すために、ハードウェアによってセット／クリアされます。

- 00 : MSI オシレータがシステムクロックとして使用されています。
- 01 : HSI16 オシレータがシステムクロックとして使用されています。
- 10 : HSE オシレータがシステムクロックとして使用されています。
- 11 : PLL がシステムクロックとして使用されています。

ビット 1:0 **SW[1:0]** : システムクロックスイッチ

これらのビットは、SYSCLK ソースを選択するために、ソフトウェアによってセット／クリアされます。

STANDBY モードを終了するとき、またはシステムクロックとして直接的または間接的に使用されている HSE オシレータに障害が発生したときに、強制的に MSI を選択するようにハードウェアによってセットされます。

- 00 : MSI オシレータがシステムクロックとして使用されています。
- 01 : HSI16 オシレータがシステムクロックとして使用されています。
- 10 : HSE オシレータがシステムクロックとして使用されています。
- 11 : PLL がシステムクロックとして使用されています。

7.3.5 クロック割り込み有効レジスタ（RCC_CIER）

アドレス : 0x10

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------------|----------------|--------------|--------------|--------------|----------------|--------------|--------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CSS LSE | HSI48 RDYIE | MSI RDYIE | PLL RDYIE | HSE RDYIE | HSI16 RDYIE | LSE RDYIE | LSI RDYIE |
| | | | | | | | | r | r | r | r | r | r | r | r |



ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **CSSLSE** : LSE CSS 割り込みフラグ

このビットは、外部 32 kHz オシレータのクロックセキュリティシステムによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : LSE CSS 割り込みは無効です。

1 : LSE CSS 割り込みは有効です。

ビット 6 **HSI48RDYIE** : HSI48 レディ割り込みフラグ

このビットは、HSI48 オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : HSI48 レディ割り込みは無効です。

1 : HSI48 レディ割り込みは有効です。

ビット 5 **MSIRDYIE** : MSI レディ割り込みフラグ

このビットは、MSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : MSI レディ割り込みは無効です。

1 : MSI レディ割り込みは有効です。

ビット 4 **PLLRDYIE** : PLL レディ割り込みフラグ

このビットは、PLL ロックによって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : PLL ロック割り込みは無効です。

1 : PLL ロック割り込みは有効です。

ビット 3 **HSERDYIE** : HSE レディ割り込みフラグ

このビットは、HSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : HSE レディ割り込みは無効です。

1 : HSE レディ割り込みは有効です。

ビット 2 **HSI16RDYIE** : HSI16 レディ割り込みフラグ

このビットは、HSI16 オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : HSI16 レディ割り込みは無効です。

1 : HSI16 レディ割り込みは有効です。

ビット 1 **LSERDYIE** : LSE レディ割り込みフラグ

このビットは、LSE オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : LSE レディ割り込みは無効です。

1 : LSE レディ割り込みは有効です。

ビット 0 **LSIRDYIE** : LSI レディ割り込みフラグ

このビットは、LSI オシレータの安定化によって発生する割り込みを有効／無効にするために、ソフトウェアによってセット／リセットされます。

0 : LSI レディ割り込みは無効です。

1 : LSI レディ割り込みは有効です。

7.3.6 クロック割り込みフラグレジスタ (RCC_CIFR)

アドレス : 0x14

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|-------------|-------------|---------------|-------------|-------------|-------------|---------------|-------------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | CSS HSEF | CSS LSEF | HSI48 RDYF | MSI RDYF | PLL RDYF | HSE RDYF | HSI16 RDYF | LSE RDYF | LSI RDYF |
| | | | | | | | r | r | r | r | r | r | r | r | r |

ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8 **CSSHSEF** : クロックセキュリティシステム割り込みフラグ

このビットは、ソフトウェアでCSSHSEC ビットに書き込むことによってリセットされます。これは、HSE クロックに障害が発生した場合に、ハードウェアによってセットされます。
0 : HSE クロック障害によるクロックセキュリティ割り込みは発生していません。
1 : HSE クロック障害によってクロックセキュリティ割り込みが発生しました。

ビット 7 **CSSLSEF** : LSE クロックセキュリティシステム割り込みフラグ

このビットは、ソフトウェアでCSSLSEC ビットに書き込むことによってリセットされます。これは、LSE クロックに障害が発生し、CSSLSE がセットされた場合に、ハードウェアによってセットされます。
0 : LSE クロック障害時に障害は検出されません。
1 : LSE クロック障害時に障害が検出されました。

ビット 6 **HSI48RDYF** : HSI48 レディ割り込みフラグ

このビットは、HSI48RDYC ビットに書き込むことにより、ソフトウェアによってリセットされます。CSS が安定し、HSI48RDYIE ビットがセットされているときに、ハードウェアによってセットされません。
0 : HSI48 クロック障害によるクロックレディ割り込みは発生していません。
1 : HSI48 クロック障害によるクロックレディ割り込みは発生していません。

ビット 5 **MSIRDYF** : MSI レディ割り込みフラグ

このビットは、ソフトウェアでMSIRDYC ビットに書き込むことによってリセットされます。MSI クロックが安定し、MSIRDYIE ビットがセットされているときに、ハードウェアによってセットされます。
0 : MSI クロック障害によるクロックレディ割り込みは発生していません。
1 : MSI クロック障害によるクロックレディ割り込み割り込みが発生しました。

ビット 4 **PLLRDYF** : PLL レディ割り込みフラグ

このビットは、ソフトウェアでPLLRDYC ビットに書き込むことによってリセットされます。PLL クロックが安定し、PLLRDYIE ビットがセットされているときに、ハードウェアによってセットされます。
0 : PLL クロック障害によるクロックレディ割り込みは発生していません。
1 : PLL クロック障害によるクロックレディ割り込みが発生しました。

ビット 3 **HSERDYF** : HSE レディ割り込みフラグ

このビットは、ソフトウェアでHSERDYC ビットに書き込むことによってリセットされます。HSE クロックが安定し、HSERDYIE ビットがセットされているときに、ハードウェアによってセットされません。
0 : HSE クロック障害によるクロックレディ割り込みは発生していません。
1 : HSE クロック障害によるクロックレディ割り込みが発生しました。

ビット 2 **HSI16RDYF** : HSI16 レディ割り込みフラグ

このビットは、ソフトウェアでHSI16RDYC ビットに書き込むことによってリセットされます。HSE クロックが安定し、HSI16RDYIE ビットがセットされているときに、ハードウェアによってセットされず。

- 0 : HSI16 クロック障害によるクロックレディ割り込みが発生しました。
- 1 : HSI16 クロック障害によるクロックレディ割り込みが発生しました。

ビット 1 **LSERDYF** : LSE レディ割り込みフラグ

このビットは、ソフトウェアでLSERDYC ビットに書き込むことによってリセットされます。LSE クロックが安定し、LSERDYIE ビットがセットされているときに、ハードウェアによってセットされます。

- 0 : LSE クロック障害によるクロックレディ割り込みは発生していません。
- 1 : LSE クロック障害によるクロックレディ割り込みが発生しました。

ビット 0 **LSIRDYF** : LSI レディ割り込みフラグ

このビットは、ソフトウェアでLSIRDYC ビットに書き込むことによってリセットされます。LSI クロックが安定し、LSIRDYIE ビットがセットされているときに、ハードウェアによってセットされます。

- 0 : LSI クロック障害によるクロックレディ割り込みは発生していません。
- 1 : LSI クロック障害によるクロックレディ割り込みが発生しました。

7.3.7 クロック割り込みクリアレジスタ (RCC_CICR)

アドレス : 0x18

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|-------------|-------------|---------------|-------------|-------------|-------------|---------------|--------------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | CSS HSEC | CSS LSEC | HSI48 RDYC | MSI RDYC | PLL RDYC | HSE RDYC | HSI16 RDYC | LSE RDYIC | LSI RDYC |
| | | | | | | | r | r | r | r | r | r | r | r | r |

ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8 **CSSHSEC** : クロックセキュリティシステム割り込みクリア

このビットは、CSSHSEF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

- 0 : 影響なし。
- 1 : CSSHSEF フラグをクリアします。

ビット 7 **CSSLSEC** : LSE クロックセキュリティシステム割り込みクリア

このビットは、CSSLSEF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

- 0 : 影響なし。
- 1 : CSSLSEF フラグをクリアします。

ビット 6 **HSI48RDYC** : HSI48 レディ割り込みクリア

このビットは、HSI48RDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

- 0 : 影響なし。
- 1 : HSI48RDYF フラグをクリアします。



ビット 5 MSIRDYC : MSI レディ割り込みクリア

このビットは、MSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : MSIRDYF フラグをクリアします。

ビット 4 PLLRDYC : PLL レディ割り込みクリア

このビットは、PLLRDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : PLLRDYF フラグをクリアします。

ビット 3 HSERDYC : HSE レディ割り込みクリア

このビットは、HSERDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : HSERDYF フラグをクリアします。

ビット 2 HSI16RDYC : HSI16 レディ割り込みクリア

このビットは、HSI16RDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : HSI16RDYF フラグをクリアします。

ビット 1 LSERDYC : LSE レディ割り込みクリア

このビットは、LSERDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : LSERDYF フラグをクリアします。

ビット 0 LSIRDYC : LSI レディ割り込みクリア

このビットは、LSIRDYF フラグをクリアするために、ソフトウェアによってセットされます。ハードウェアによってリセットされます。

0 : 影響なし。

1 : LSIRDYF フラグをクリアします。

7.3.8 GPIO リセットレジスタ (RCC_IOPRSTR)

アドレス : 0x1C

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|-------------|------|------|------|-------------|-------------|-------------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IOPH RST | Res. | Res. | Res. | IOPD RST | IOPC RST | IOPB RST | IOPA RST |
| | | | | | | | | rw | | | | rw | rw | rw | rw |

ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **IOPHRST** : I/O ポート H リセット

このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : I/O ポート H をリセットします。

ビット 6:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **IOPDRST** : I/O ポート D リセット

このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : I/O ポート D をリセットします。

ビット 2 **IOPCRST** : I/O ポート C リセット

このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : I/O ポート C をリセットします。

ビット 1 **IOPBRST** : I/O ポート B リセット

このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : I/O ポート B をリセットします。

ビット 0 **IOPARST** : I/O ポート A リセット

このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : I/O ポート A をリセットします。

7.3.9 AHB ペリフェラルリセットレジスタ (RCC_AHBRSTR)

アドレスオフセット : 0x20

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|------|------------|------|------|------|-------------|------|------|------|------------|------|------|------|------------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | CRYP RST | Res. | Res. | Res. | RNGRS T | Res. | Res. | Res. | TSCRST |
| | | | | | | | rw | | | | rw | | | | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | CRC RST | Res. | Res. | Res. | MIF RST | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DMA RST |
| | | | rw | | | | rw | | | | | | | | rw |

ビット 31:25 予約済みであり、リセット値のままにしておかなければなりません。

ビット 24 **CRYPTRST** : 暗号モジュールリセット

このビットは、ソフトウェアによってセット／リセットされます。

0 : 影響なし。

1 : CRYPTO モジュールをリセットします。

ビット 23:21 予約済みであり、リセット値のままにしておかなければなりません。

ビット 20 **RNGRST** : 乱数発生回路モジュールリセット

このビットは、ソフトウェアによってセット／リセットされます。

0 : 影響なし。

1 : RNG モジュールをリセットします。

ビット 19:17 予約済みであり、リセット値のままにしておかなければなりません。

ビット 16 **TSCRST** : タッチセンシングリセット

このビットは、ソフトウェアによってセット／リセットされます。

0 : 影響なし。

1 : タッチセンシングモジュールをリセットします。

ビット 15 : 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **CRCRST** : テスト統合モジュールリセット

このビットは、ソフトウェアによってセット／リセットされます。

0 : 影響なし。

1 : テスト統合モジュールをリセットします。

ビット 11:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8 **MIFRST** : メモリインタフェースリセット

このビットは、ソフトウェアによってセット／リセットされます。

このリセットは、E2 が I_{DDQ} モードの場合にのみ有効になります。

0 : 影響なし。

1 : メモリインタフェースをリセットします。

ビット 7:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **DMARST** : DMA リセット

このビットは、ソフトウェアによってセット／リセットされます。

0 : 影響なし。

1 : DMA をリセットします。

7.3.10 APB2 ペリフェラルリセットレジスタ (RCC_APB2RSTR)

アドレスオフセット : 0x24

リセット値 : 0x00000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|---------------|------|-------------|------|------|------------|------|------|------------|--------------|------|------|--------------|------|---------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG RST | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | rw | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | USART1 RST | Res. | SPI1 RST | Res. | Res. | ADC RST | Res. | Res. | Res. | TIM22 RST | Res. | Res. | TIM21 RST | Res. | SYSCFG RST |
| | rw | | rw | | | rw | | | | rw | | | rw | | rw |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **DBG RST** : DBG リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : DBG をリセットします。

ビット 21:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **USART1 RST** : USART1 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : USART1 をリセットします。

ビット 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **SPI1 RST** : SPI 1 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : SPI 1 をリセットします。

ビット 11:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **ADC RST** : ADC インタフェースリセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : ADC インタフェースをリセットします。

ビット 8:6 予約済みであり、リセット値のままにしておかなければなりません。

ビット 5 **TIM22 RST** : TIM22 タイマリセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : TIM22 タイマをリセットします。

ビット 4:3 予約済みであり、リセット値のままにしておかなければなりません。

ビット 2 **TIM21RST** : TIM21 タイマリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : TIM21 タイマをリセットします。

ビット 1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **SYSCFGRST** : システム設定コントローラリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : システム設定コントローラをリセットします。

7.3.11 APB1 ペリフェラルリセットレジスタ（RCC_APB1RSTR）

アドレスオフセット : 0x28
リセット値 : 0x0000 0000
アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|---------------|-------------|------------|------------|-------------|------|--------|------|--------|-------------|-------------|-------------|------|----------------|---------------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| LPTIM1 RST | Res. | DACRS T | PWR RST | CRS RST | Res. | Res. | Res. | USBRST | I2C2 RST | I2C1 RST | Res. | Res. | LPUART1 RST | USART2 RST | Res. |
| rw | | rw | rw | rw | | | | rw | rw | rw | | | rw | rw | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | SPI2 RST | Res. | Res. | WWDG RST | Res. | LCDRST | Res. | Res. | Res. | Res. | TIM6 RST | Res. | Res. | Res. | TIM2 RST |
| | rw | | | rw | | rw | | | | | rw | | | | rw |

ビット 31 **LPTIM1RST** : 低電力タイマリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : 低電力タイマをリセットします。

ビット 30 予約済みであり、リセット値のままにしておかなければなりません。

ビット 29 **DACRST** : DAC インタフェースリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : DAC インタフェースリセット

ビット 28 **PWRRST** : 電源インタフェースリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : 電源インタフェースをリセットします。

ビット 27 **CRSRST** : クロックリカバリシステムリセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : クロックリカバリシステムをリセットします。

ビット 26:24 予約済みであり、リセット値のままにしておかなければなりません。

ビット 23 **USBRST** : USB リセット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 影響なし。
1 : USB をリセットします。

ビット 22 I2C2RST : I2C2 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : I2C2 をリセットします。

ビット 21 I2C1RST : I2C1 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : I2C1 をリセットします。

ビット 20:19 予約済みであり、リセット値のままにしておかなければなりません。

ビット 18 LPUART1RST : LPUART1 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : LPUART1 をリセットします。

ビット 17 UART2RST : UART2 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : UART2 をリセットします。

ビット 16:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 SPI2RST : SPI2 リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : SPI2 をリセットします。

ビット 13:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11 WWDGRST : ウィンドウ型ウォッチドッグリセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : ウィンドウ型ウォッチドッグをリセットします。

ビット 10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 LCDRST : LCD リセット

このビットは、ソフトウェアによってセット／クリアされます。

0 : 影響なし。

1 : LCD をリセットします。

ビット 8:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 TIM6RST : タイマ 6 リセット

ソフトウェアでセット／クリアされます。

0 : 影響なし。

1 : timer6 をリセットします。

ビット 3:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 TIM2RST : Timer2 リセット

ソフトウェアでセット／クリアされます。

0 : 影響なし。

1 : timer2 をリセットします。

7.3.12 GPIO クロック有効レジスタ (RCC_IOPENR)

アドレス : 0x2C

リセット値 : 0x0000 0000

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------------|------|------|------|------------|------------|------------|------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IOPH EN | Res. | Res. | Res. | IOPD EN | IOPC EN | IOPB EN | IOPA EN |
| | | | | | | | | rw | | | | rw | rw | rw | rw |

ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **IOPHEN** : I/O ポート H クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : ポート H クロック無効

1 : ポート H クロック有効

ビット 6:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **IOPDEN** : I/O ポート D クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : ポート D クロック無効

1 : ポート D クロック有効

ビット 2 **IOPCEN** : I/O ポート C クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : ポート C クロック無効

1 : ポート C クロック有効

ビット 1 **IOPBEN** : I/O ポート B クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : ポート B クロック無効

1 : ポート B クロック有効

ビット 0 **IOPAEN** : I/O ポート A クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : ポート A クロック無効

1 : ポート A クロック有効

7.3.13 AHB ペリフェラルクロック有効レジスタ (RCC_AHBENR)

アドレスオフセット : 0x30

リセット値 : 0x0000 0100

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの値はソフトウェアによって読み出しできず、返される値は常に 0x0 になります。

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|------|-----------|------|------|------|------------|------|------|------|-------|------|------|------|-------------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | CRYP EN | Res. | Res. | Res. | RNGEN | Res. | Res. | Res. | TOUCHE N |
| | | | | | | | rw | | | | rw | | | | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | CRC EN | Res. | Res. | Res. | MIF EN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DMA EN |
| | | | rw | | | | rw | | | | | | | | rw |

ビット 31:25 予約済みであり、リセット値のままにしておかなければなりません。

ビット 24 **CRYPEN** : 暗号クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : 暗号クロック無効

1 : 暗号クロック有効

ビット 23:21 予約済みであり、リセット値のままにしておかなければなりません。

ビット 20 **RNGEN** : 乱数発生器クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : RNG クロック無効

1 : RNG クロック有効

ビット 19:17 予約済みであり、リセット値のままにしておかなければなりません。

ビット 16 **TOUCHEN** : タッチセンシングクロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : タッチセンシングクロック無効

1 : タッチセンシングクロック有効

ビット 15 : 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **CRCEN** : CRC クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : テスト統合モジュールクロック無効

1 : テスト統合モジュールクロック有効

ビット 11:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8 **MIFEN** : NVM インタフェースクロックイネーブルビット

このビットは、ソフトウェアによってセット／リセットされます。

このリセットは、NVM がパワーダウンモードの場合にのみ有効になります。

0 : NVM インタフェースクロック無効

1 : NVM インタフェースクロック有効

ビット 7:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **DMAEN** : DMA クロックイネーブルビット

このビットは、ソフトウェアによってセット／リセットされます。

0 : DMA クロック無効

1 : DMA クロック有効

7.3.14 APB2 ペリフェラルクロック有効レジスタ (RCC_APB2ENR)

アドレス : 0x34

リセット値 : 0x0000 0000

アクセス : ワード、ハーフワード、およびバイトアクセス

APB2 ドメイン内のペリフェラルへのアクセス中にアクセスが発生する場合を除き、ノーウェイトステート。この場合、APB2 ペリフェラルへのアクセスが終了するまでウェイトステートが挿入されず。

注 : ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの値はソフトウェアによって読み出しできず、返される値は常に 0x0 になります。

| | | | | | | | | | | | | | | | |
|------|--------------|------|------------|------|------|-----------|------|------------|-----------|-------------|------|------|-------------|------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG EN | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | rw | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | USART1 EN | Res. | SPI1 EN | Res. | Res. | ADC EN | Res. | MIFI EN | Res. | TIM22 EN | Res. | Res. | TIM21 EN | Res. | SYSCFE N |
| | rw | | rw | | | rw | | | | rw | | | rw | | rw |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **DBGEN** : DBG クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : DBG クロック無効

1 : DBG クロック有効

ビット 21:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **USART1EN** : USART1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : USART1 クロック無効

1 : USART1 クロック有効

ビット 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **SPI1EN** : SPI1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SPI1 クロック無効

1 : SPI1 クロック有効

ビット 11:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **ADCCEN** : ADC クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : ADC クロック無効

1 : ADC クロック有効

ビット 8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **MIFIEN** : MiFaRe ファイアウォールクロックイネーブルビット

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。

0 : MIFI クロック無効

1 : MIFI クロック有効

ビット 6 予約済みであり、リセット値のままにしておかなければなりません。



ビット 5 **TIM22EN** : TIM22 タイマクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : TIM22 クロック無効

1 : TIM22 クロック有効

ビット 4:3 予約済みであり、リセット値のままにしておかなければなりません。

ビット 2 **TIM21EN** : TIM21 タイマクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : TIM21 クロック無効

1 : TIM21 クロック有効

ビット 1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **SYSCFGEN** : システム設定コントローラクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : システム設定コントローラクロック無効

1 : システム設定コントローラクロック有効

7.3.15 APB1 ペリフェラルクロック有効レジスタ (RCC_APB1ENR)

アドレス : 0x38

リセット値 : 0x0000 0000

アクセス : ワード、ハーフワード、およびバイトアクセス

APB1 ドメイン内のペリフェラルへのアクセス中にアクセスが発生する場合を除き、ノーウェイトステート。この場合、ウェイトステートは APB1 ペリフェラルへのアクセスが終了するまで挿入されず。

注 : ペリフェラルクロックがアクティブでない場合、ペリフェラルレジスタの値はソフトウェアによって読み出しできず、返される値は常に 0x0 になります。

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|--------------|------------|-------|-----------|------------|------|-------|------|-------|------------|------------|------------|------|---------------|--------------|------------|
| LPTIM1 EN | Res. | DACEN | PWR EN | CRSEN | Res. | Res. | Res. | USBEN | I2C2 EN | I2C1 EN | Res. | Res. | LPUART1 EN | USART2 EN | Res. |
| rw | | rw | rw | rw | | | | rw | rw | rw | | | rw | rw | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | SPI2 EN | Res. | Res. | WWDGE N | Res. | LCDEN | Res. | Res. | Res. | Res. | TIM6 EN | Res. | Res. | Res. | TIM2 EN |
| | rw | | | rw | | rw | | | | | rw | | | | rw |

ビット 31 **LPTIM1EN** : 低電力タイマクロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 低電力タイマクロック無効
1 : 低電力タイマクロック有効

ビット 30 予約済みであり、リセット値のままにしておかなければなりません。

ビット 29 **DACEN** : DAC インタフェースクロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : DAC インタフェースクロック無効
1 : DAC インタフェースクロック有効

ビット 28 **PWREN** : 電源インタフェースクロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : 電源インタフェースクロック無効
1 : 電源インタフェースクロック有効

ビット 27 **CRSEN** : クロックリカバリシステムクロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : クロックリカバリシステムクロック無効
1 : クロックリカバリシステムクロック有効

ビット 26:24 予約済みであり、リセット値のままにしておかなければなりません。

ビット 23 **USBEN** : USB クロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : USB クロック無効
1 : USB クロック有効

ビット 22 **I2C2EN** : I2C2 クロックイネーブルビット
このビットは、ソフトウェアによってセット／クリアされます。
0 : I2C2 クロック無効
1 : I2C2 クロック有効

ビット 21 **I2C1EN** : I2C1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : I2C1 クロック無効

1 : I2C1 クロック有効

ビット 20:19 予約済みであり、リセット値のままにしておかなければなりません。

ビット 18 **LPUART1EN** : LPUART1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : LPUART1 クロック無効

1 : LPUART1 クロック有効

ビット 17 **UART2EN** : UART2 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : UART2 クロック無効

1 : UART2 クロック有効

ビット 16:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **SPI2EN** : SPI2 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SPI2 クロック無効

1 : SPI2 クロック有効

ビット 13:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11 **WWDGEN** : ウィンドウ型ウォッチドッグクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : ウィンドウ型ウォッチドッグクロック無効

1 : ウィンドウ型ウォッチドッグクロック有効

ビット 10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **LCDEN** : LCD クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : LCD クロック無効

1 : LCD クロック有効

ビット 8:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **TIM6EN** : タイマ 6 クロックイネーブルビット

ソフトウェアでセット／クリアされます。

0 : タイマ 6 クロック無効

1 : タイマ 6 クロック有効

ビット 3:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **TIM2EN** : タイマ 2 クロックイネーブルビット

ソフトウェアでセット／クリアされます。

0 : タイマ 2 クロック無効

1 : タイマ 2 クロック有効

7.3.16 SLEEP モード時 GPIO クロックイネーブルレジスタ (RCC_IOPSMENR)

アドレス : 0x3C

リセット値 : 0x0000 008F

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|--------------|------|------|------|--------------|--------------|--------------|--------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IOPHSM EN | Res. | Res. | Res. | IOPDSM EN | IOPCSM EN | IOPBSM EN | IOPASM EN |
| | | | | | | | | rw | | | | rw | rw | rw | rw |

ビット 31 : 8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **IOPHSMEN** : SLEEP モード時ポート H クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ポート H クロックは無効です。

1 : SLEEP モード時、ポート H クロックは有効です (IOPHEN で有効にした場合)。

ビット 6:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **IOPDSMEN** : SLEEP モード時ポート D クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ポート D クロックは無効です。

1 : SLEEP モード時、ポート D クロックは有効です (IOPDEN で有効にした場合)。

ビット 2 **IOPCSMEN** : SLEEP モード時ポート C クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ポート C クロックは無効です。

1 : SLEEP モード時、ポート C クロックは有効です (IOPCEN で有効にした場合)。

ビット 1 **IOPBSMEN** : SLEEP モード時ポート B クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ポート B クロックは無効です。

1 : SLEEP モード時、ポート B クロックは有効です (IOPBEN で有効にした場合)。

ビット 0 **IOPASMEN** : SLEEP モード時ポート A クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ポート A クロックは無効です。

1 : SLEEP モード時、ポート A クロックは有効です (IOPAEN で有効にした場合)。

7.3.17 SLEEP モード時 AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBSMENR)

アドレス : 0x40

リセット値 : 0x0111 1301

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|------|-------------|------|------|--------------|--------------|------|------|------|-------------|------|------|------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | CRYP SMEN | Res. | Res. | Res. | RNGSM EN | Res. | Res. | Res. | TSCSME N |
| | | | | | | | rw | | | | rw | | | | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | CRC SMEN | Res. | Res. | SRAM SMEN | MIF SMEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DMA SMEN |
| | | | rw | | | rw | rw | | | | | | | | rw |

ビット 31:25 予約済みであり、リセット値のままにしておかなければなりません。

ビット 24 **CRYPTSMEN** : SLEEP モード時暗号クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、暗号クロックは無効です。

1 : SLEEP モード時、暗号クロックは有効です。

ビット 23:21 予約済みであり、リセット値のままにしておかなければなりません。

ビット 20 **RNGSMEN** : SLEEP モード時乱数発生回路クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、RNG クロックは無効です。

1 : SLEEP モード時、RNG クロックは有効です (RNGEN で有効にした場合)。

ビット 19:17 予約済みであり、リセット値のままにしておかなければなりません。

ビット 16 **TSCSMEN** : SLEEP モード時タッチセンシングクロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、タッチセンシングクロックは無効です。

1 : SLEEP モード時、タッチセンシングクロックは有効です (TOUCHEN で有効にした場合)。

ビット 15 : 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **CRCSMEN** : SLEEP モード時 CRC クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、テスト統合モジュールクロックは無効です。

1 : SLEEP モード時、テスト統合モジュールクロックは有効です (CRCEN で有効にした場合)。

ビット 11:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **SRAMSMEN** : SLEEP モード時 SRAM インタフェースクロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、NVM インタフェースクロックは無効です。

1 : SLEEP モード時、NVM インタフェースクロックは有効です。

ビット 8 **MIFSMEN** : SLEEP モード時 NVM インタフェースクロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、NVM インタフェースクロックは無効です。

1 : SLEEP モード時、NVM インタフェースクロックは有効です。

ビット 7:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **DMASMEN** : SLEEP モード時 DMA クロックイネーブルビット

このビットは、ソフトウェアによってセット/リセットされます。

0 : SLEEP モード時、DMA クロックは無効です。

1 : SLEEP モード時、DMA クロックは有効です。

7.3.18 SLEEP モード時 APB2 ペリフェラルクロックイネーブルレジスタ (RCC_APB2SMENR)

アドレス : 0x44

リセット値 : 0x0040 5225

アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|-------------|------|-----------|------|------|----------|------|------|----------|------------|------|------|------------|------|------------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG SMEN | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | rw | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | USART1 SMEN | Res. | SPI1 SMEN | Res. | Res. | ADC SMEN | Res. | Res. | Res. | TIM22 SMEN | Res. | Res. | TIM21 SMEN | Res. | SYSCFS MEN |
| | rw | | rw | | | rw | | | | rw | | | rw | | rw |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **DBGSMEN** : SLEEP モード時 DBG クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、DBG クロックは無効です。

1 : SLEEP モード時、DBG クロックは有効です (DBGEN で有効にした場合)。

ビット 21:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **USART1SMEN** : SLEEP モード時 USART1 クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、USART1 クロックは無効です。

1 : SLEEP モード時、USART1 クロックは有効です (USART1EN で有効にした場合)。

ビット 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **SPI1SMEN** : SLEEP モード時 SPI1 クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、SPI1 クロックは無効です。

1 : SLEEP モード時、SPI1 クロックは有効です (SPI1EN で有効にした場合)。

ビット 11:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **ADCSMEN** : SLEEP モード時 ADC クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、ADC クロックは無効です。

1 : SLEEP モード時、ADC クロックは有効です (ADCEN で有効にした場合)。



ビット 8:6 予約済みであり、リセット値のままにしておかなければなりません。

ビット 5 **TIM22SMEN** : SLEEP モード時 TIM22 タイマクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、TIM22 クロックは無効です。

1 : SLEEP モード時、TIM22 クロックは有効です (TIM22EN で有効にした場合)。

ビット 4:3 予約済みであり、リセット値のままにしておかなければなりません。

ビット 2 **TIM21SMEN** : SLEEP モード時 TIM21 タイマクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、TIM21 クロックは無効です。

1 : SLEEP モード時、TIM21 クロックは有効です (TIM21EN で有効にした場合)。

ビット 1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **SYSCFGSMEN** : SLEEP モード時システム設定コントローラクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、システム設定コントローラクロックは無効です。

1 : SLEEP モード時、システム設定コントローラクロックは有効です。

7.3.19 SLEEP モード時 APB1 ペリフェラルクロックイネーブルレジスタ (RCC_APB1SMENR)

アドレス : 0x48

リセット値 : 0xB8E6 4A11

注 : アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|-------------|-----------|----------|----------|----------|------|---------|------|----------|-----------|-----------|-----------|------|--------------|-------------|-----------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| LPTIM1 SMEN | Res. | DACSM EN | PWR SMEN | CRSSM EN | Res. | Res. | Res. | USBSM EN | I2C2 SMEN | I2C1 SMEN | Res. | Res. | LPUART1 SMEN | USART2 SMEN | Res. |
| rw | | rw | rw | rw | | | | rw | rw | rw | | | rw | rw | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | SPI2 SMEN | Res. | Res. | WWDGSMEN | Res. | LCDSMEN | Res. | Res. | Res. | Res. | TIM6 SMEN | Res. | Res. | Res. | TIM2 SMEN |
| | rw | | | rw | | rw | | | | | rw | | | | rw |

ビット 31 **LPTIM1SMEN** : SLEEP モード時低電力タイマクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、低電力タイマクロックは無効です。

1 : SLEEP モード時、低電力タイマクロックは有効です (LPTIM1EN で有効にした場合)。

ビット 30 予約済みであり、リセット値のままにしておかなければなりません。

ビット 29 **DACSMEN** : SLEEP モード時 DAC インタフェースクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、DAC インタフェースクロックは無効です。

1 : SLEEP モード時、DAC インタフェースクロックは有効です (DACEN で有効にした場合)。

ビット 28 **PWRSMEN** : SLEEP モード時電源インタフェースクロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。

0 : SLEEP モード時、電源インタフェースクロックは無効です。

1 : SLEEP モード時、電源インタフェースクロックは有効です (PWREN で有効にした場合)。

ビット 27 **CRSSMEN** : SLEEP モード時クロックリカバリシステムクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、クロックリカバリシステムクロックは無効です。

1 : SLEEP モード時、クロックリカバリシステムクロックは有効です (CRSEN で有効にした場合)。

ビット 26:24 予約済みであり、リセット値のままにしておかなければなりません。

ビット 23 **USBSMEN** : SLEEP モード時 USB クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、USB クロックは無効です。

1 : SLEEP モード時、USB クロックは有効です (USBEN で有効にした場合)。

ビット 22 **I2C2SMEN** : SLEEP モード時 I2C2 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、I2C2 クロックは無効です。

1 : SLEEP モード時、I2C2 クロックは有効です (I2C2EN で有効にした場合)。

ビット 21 **I2C1SMEN** : SLEEP モード時 I2C1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、I2C1 クロックは無効です。

1 : SLEEP モード時、I2C1 クロックは有効です (I2C1EN で有効にした場合)。

ビット 20:19 予約済みであり、リセット値のままにしておかなければなりません。

ビット 18 **LPUART1SMEN** : SLEEP モード時 LPUART1 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、LPUART1 クロックは無効です。

1 : SLEEP モード時、LPUART1 クロックは有効です (LPUART1EN で有効にした場合)。

ビット 17 **UART2SMEN** : SLEEP モード時 UART2 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、UART2 クロックは無効です。

1 : SLEEP モード時、UART2 クロックは有効です (UART2EN で有効にした場合)。

ビット 16:15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **SPI2SMEN** : SLEEP モード時 SPI2 クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、SPI2 クロックは無効です。

1 : SLEEP モード時、SPI2 クロックは有効です (SPI2SEN で有効にした場合)。

ビット 13:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11 **WWDGSMEN** : SLEEP モード時ウィンドウ型ウォッチドッグクロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、ウィンドウ型ウォッチドッグクロックは無効です。

1 : SLEEP モード時、ウィンドウ型ウォッチドッグクロックは有効です (WWDGEN で有効にした場合)。

ビット 10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **LCDSMEN** : SLEEP モード時 LCD クロックイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : SLEEP モード時、LCD クロックは無効です。

1 : SLEEP モード時、LCD クロックは有効です (LCDEN で有効にした場合)。

ビット 8:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **TIM6SMEN** : SLEEP モード時タイマ 6 クロックイネーブルビット
ソフトウェアでセット／クリアされます。
0 : SLEEP モード時、タイマ 6 クロックは無効です。
1 : SLEEP モード時、タイマ 6 クロックは有効です (TIM6EN で有効にした場合)。

ビット 3:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **TIM2SMEN** : SLEEP モード時 Timer2 クロックイネーブルビット
ソフトウェアでセット／クリアされます。
0 : SLEEP モード時、Timer2 クロックは無効です。
1 : SLEEP モード時、Timer2 クロックは有効です (TIM2EN で有効にした場合)。

7.3.20 クロック設定レジスタ (RCC_CCIPR)

アドレス : 0x4C
リセット値 : 0x0000 0000
アクセス : ノーウェイトステート、ワード、ハーフワード、およびバイトアクセス

| | | | | | | | | | | | | | | | |
|------|------|----------|----------|-------------|-------------|------|------|------|------|------|------|------------|------------|------------|------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | HSI48SEL | Res. | Res. | Res. | Res. | Res. | Res. | LPTIM1SEL1 | LPTIM1SEL0 | Res. | Res. |
| | | | | | rw | | | | | | | rw | rw | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | I2C1SEL1 | I2C1SEL0 | LPUART1SEL1 | LPUART1SEL0 | Res. | Res. | Res. | Res. | Res. | Res. | USART2SEL1 | USART2SEL0 | USART1SEL1 | USART1SEL0 |
| | | rw | rw | rw | rw | | | | | | | rw | rw | rw | rw |

ビット 31:27 予約済みであり、リセット値のままにしておかなければなりません。

ビット 26 **HSI48SEL** : 48 MHz HSI48 クロックソース選択ビット
このビットは、USB と RNG の HSI48 クロックソースを選択するために、ソフトウェアによってセット／クリアされます。
0 : PLL USB クロックが HSI48 クロックとして選択されます。
1 : RC48 クロックが HSI48 クロックとして選択されます。

ビット 25:20 予約済みであり、リセット値のままにしておかなければなりません。

ビット 19:18 **LPTIM1SEL** : 低電力タイマクロックソース選択ビット
このビットは、ソフトウェアによってセット／クリアされます。
00 : APB クロックが LP タイマクロックとして選択されます。
01 : LSI クロックが LP タイマクロックとして選択されます。
10 : HSI16 クロックが LP タイマクロックとして選択されます。
11 : LSE クロックが LP タイマクロックとして選択されます。

ビット 17:14 予約済みであり、リセット値のままにしておかなければなりません。

ビット 13:12 **I2C1SEL** : I2C1 クロックソース選択ビット
このビットは、ソフトウェアによってセット／クリアされます。
00 : APB クロックが I2C1 クロックとして選択されます。
01 : システムクロックが I2C1 クロックとして選択されます。
10 : HSI16 クロックが I2C1 クロックとして選択されます。
11 : 未使用



ビット 11:10 **LPUART1SEL** : LPUART1 クロックソース選択ビット

- このビットは、ソフトウェアによってセット／クリアされます。
- 00 : APB クロックが LPUART1 クロックとして選択されます。
- 01 : システムクロックが LPUART1 クロックとして選択されます。
- 10 : HSI16 クロックが LPUART1 クロックとして選択されます。
- 11 : LSE クロックが LPUART1 クロックとして選択されます。

ビット 9:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3:2 **USART2SEL** : USART2 クロックソース選択ビット

- このビットは、ソフトウェアによってセット／クリアされます。
- 00 : APB クロックが USART2 クロックとして選択されます。
- 01 : システムクロックが USART2 クロックとして選択されます。
- 10 : HSI16 クロックが USART2 クロックとして選択されます。
- 11 : LSE クロックが USART2 クロックとして選択されます。

ビット 1:0 **USART1SEL** : USART1 クロックソース選択ビット

- このビットは、ソフトウェアによってセット／クリアされます。
- 00 : APB クロックが USART1 クロックとして選択されます。
- 01 : システムクロックが USART1 クロックとして選択されます。
- 10 : HSI16 クロックが USART1 クロックとして選択されます。
- 11 : LSE クロックが USART1 クロックとして選択されます。

7.3.21 制御／ステータスレジスタ (RCC_CSR)

アドレス : 0x50

パワーオンリセット値 : 0x0C00 0000

アクセス : 0 ≤ ウェイトステート ≤ 3、ワード、ハーフワード、およびバイトアクセス

このレジスタへの連続したアクセスの場合、ウェイトステートが挿入されます。

注 : **RCC 制御およびステータスレジスタ (RCC_CSR) の LSEON、LSEBYP、RTCSEL、LSEDRV、および RTCEN ビットは、RTC ドメインにあります。これらのビットはリセット後に書き込み保護されるため、修正するには、電源制御レジスタ (PWR_CR) の DBP ビットをセットする必要があります。詳細については、セクション 6.1.3 : RTC および RTC バックアップレジスタを参照してください。これらのビットは、RTC ドメインリセット後にのみリセットされます (セクション 6.1.3 を参照)。内部または外部リセットは、これらのビットに影響しません。**

| | | | | | | | | | | | | | | | |
|--------------|--------------|--------------|-------------|-------------|-------------|-------------|-------|------|------|------|------|-----------------------|--------|-------------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| LPWR RSTF | WWDG RSTF | IWDG RSTF | SFT RSTF | POR RSTF | PIN RSTF | OBLRS TF | RMVF | Res. | Res. | Res. | Res. | RTC RST. </8913 | RTC EN | RTCSEL[1:0] | |
| rw | rw | rw | rw | rw | rw | rw | rw | | | | | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | CSSLSE D | CSSLSE ON | LSEDRV[1:0] | | LSE BYP | LSERDY | LSEON | Res. | Res. | Res. | Res. | Res. | Res. | LSI RDY | LSION |
| | rw | rw | rw | | rw | r | rw | | | | | | | r | rw |

ビット 31 LPWRRSTF : 低電力リセットフラグ

このビットは、低電力管理リセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むか、POR によってクリアされます。

0 : 低電力管理リセットは発生していません。

1 : 低電力管理リセットが発生しました。

低電力管理リセットの詳細については、[セクション : 低電力管理リセット](#)を参照してください。

ビット 30 WWDGRSTF : ウィンドウ型ウォッチドッグリセットフラグ

このビットは、ウィンドウ型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むか、POR によってクリアされます。

0 : ウィンドウ型ウォッチドッグリセットは発生していません。

1 : ウィンドウ型ウォッチドッグリセットが発生しました。

ビット 29 IWDGRSTF : 独立型ウォッチドッグリセットフラグ

このビットは、 V_{DD} ドメインから独立型ウォッチドッグリセットが発生したときに、ハードウェアによってセットされます。
RMVF ビットに書き込むか、POR によってクリアされます。

0 : ウォッチドッグリセットは発生していません。

1 : ウォッチドッグリセットが発生しました。

ビット 28 SFTRSTF : ソフトウェアリセットフラグ

このビットは、ソフトウェアリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むか、POR によってクリアされます。

0 : ソフトウェアリセットは発生していません。

1 : ソフトウェアリセットが発生しました。

ビット 27 PORRSTF : POR/PDR リセットフラグ

このビットは、POR/PDR リセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むことによってクリアされます。

0 : POR/PDR リセットは発生していません。

1 : POR/PDR リセットが発生しました。

ビット 26 PINRSTF : PIN リセットフラグ

このビットは、NRST ピンからリセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むか、POR によってクリアされます。

0 : NRST ピンからのリセットは発生していません。

1 : NRST ピンからリセットが発生しました。

ビット 25 OBLRSTF : オプションバイトローディングリセットフラグ

このビットは、OBL リセットが発生したときに、ハードウェアによってセットされます。

RMVF ビットに書き込むか、POR によってクリアされます。

0 : OBL リセットは発生していません。

1 : OBL リセットが発生しました。

ビット 24 RMVF : リセットフラグ解除

このビットは、リセットフラグをクリアするために、ソフトウェアによってセットされます。

0 : 影響なし。

1 : リセットフラグをクリアします。

ビット 23:20 予約済みであり、リセット値のままにしておかなければなりません。

ビット 19 RTCRST : RTC ソフトウェアリセットビット

このビットは、ソフトウェアによってセット／クリアされます。

0 : リセットは有効ではありません。

1 : RTC ペリフェラル、クロックソース選択、およびバックアップレジスタをリセットします。

ビット 18 RTCEN : RTC クロックイネーブルビット

このビットは、ソフトウェアによってセット/クリアされます。
RTCRST ビットをセットするか、POR によってリセットされます。
0 : RTC クロック無効
1 : RTC クロック有効

ビット 17:16 RTCSEL[1:0] : RTC および LCD クロックソース選択ビット

これらのビットは、RTC のクロックソースを選択するために、ソフトウェアによってセットされます。
RTC および LCD クロックソースが選択されると、RTCRST がセットされるかパワーオンリセットが発生するまで切り替えることはできません。唯一の例外は、LSE オシレータクロックが選択され、LSE クロックが停止したことが CSSHSE によって検出された場合で、この場合はクロックを切り替えることができます。

00 : クロックなし。

01 : LSE オシレータクロックが RTC/LCD クロックとして使用されます。

10 : LSI オシレータクロックが RTC/LCD クロックとして使用されます。

11 : プログラム可能なプリスケアラ (RCC クロック制御レジスタ (RCC_CR) の RTCPRE[1:0] ビットを使用して選択) で分周された HSE オシレータクロックが RTC/LCD クロックとして使用されます。

LSE または LSI が RTC クロックソースとして使用されている場合、RTC は引き続き 低電力の STOP および STANDBY モードで動作し、ウェイクアップソースとして使用できます。ただし、HSE クロックが RTC クロックソースとして使用される場合は、RTC を低電力の STOP および STANDBY モードで使用することはできません。

ビット 15 : 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 CSSLSED : LSE の CSS 障害検出フラグ

このビットは、外部 32 kHz オシレータ (LSE) のクロックセキュリティシステムによって障害が検出されたことを示すために、ハードウェアによってセットされます。

パワーオンリセットまたは RTC ソフトウェアリセット (RTCRST ビット) によってクリアされます。

0 : LSE (32 kHz オシレータ) に障害が検出されません。

1 : LSE (32 kHz オシレータ) に障害が検出されました。

ビット 13 CSSLSEON : LSE イネーブルビットの CSS

このビットは、LSE (32 kHz オシレータ) のクロックセキュリティシステムを有効にするために、ソフトウェアによってセットされます。

CSSLSEON は、LSE オシレータと LSI オシレータを有効 (LSEON および LSION ビットを有効化) かつレディ状態 (ハードウェアで LSERDY および LSIRDY フラグをセット) にし、RTCSEL ビットを選択した後で、有効にする必要があります。

このビットは、一度有効にすると、LSE の障害検出 (CSSLSED = 1) の後を除き、無効にすることはできません。この場合、CSSLSEON ビットをソフトウェアで無効にする必要があります。

パワーオンリセットおよび RTC ソフトウェアリセット (RTCRST ビット) によってリセットできます。

0 : LSE (32 kHz オシレータ) の CSS オフ

1 : LSE (32 kHz オシレータ) の CSS オン

ビット 12~11 LSEDRV : LSE オシレータの駆動能力ビット

これらのビットは、LSE オシレータの駆動能力を選択するために、ソフトウェアによってセットされます。

パワーオンリセットまたは RTC リセットによってクリアされます。一度「00」が書き込まれると、ソフトウェアで LSEDRV の内容を変更することはできません。

00 : 最低駆動

01 : 中低駆動

10 : 中高駆動

11 : 最高駆動

ビット 10 **LSEBYP** : 外部低速オシレータバイパスビット

このビットは、デバッグモードのオシレータをバイパスするために、ソフトウェアによってセット／クリアされます。このビットに書き込めるのは、LSE オシレータが無効なときだけです。
RTRCRST ビットをセットするか、POR によってリセットされます。
0 : LSE オシレータはバイパスされません。
1 : LSE オシレータはバイパスされます。

ビット 9 **LSERDY** : 外部低速オシレータレディビット

このビットは、LSE オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSEON ビットがクリアされた後、LSERDY は、LSE オシレータクロックの 6 サイクル後にローになります。
RTRCRST ビットをセットするか、POR によってリセットされます。
0 : 外部 32 kHz オシレータはレディ状態ではありません。
1 : 外部 32 kHz オシレータはレディ状態です。

ビット 8 **LSEON** : 外部低速オシレータイネーブルビット

このビットは、ソフトウェアによってセット／クリアされます。
RTRCRST ビットをセットするか、POR によってリセットされます。
0 : LSE オシレータオフ
1 : LSE オシレータオン

ビット 7:2 予約済みであり、リセット値のままにしておかなければなりません。

ビット 1 **LSIRDY** : 内部低速オシレータレディビット

このビットは、LSI オシレータが安定していることを示すために、ハードウェアによってセット／クリアされます。LSION ビットがクリアされた後、LSIRDY は、LSI オシレータクロックの 3 サイクル後にローになります。
このビットは、システムリセットによってリセットされます。
0 : LSI オシレータはレディ状態ではありません。
1 : LSI オシレータはレディ状態です。

ビット 0 **LSION** : 内部低速オシレータイネーブル

このビットは、ソフトウェアによってセット／クリアされます。
システムリセットによってリセットされます。
0 : LSI オシレータオフ
1 : LSI オシレータオン

7.3.22 RCC レジスタマップ

次の表に、RCC レジスタマップとリセット値を示します。

表 33. RCC レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-----------|--------------|------|------|------|------|------|---------|--------|-------------|------|---------------------|----------|--------|--------|-------|------|-------------------|------|------|------|----------------|------|--------|-------|---------------|------|------|-----------|------------|-----------|------------|---------|
| 0x00 | RCC_CR | Res. | Res. | Res. | Res. | Res. | Res. | PLL RDY | PLL ON | Res. | Res. | RTC PRE [1:0] | CSSLSEON | HSEBYP | HSERDY | HSEON | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MSIRDY | MSION | Res. | Res. | Res. | HSI16DIVF | HSI16DIVEN | HSI16RDYF | HSI16KERON | HSI16ON |
| | リセット値 | | | | | | | 0 | 0 | | | X | X | 0 | X | 0 | 0 | | | | | | | 1 | 1 | | | | 0 | 0 | 0 | 0 | 0 |
| 0x04 | RCC_ICSCR | MSITRIM[7:0] | | | | | | | | MSICAL[7:0] | | | | | | | | MSIRAN GE[2:0] | | | | HSI16TRIM[4:0] | | | | HSI16CAL[7:0] | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x |



表 33. RCC レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | |
|-----------|--------------|-----------|-----------------|--------|--------|--------|-----------------|----|----|---------------------|---------|-------------|----|----|----|----|--------------------|---------------|-----------|----------------|----|----|----------------|---|-----------|---------|------------|--------------|----------|-------------|------------|-----------|----------|---------|--|--|--|--|----------|---------|
| 0x08 | RCC_CRRCR | Res. | | | | | | | | | | | | | | | | HSI48CAL[7:0] | | | | | | | | | | | | | | | | | | | | | HSI48RDY | HSI48ON |
| | Reset value | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | 0 | 0 | | | | | | |
| 0x0C | RCC_CFGR | Res. | MCOPRE [2:0] | | | | MCOSEL [3:0] | | | PLL DIV [1:0] | | PLLMUL[3:0] | | | | | PLLSRC STOPWUOK | | | PPRE2 [2:0] | | | PPRE1 [2:0] | | HPRE[3:0] | | | SWS [1:0] | | SW [1:0] | | | | | | | | | | |
| | リセット値 | | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x10 | RCC_CIER | Res. | | | | | | | | | | | | | | | | | | | | | | | | CSSLSE | HSI48RDYIE | MSIRDYIE | PLLRDYIE | HSERDYIE | HSI16RDYIE | LSERDYIE | LSIRDYIE | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x14 | RCC_CIFR | Res. | | | | | | | | | | | | | | | | | | | | | | | | CSSHSEF | CSSLSEF | HSI48RDYF | MSIRDYF | PLLRDYF | HSERDYF | HSI16RDYF | LSERDYF | LSIRDYF | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x18 | RCC_CICR | Res. | | | | | | | | | | | | | | | | | | | | | | | | CSSHSEC | CSSLSEC | HSI48RDYC | MSIRDYC | PLLRDYC | HSERDYC | HSI16RDYC | LSERDYC | LSIRDYC | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x1C | RCC_IOPRSTR | Res. | | | | | | | | | | | | | | | | | | | | | | | | IOPHRST | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | | | | | | | | | | |
| 0x20 | RCC_AHBRSTR | Res. | | | | | | | | CRYPRST | | | | | | | TSCRST | | | | | | | | | MIFRST | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | 0 | | | | 0 | | | | 0 | | | | | | | | 0 | | | | | | | | | | | | | | | |
| 0x24 | RCC_APB2RSTR | Res. | | | | | | | | DBGFRST | | | | | | | | | USART1RST | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | 0 | | | | | | | | | 0 | | 0 | | | | 0 | | | | | | | | | | | | | | | |
| 0x28 | RCC_APB1RSTR | LPTIM1RST | | DACRST | PWRRST | CRSRST | | | | USBRST | I2C2RST | I2C1RST | | | | | | | SPI2RST | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | | 0 | 0 | 0 | | | | 0 | 0 | 0 | | | 0 | 0 | | | 0 | | | | 0 | | 0 | | | | 0 | | | | | | | | | | | |
| 0x2C | RCC_IOPENR | Res. | | | | | | | | | | | | | | | | | | | | | | | | IOPEN | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x30 | RCC_AHBENR | Res. | | | | | | | | CRYPEN | | | | | | | | | | | | | | | | | | | | | | | | DMAEN | | | | | | |
| | リセット値 | | | | | | | | 0 | | | | 0 | | | | 0 | | | | | | | | 1 | | | | | | | | | | | | | | | |

表 33. RCC レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----------|-------------------|------------|----------|----------|---------|---------|----------|---------|---------|---------|----------|----------|---------|-----------------|-------------|------------|---------|---------|------------|---------|----------|----------|--------|----------|---------|----------|------|---------|-----------|----------|------------|------------|-------------|------------|
| 0x34 | RCC_APB2ENR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBGEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | USART1EN | Res. | SPI1EN | Res. | Res. | ADCEN | Res. | MFIEN | Res. | TIM22EN | Res. | Res. | TIM21EN | Res. | SYS_CFGEN | |
| | リセット値 | | | | | | | | | | 0 | | | | | | | | 0 | | 0 | | | 0 | | 0 | | 0 | | | 0 | | 0 | |
| 0x38 | RCC_APB1ENR | LPTIM1EN | Res. | DACEN | PWREN | CRSEN | Res. | Res. | Res. | USBEN | I2C2EN | I2C1EN | Res. | Res. | LPUART1EN | USART2EN | Res. | Res. | SPI2EN | Res. | Res. | VWDGEN | Res. | Res. | LCDEN | Res. | Res. | Res. | Res. | TIM6EN | Res. | Res. | TIM2EN | |
| | リセット値 | 0 | | 0 | 0 | 0 | | | | 0 | 0 | 0 | | | 0 | 0 | | | 0 | | | 0 | | | 0 | | | | | 0 | | | 0 | |
| 0x3C | RCC_IOPSMEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IOPHSMEN | Res. | Res. | Res. | IOPDSMEN | Res. | IOPBSMEN | IOPASMEN | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 1 | | | | 1 | | 1 | | 1 |
| 0x40 | RCC_AHBSM ENR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CRYSMEN | Res. | Res. | Res. | RNGSMEN | Res. | Res. | Res. | TSCSMEN | Res. | Res. | CRCSMEN | Res. | Res. | Res. | SRAMSMEN | MIFSMEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DMASMEN | |
| | リセット値 | | | | | | | | 1 | | | | 1 | | | | 1 | | | | 1 | | | 1 | 1 | | | | | | | | 1 | |
| 0x44 | RCC_APB2SM ENR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBGSMEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | USART1SMEN | Res. | SPI1SMEN | Res. | Res. | ADCSMEN | Res. | Res. | Res. | Res. | TIM22SMEN | Res. | Res. | TIM2SMEN | SYS_CFGSMEN | |
| | リセット値 | | | | | | | | | | 1 | | | | | | | | 1 | | 1 | | | 1 | | | | | 1 | | | 1 | | 1 |
| 0x48 | RCC_APB1 SMENR | LPTIM1SMEN | Res. | DACSMEN | PWRSMEN | CRSSMEN | Res. | Res. | Res. | USBSMEN | I2C2SMEN | I2C1SMEN | Res. | Res. | LPUART1SMEN | USART2SMEN | Res. | Res. | SPI2SMEN | Res. | Res. | VWDGSMEN | Res. | Res. | LCDSMEN | Res. | Res. | Res. | Res. | Res. | TIM6SMEN | Res. | Res. | TIM2SMEN |
| | リセット値 | 1 | | 1 | 1 | 1 | | | | 1 | 1 | 1 | | | 1 | 1 | | | 1 | | | 1 | | 1 | | 1 | | | | 1 | | | 1 | |
| 0x4C | RCC_CCIPR | Res. | Res. | Res. | Res. | Res. | HSI48SEL | Res. | Res. | Res. | Res. | Res. | Res. | LPTIM1SEL1 | LPTIM1SEL0 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | USART2SEL1 | USART2SEL0 | USART1SEL1 | USART1SEL0 |
| | リセット値 | | | | | | 0 | | | | | | | 0 | 0 | | | | | 0 | 0 | 0 | 0 | | | | | | | 0 | 0 | 0 | 0 | |
| 0x50 | RCC_CSR | LPWRSTF | VWDGRSTF | IWDGRSTF | SFTRSTF | PORRSTF | PINRSTF | OBLRSTF | RMVF | Res. | Res. | Res. | Res. | RTCSEL [1:0] | Res. | Res. | Res. | CSSLSED | CSSLSEON | Res. | Res. | Res. | LSERDY | LSERDY | LSERDY | Res. | Res. | Res. | Res. | Res. | Res. | LSIRDY | LSION | |
| | リセット値 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

8 クロックリカバリシステム (CRS)

8.1 概要

クロックリカバリシステム (CRS) は、高精度トリミング可能な内蔵 RC オシレータ HSI48 に作用する、高度なデジタルコントローラです。選択可能な同期信号との比較に基づいて、オシレータ出力周波数チェックを行うための強力な手段を提供しています。CRS では、測定した周波数の誤差値に基づいてオシレータのトリミングを自動調整することが可能です。また、手動でトリミングすることも可能です。

CRS は、USB ペリフェラルに高精度なクロックを提供する場合に最適です。この場合、USB ホストによって正確に 1 ms 間隔で送信される USB バス上のフレーム開始 (SOF) パケット信号から同期信号を取得できます。

同期信号は、LSE オシレータ出力または外部ピンからも取得することができ、またユーザソフトウェアから生成することもできます。

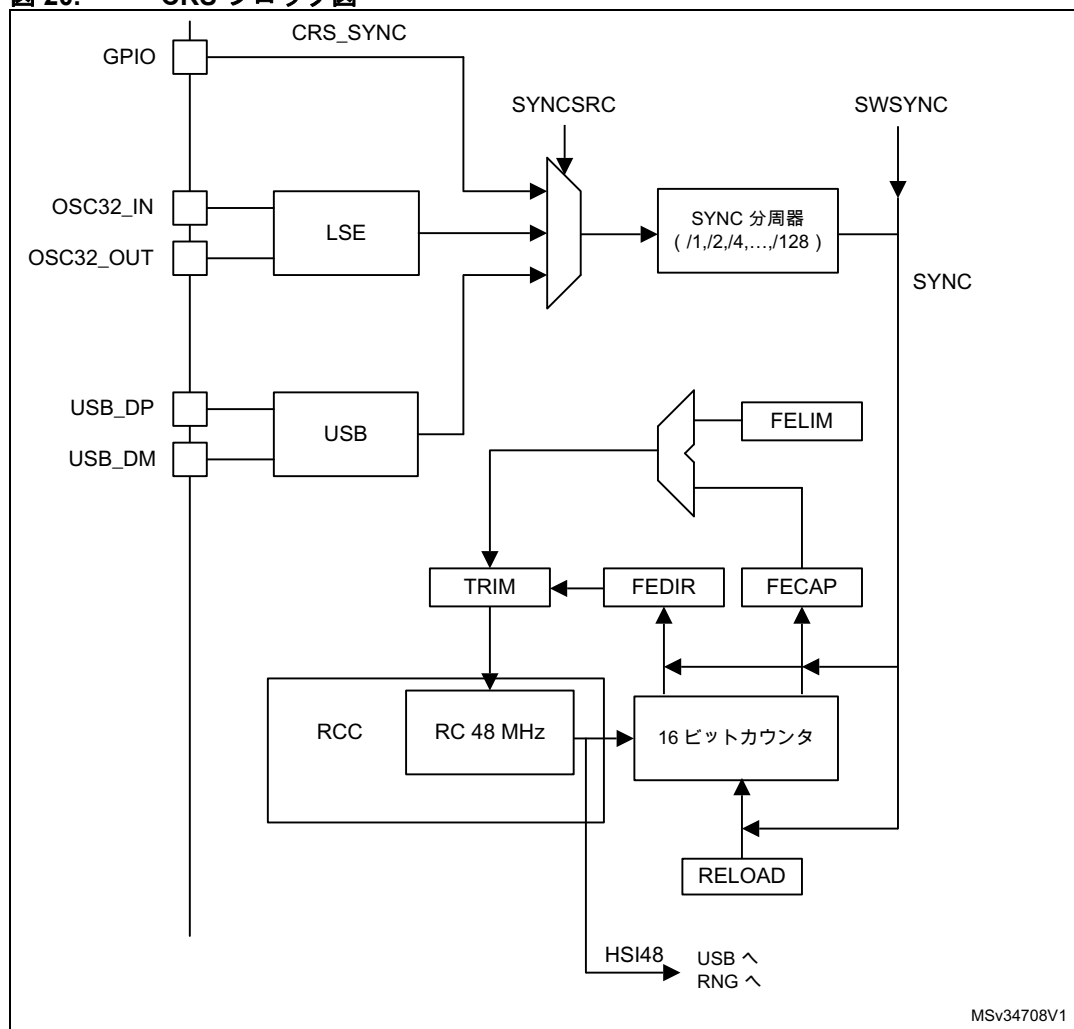
8.2 CRS の主な機能

- 次のプログラム可能なプリスケールと極性を持つ、選択可能な同期ソース：
 - － 外部ピン
 - － LSE オシレータ出力
 - － USB SOF パケット受信
- ソフトウェアによって同期パルスを生成する可能性
- CPU 不要の自動オシレータトリミング機能
- より迅速なスタートアップのための手動制御オプション
- 自動誤差値のキャプチャと再ロードが可能な、16 ビットの周波数誤差カウンタ
- 自動周波数誤差値の評価とステータスレポートの、プログラム可能な制限値
- マスク可能な割り込み／イベント：
 - － 期待される同期 (ESYNC)
 - － 同期 OK (SYNCOK)
 - － 同期警告 (SYNCWARN)
 - － 同期またはトリミングエラー (ERR)

8.3 CRS の機能詳細

8.3.1 CRS ブロック図

図 20. CRS ブロック図



8.3.2 同期入力

CRS_CFGR レジスタから選択可能な CRS 同期 (SYNC) ソースは、外部 CRS_SYNC ピンからの信号、LSE クロック、または USB SOF 信号のいずれかです。より安定した SYNC 入力を得るには、グリッチをフィルタするために、シンプルなデジタルフィルタ (3 対 2 の多数決方式、HSI48 クロックによるサンプリング方法) が導入されています。このソース信号には設定可能な極性があり、適切な周波数範囲 (通常 1 kHz 前後) の同期信号を取得するために、プログラム可能なバイナリプリスケアラによって分周することができます。

CRS 同期ソース設定の詳細については、[セクション 8.6.2 : CRS 設定レジスタ \(CRS_CFGR\)](#) を参照してください。

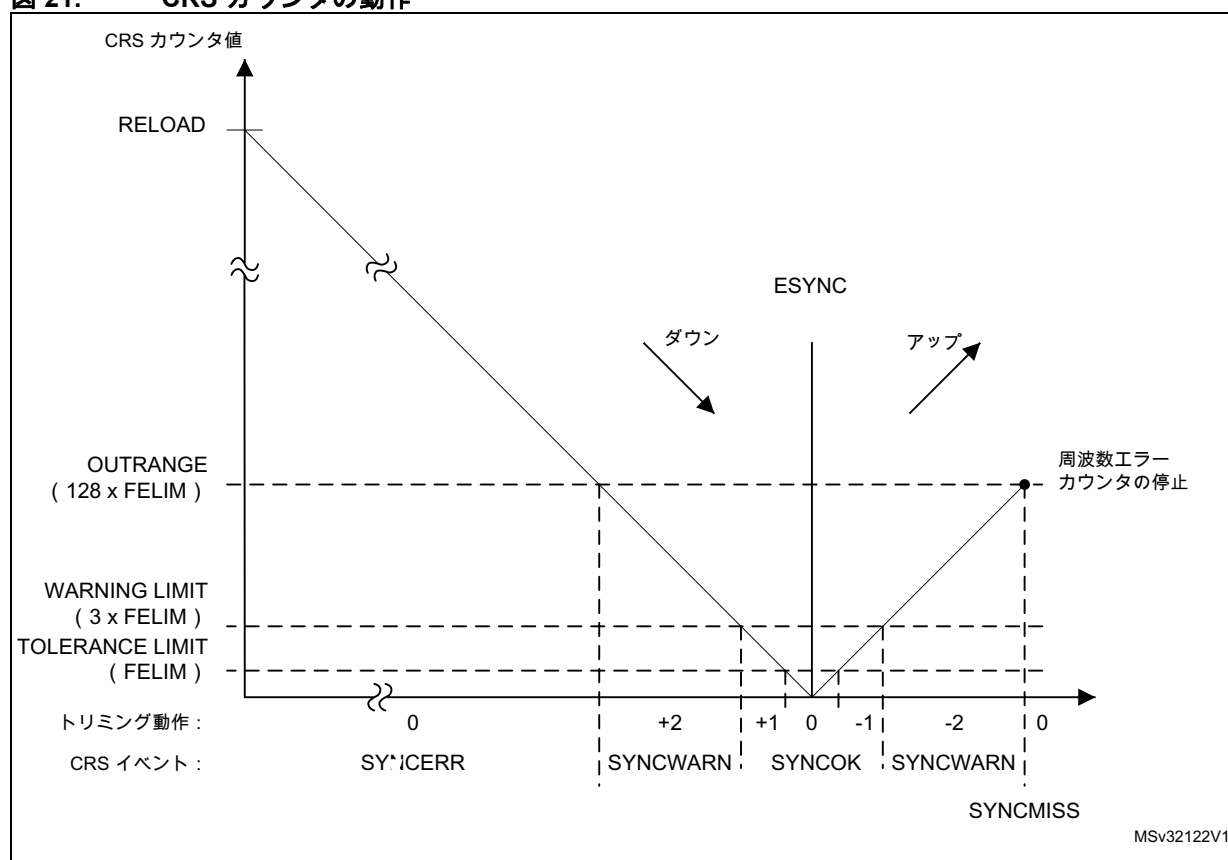
CRS_CR レジスタの SWSYNC ビットをセットして、ソフトウェアで同期イベントを生成することもできます。

8.3.3 周波数誤差測定

この周波数誤差カウンタは、各 SYNC イベントで RELOAD 値によって再ロードされる 16 ビットのダウン／アップカウンタです。ESYNC（期待される同期）イベントが生成されるゼロ値に達するまでカウントダウンします。次に、OUTRANGE 制限によって最終的に停止する（SYNC イベントが受信されない場合）までカウントアップし、SYNCMISS イベントを生成します。OUTRANGE 制限は、周波数誤差制限 (CRS_CFGR レジスタの FELIM フィールド) に 128 を乗算した値として定義されます。

SYNC イベントが検出されると、周波数誤差カウンタの実際の値とそのカウント方向が FECAP（周波数誤差キャプチャ）フィールドと CRS_ISR レジスタの FEDIR（周波数誤差方向）ビットに保管されます。ダウンカウントフェーズで（ゼロ値に達する前に）SYNC イベントが検出されたときは、実際の周波数がターゲットよりも小さいことを意味します（そのため、TRIM 値をインクリメントする必要があります）。アップカウントフェーズ中に検出されると、実際の周波数は高くなります（そのため、TRIM 値をデクリメントする必要があります）。

図 21. CRS カウンタの動作



8.3.4 周波数誤差評価および自動トリミング

測定した周波数誤差は、その値といくつかの制限を比較して評価されます。

- CRS_CFGR レジスタの FELIM フィールドに直接与えられる TOLERANCE LIMIT
- $3 * \text{FELIM}$ 値として定義される WARNING LIMIT
- $128 * \text{FELIM}$ 値として定義される OUTRANGE (エラー制限)

この比較の結果は、ステータスの表示を生成し、CRS_CR レジスタの AUTOTRIMEN ビットをセットすることで有効化される自動トリミングを制御するために使用されます。

- 周波数誤差が許容制限を下回る場合は、TRIM フィールドの実際のトリミング値が適切であり、トリミング動作が不要であることを意味します。
 - SYNCOK ステータスが示されます。
 - TRIM 値は、AUTOTRIM モードでは変更されません。
- 周波数誤差が警告制限を下回り、許容制限を上回るか同じである場合は、なんらかのトリミング動作が必要であることを意味します。ただし、1 回のトリミングステップによる調整で、十分適切な TRIM 値に達することができます。
 - SYNCOK ステータスが示されます。
 - AUTOTRIM モードで、1 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が警告制限を上回るか同じであり、エラー制限を下回る場合は、より強力なトリミング動作が必要であることを意味します。次の周期までに適切な TRIM 値に達することができない危険性もあります。
 - SYNCWARN ステータスが示されます。
 - AUTOTRIM モードで、2 回のトリミングステップによって TRIM 値が調整されます。
- 周波数誤差が誤差制限を上回るか同じである場合は、周波数がトリミング範囲外であることを意味します。これは、SYNC 入力がクリーンではなく、SYNC パルスが一部欠落している場合に発生する可能性があります (たとえば、USB SOF が 1 つ破損している場合)。
 - SYNCERR または SYNCMISS ステータスが示されます。
 - TRIM 値は、AUTOTRIM モードでは変更されません。

注 : *TRIM フィールドの実際の値が制限値に近く、自動トリミングによってオーバーフローまたはアンダーフローが強制される場合、TRIM 値が誤差ちょうどに設定され、TRIMOVF ステータスが示されます。AUTOTRIM モード (CRS_CR レジスタにセットされた AUTOTRIMEN ビット) では、CRS_CR の TRIM フィールドがハードウェアによって調整され、読み出し専用になります。*

8.3.5 CRS の初期化と設定

RELOAD 値

RELOAD 値は、ターゲット周波数とプリスケール後の同期ソースの周波数間の比率に従って選択します。ゼロ値で期待される同期に達するために、1 ずつ減算します。計算式は、次のとおりです。

$$\text{RELOAD} = (f_{\text{TARGET}} / f_{\text{SYNC}}) - 1$$

RELOAD フィールドのリセット値は、48 MHz のターゲット周波数と 1 kHz の同期信号周波数 (USB からの SOF 信号) に対応します。

FELIM 値

FELIM 値の選択は、HSI48 発振特性とその標準的なトリミングステップサイズに、深く関係しています。最適値は、HSI48 オシレータクロック数を表すトリミングステップサイズの半分の値に対応します。次の計算式を使用できます。

$$FELIM = (f_{TARGET} / f_{SYNC}) * STEP[\%] / 100\% / 2$$

最高のトリム応答を取得するために、結果の小数点以下の値は常に切り上げられます。アプリケーション内でトリミング動作を頻繁に行う必要がない場合は、FELIM 値を少し上げることでトリミングのヒステリシスを増やすことができます。

FELIM フィールドのリセット値は、 $(f_{TARGET} / f_{SYNC}) = 48000$ と標準的なトリミングステップサイズ (0.14%) に対応しています。

注意 : RELOAD および FELIM フィールドの誤った設定に対するハードウェアの保護はなく、不規則なトリム応答を引き起こす可能性があります。期待される動作モードでは、RELOAD 値を（同期ソース周波数に対して）適切に設定することを求めます。この値は、128 * FELIM 値よりも大きい値です (OUTRANGE 制限)。

8.4 CRS 低電力モード

表 34. 低電力モードが CRS に与える影響

| モード | 説明 |
|---------|---|
| SLEEP | 影響はありません。 CRS 割り込みによって、デバイスは SLEEP モードから復帰します。 |
| STOP | CRS レジスタを停止します。 |
| STANDBY | CRS によって、動作は STOP または STANDBY モードが終了し、HSI48 オシレータがリスタートするまで停止します。 |

8.5 CRS 割り込み

表 35. 割り込み制御ビット

| 割り込みイベント | イベントフラグ | 制御ビットの有効化 | フラグビットのクリア |
|--|-----------|------------|------------|
| 期待される同期 | ESYNCF | ESYNCE | ESYNCC |
| 同期 OK | SYNCOKF | SYNCOKE | SYNCOKC |
| 同期警告 | SYNCWARNF | SYNCWARNIE | SYNCWARNC |
| 同期またはトリミングエラー (TRIMOVF、SYNCMISS、SYNCERR) | ERRF | ERRIE | ERRC |

8.6 CRS レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの [セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスすることができます。

8.6.1 CRS 制御レジスタ (CRS_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 2000

| | | | | | | | | | | | | | | | |
|------|------|-----------|------|------|------|------|------|------------|----------------|------|------|-------------|-------|----------------|--------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | TRIM[5:0] | | | | | | SWSYN C | AUTOTR IMEN | CEN | Res. | ESYNCI E | ERRIE | SYNCW ARNIE | SYNCO KIE |
| | | rw | rw | rw | rw | rw | rw | rl_w | rw | rw | | rw | rw | rw | rw |

ビット 31:14 予約済みであり、リセット値のままにしておかなければなりません。

ビット 13:8 **TRIM[5:0]** : HSI48 オシレータ高精度トリミング

これらのビットにより、HSI48 オシレータにユーザプログラム可能なトリミング値が使用できます。HSI48 の周波数に影響する電圧や温度の変化に対応できるようにプログラミングできます。デフォルト値は 32 (トリミングインターバルの中間に相当) です。トリミングステップは、2 つの連続する TRIM ステップの間で、67 kHz 前後です。高い TRIM 値は、高い出力周波数に対応します。AUTOTRIMEN ビットがセットされると、このフィールドはハードウェアによって調整され、読み出し専用になります。

ビット 7 **SWSYNC** : ソフトウェア SYNC イベントの生成

このビットは、ソフトウェア SYNC イベントを生成するためにソフトウェアによってセットされます。ハードウェアによって自動的にクリアされます。
0 : 影響なし。
1 : ソフトウェア SYNC イベントが生成されます。

ビット 6 **AUTOTRIMEN** : 自動トリミング有効

このビットは、2 つの SYNC イベント間で測定した周波数誤差に応じて、TRIM ビットの自動ハードウェア調整を有効にします。このビットがセットされると、TRIM ビットが読み出し専用になります。TRIM 値は、測定した周波数誤差値に応じて、ハードウェアで 1 度に 1 または 2 ステップずつ調整できます。詳細については、[セクション 8.3.4 : 周波数誤差評価および自動トリミング](#)を参照してください。
0 : 自動トリミングは無効です。ユーザは、TRIM ビットを調整できます。
1 : 自動トリミングは有効です。TRIM ビットは読み出し専用で、ハードウェアによって制御されます。

ビット 5 **CEN** : 周波数誤差カウンタ有効

このビットは、周波数誤差カウンタのオシレータクロックを有効にします。
0 : 周波数誤差カウンタは無効です。
1 : 周波数誤差カウンタは有効です。
このビットをセットすると、CRS_CFGR レジスタは書き込み保護され、変更できません。

ビット 4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **ESYNCE** : 期待される SYNC 割り込み有効

0 : 期待される SYNC (ESYNCF) 割り込みは無効です。
1 : 期待される SYNC (ESYNCF) 割り込みは有効です。

ビット 2 **ERRIE** : 同期またはトリミングエラー割り込み有効

0 : 同期またはトリミングエラー (ERRF) 割り込みは無効です。

1 : 同期またはトリミングエラー (ERRF) 割り込みは有効です。

ビット 1 **SYNCWARNIE** : SYNC 警告割り込み有効

0 : SYNC 警告 (SYNCWARNF) 割り込みは無効です。

1 : SYNC 警告 (SYNCWARNF) 割り込みは有効です。

ビット 0 **SYNCOKIE** : SYNC イベント OK 割り込み有効

0 : SYNC イベント OK (SYNCOKF) 割り込みは無効です。

1 : SYNC イベント OK (SYNCOKF) 割り込みは有効です。

8.6.2 CRS 設定レジスタ (CRS_CFGR)

このレジスタに書き込めるのは、周波数誤差カウンタが無効なときだけです (CEN ビットは CRS_CR でクリアされる)。このカウンタを有効にすると、このレジスタは書き込み保護されます。

アドレスオフセット : 0x04

リセット値 : 0x2022 BB7F

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

- ビット 31 **SYNCPOL** : SYNC 極性選択
- このビットは、SYNC 信号ソースの入力の極性を選択するために、ソフトウェアによってセット／クリアされます。
- 0 : SYNC は立ち上がりエッジでアクティブです (デフォルト)。
- 1 : SYNC は立ち下がりエッジでアクティブです。
- ビット 30 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 29:28 **SYNCSRC[1:0]** : SYNC 信号ソース選択
- これらのビットは、SYNC 信号ソースを選択するために、ソフトウェアによってセット／クリアされます。
- 00 : SYNC 信号ソースとして GPIO を選択します。
- 01 : SYNC 信号ソースとして LSE を選択します。
- 10 : SYNC 信号ソースとして USB SOF を選択します (デフォルト)。
- 11 : 予約済み
- ビット 27 予約済みであり、リセット値のままにしておかなければなりません。

- ビット 26:24 **SYNCDIV[2:0]** : SYNC 分周器
- これらのビットは、SYNC 信号の分周比を制御するために、ソフトウェアによってセット／クリアされます。
- 000 : SYNC は分周されません (デフォルト)。
- 001 : SYNC は 2 分周されます。
- 010 : SYNC は 4 分周されます。
- 011 : SYNC は 8 分周されます。
- 100 : SYNC は 16 分周されます。
- 101 : SYNC は 32 分周されます。
- 110 : SYNC は 64 分周されます。
- 111 : SYNC は 128 分周されます。
- ビット 23:16 **FELIM[7:0]** : 周波数誤差制限
- FELIM には、CRS_ISR レジスタの FECAP[15:0] ビットにラッチされた、キャプチャされた周波数誤差値を評価するために使用する値が含まれます。FECAP 評価の詳細については、[セクション 8.3.4 : 周波数誤差評価および自動トリミング](#)を参照してください。
- ビット 15:0 **RELOAD[15:0]** : カウンタ再ロード値
- RELOAD は、SYNC イベントごとに周波数誤差カウンタにロードされる値です。
- カウンタの動作の詳細については、[セクション 8.3.3 : 周波数誤差測定](#)を参照してください。

8.6.3 CRS 割り込みおよびステータスレジスタ (CRS_ISR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|------|------|------|------|---------|----------|---------|------|------|------|------|--------|------|-----------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| FECAP[15:0] | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| FEDIR | Res. | Res. | Res. | Res. | TRIMOVF | SYNCMISS | SYNCERR | Res. | Res. | Res. | Res. | ESYNCF | ERRF | SYNCWARNF | SYNCOKF |
| r | | | | | r | r | r | | | | | r | r | r | r |

ビット 31:16 **FECAP[15:0]** : 周波数誤差キャプチャ

FECAP は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタ値です。
FECAP の使用の詳細については、[セクション 8.3.4 : 周波数誤差評価および自動トリミング](#)を参照してください。

ビット 15 **FEDIR** : 周波数誤差方向

FEDIR は、最後の SYNC イベント時にラッチされた、周波数誤差カウンタのカウント方向です。実際の周波数が、ターゲットを下回るか上回るかを示します。
0 : アップカウント方向。実際の周波数はターゲットを上回ります。
1 : ダウンカウント方向。実際の周波数はターゲットを下回ります。

ビット 14:11 予約済みであり、リセット値のままにしておかなければなりません。

ビット 10 **TRIMOVF** : トリミングのオーバーフローまたはアンダーフロー

このフラグは、自動トリミングによって TRIM 値のオーバーフローまたはアンダーフローが試みられた場合に、ハードウェアによってセットされます。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。
0 : トリミングエラーを信号化しません。
1 : トリミングエラーを信号化します。

ビット 9 **SYNCMISS** : SYNC の欠落

このフラグは、周波数誤差カウンタが値 FELIM * 128 に達し、SYNC が検出されない場合にハードウェアによってセットされます。これは、SYNC パルスが失われているということ、または TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が高すぎる）、他の操作を行う必要があるということを意味します。この時点で、周波数誤差カウンタは停止し（次の SYNC を待つ）、CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。
0 : SYNC の欠落エラーを信号化しません。
1 : SYNC の欠落エラーを信号化します。

ビット 8 **SYNCERR** : SYNC エラー

このフラグは、SYNC パルスが ESYNC イベントより前に発生し、測定した周波数誤差が FELIM * 128 以上である場合に、ハードウェアによってセットされます。これは、TRIM 値を調整して補正するには周波数誤差が大きすぎ（内部周波数が低すぎる）、他の操作を行う必要があるということを意味します。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRC ビットをセットすることで、ソフトウェアによってクリアされます。
0 : SYNC エラーを信号化しません。
1 : SYNC エラーを信号化します。

ビット 7:4 予約済みであり、リセット値のままにしておかなければなりません。



ビット 3 **ESYNCF** : 期待される SYNC フラグ

このフラグは、周波数誤差カウンタがゼロ値に達したときに、ハードウェアによってセットされます。CRS_CR レジスタの ESYNCF ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ESYNCC ビットをセットすることで、ソフトウェアによってクリアされます。

0 : 期待された SYNC を信号化しません。

1 : 期待された SYNC を信号化します。

ビット 2 **ERRF** : エラーフラグ

このフラグは、同期またはトリミングエラー時に、ハードウェアによってセットされます。TRIMOVF、SYNCMISS、および SYNCERR ビットの論理和です。CRS_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの ERRF ビットをセットすることへの反応として、ソフトウェアによってクリアされます。TRIMOVF、SYNCMISS、および SYNCERR ビットをクリアします。

0 : 同期またはトリミングエラーを信号化しません。

1 : 同期またはトリミングエラーを信号化します。

ビット 1 **SYNCWARNF** : SYNC 警告フラグ

このフラグは、測定した周波数誤差が $FELIM * 3$ 以上で $FELIM * 128$ 未満である場合に、ハードウェアによってセットされます。周波数誤差を補正するには、TRIM 値を複数のステップで調整する必要があります。CRS_CR レジスタの SYNCWARNIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCWARNF ビットをセットすることで、ソフトウェアによってクリアされます。

0 : SYNC 警告を信号化しません。

1 : SYNC 警告を信号化します。

ビット 0 **SYNCOKF** : SYNC イベント OK フラグ

このフラグは、測定した周波数誤差が $FELIM * 3$ 未満である場合に、ハードウェアによってセットされます。これは、TRIM 値の補正が必要であること、または周波数誤差の補正は 1 つのトリミングステップで十分であることを意味します。CRS_CR レジスタの SYNCOKIE ビットがセットされている場合、割り込みが生成されます。CRS_ICR レジスタの SYNCOKF ビットをセットすることで、ソフトウェアによってクリアされます。

0 : SYNC イベント OK フラグを信号化しません。

1 : SYNC イベント OK フラグを信号化します。

8.6.4 CRS 割り込みフラグクリアレジスタ (CRS_ICR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|--------|------|-----------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ESYNCC | ERRC | SYNCWARNC | SYNCOKC |
| | | | | | | | | | | | | rw | rw | rw | rw |

ビット 31:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **ESYNCC** : 期待 SYNC クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの ESYNCF フラグをクリアします。

ビット 2 **ERRC** : エラークリアフラグ

このビットに 1 を書き込むと、TRIMOVF、SYNCMISS、および SYNCERR ビットがクリアされ、結果として CRS_ISR レジスタの ERRF フラグもクリアされます。

ビット 1 **SYNCWARNC** : SYNC 警告クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCWARNF フラグをクリアします。

ビット 0 **SYNCOKC** : SYNC イベント OK クリアフラグ

このビットに 1 を書き込むと、CRS_ISR レジスタの SYNCOKF フラグをクリアします。

8.6.5 CRS レジスタマップ

表 36. CRS レジスタマップとリセット値

| オフ | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|------|----------|-------------|------|----------------|------|------|----------------|------|------------|------|------|------|------|------|------|--------------|-------|------|------|-----------|------|------|---------|----------|---------|--------|------------|------|------|--------|--------|------------|-----------|---------|
| 0x00 | CRS_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TRIM[5:0] | | | | | | SWSYNC | AUTOTRIMEN | CEN | Res. | ESYNCE | ERRIE | SYNCWARNIE | SYNCOKIE | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x04 | CRS_CFGR | SYNCPOL | Res. | SYNC SRC [1:0] | | Res. | SYNC DIV [2:0] | | FELIM[7:0] | | | | | | | RELOAD[15:0] | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | | 1 | 0 | | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | |
| 0x08 | CRS_ISR | FECAP[15:0] | | | | | | | | | | | | | | | FEDIR | Res. | Res. | Res. | Res. | Res. | TRIMOVF | SYNCMISS | SYNCERR | Res. | Res. | Res. | Res. | ESYNCF | ERRF | SYNCWARNF | SYNCOKF | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | | | | | 0 | 0 | 0 | |
| 0x0C | CRS_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ESYNCC | ERRC | SYNCWARNC | SYNCOKC |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

9 汎用 I/O (GPIO)

9.1 概要

各汎用 I/O ポートにはそれぞれ 4 つの 32 ビット設定レジスタ (GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR)、2 つの 32 ビットデータレジスタ (GPIOx_IDR、GPIOx_ODR)、および 1 つの 32 ビットセット/リセットレジスタ (GPIOx_BSRR) があります。さらに、すべての GPIO には、それぞれ 32 ビットロックレジスタ (GPIOx_LCKR) と 2 つの 32 ビットオルタネート機能選択レジスタ (GPIOx_AFRH、GPIOx_AFRL) があります。

9.2 GPIO の主な機能

- 出力状態：プッシュプルまたはオープンドレイン + プルアップ/プルダウン
- 出力データレジスタ (GPIOx_ODR) またはペリフェラル (オルタネート機能出力) からの出力データ
- 各 I/O のスピード選択
- 入力状態：フローティング、プルアップ/プルダウン、アナログ
- 入力データレジスタ (GPIOx_IDR) またはペリフェラル (オルタネート機能入力) への入力データ
- GPIOx_ODR へのビット単位の書き込みアクセス用のビットセット/リセットレジスタ (GPIOx_BSRR)
- I/O ポート設定を固定するロック機構 (GPIOx_LCKR)
- アナログ機能
- オルタネート機能選択レジスタ
- 2 クロックサイクルごとに変化可能な高速トグル
- 柔軟性の高いピンの多重化により、I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用可能

9.3 GPIO の機能説明

各 I/O ポートの特定のハードウェア特性については、データシートに記載されています。汎用 I/O (GPIO) ポートの各ポートビットは、ソフトウェアによって以下の動作モードを個別に設定できます。

- 入力フローティング
- 入力プルアップ
- 入力プルダウン
- アナログ
- プルアップまたはプルダウン機能を持つ出力オープンドレイン
- プルアップまたはプルダウン機能を持つ出力プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能プッシュプル
- プルアップまたはプルダウン機能を持つオルタネート機能オープンドレイン

各 I/O ポートビットは自由にプログラム可能ですが、I/O ポートレジスタには 32 ビットワード、ハーフワード、またはバイト単位でアクセスする必要があります。GPIOx_BSRR および GPIOx_BRR レジスタを使用すると、任意の GPIOx_ODR レジスタに不可分な読み出し/変更アクセスを行うことができます。これにより、読み出しと変更アクセスの間に IRQ が発生するリスクを回避できます。



図 22 および 図 23 は、標準 I/O ポートビットと 5 V トレラント I/O ポートビットの基本構造をそれぞれ示します。表 38 には、実行可能なポートビットの設定を示します。

図 22. I/O ポートビットの基本構造

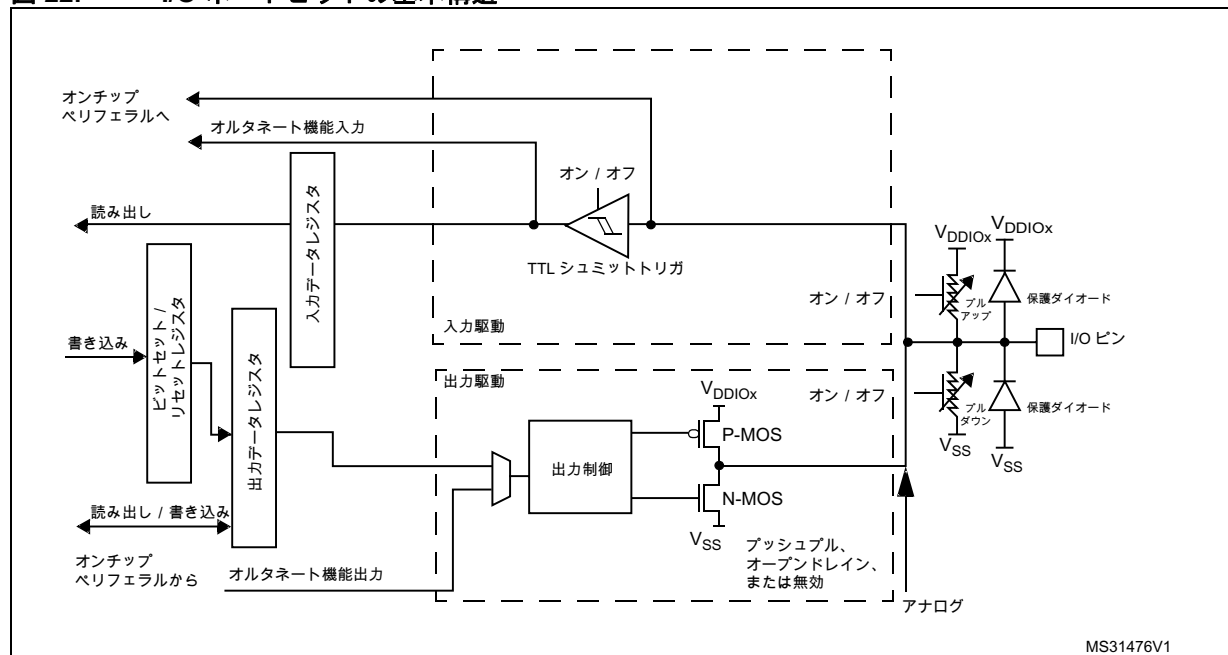
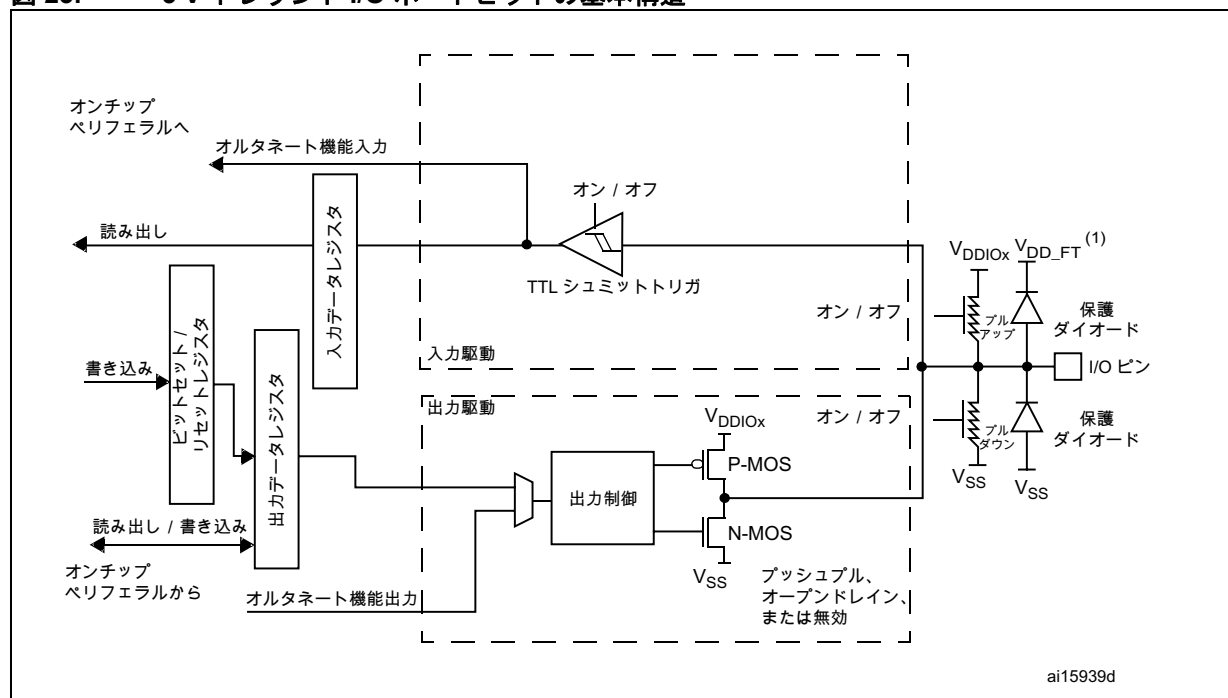


図 23. 5 V トレラント I/O ポートビットの基本構造



1. V_{DD_FT} は、5 V トレラント I/O に固有の電位であり、 V_{DD} とは異なります。

表 37. ポートビット設定表⁽¹⁾

| MODE(i) [1:0] | OTYPER(i) | OSPEED(i) [1:0] | | PUPD(i) [1:0] | | I/O 設定 | |
|------------------|-----------|--------------------|---|------------------|---|------------------|---------|
| 01 | 0 | SPEED [1:0] | | 0 | 0 | GP 出力 | PP |
| | 0 | | | 0 | 1 | GP 出力 | PP + PU |
| | 0 | | | 1 | 0 | GP 出力 | PP + PD |
| | 0 | | | 1 | 1 | 予約済み | |
| | 1 | | | 0 | 0 | GP 出力 | OD |
| | 1 | | | 0 | 1 | GP 出力 | OD + PU |
| | 1 | | | 1 | 0 | GP 出力 | OD + PD |
| | 1 | | | 1 | 1 | 予約済み (GP 出力 OD) | |
| 10 | 0 | SPEED [1:0] | | 0 | 0 | AF | PP |
| | 0 | | | 0 | 1 | AF | PP + PU |
| | 0 | | | 1 | 0 | AF | PP + PD |
| | 0 | | | 1 | 1 | 予約済み | |
| | 1 | | | 0 | 0 | AF | OD |
| | 1 | | | 0 | 1 | AF | OD + PU |
| | 1 | | | 1 | 0 | AF | OD + PD |
| | 1 | | | 1 | 1 | 予約済み | |
| 00 | x | x | x | 0 | 0 | 入力 | フローティング |
| | x | x | x | 0 | 1 | 入力 | PU |
| | x | x | x | 1 | 0 | 入力 | PD |
| | x | x | x | 1 | 1 | 予約済み (入力フローティング) | |
| 11 | x | x | x | 0 | 0 | 入力/出力 | アナログ |
| | x | x | x | 0 | 1 | 予約済み | |
| | x | x | x | 1 | 0 | | |
| | x | x | x | 1 | 1 | | |

1. GP = 汎用、PP = プッシュプル、PU = プルアップ、PD = プルダウン、OD = オープンドレイン、AF = オルタネート機能

9.3.1 汎用 I/O (GPIO)

リセット中とリセット直後は、オルタネート機能は有効ではなく、ほとんどの I/O ポートはアナログモードに設定されています。

リセット後、デバッグピンはオルタネート機能のプルアップ/プルダウンに設定されています。

- PA14 : SWCLK プルダウン
- PA13 : SWDIO プルアップ

ピンが出力として設定されている場合、出力データレジスタ (GPIOx_ODR) に書き込まれた値が I/O ピンに出力されます。出力ドライバをプッシュプルモードまたはオープンドレインモード (ローレベルのみが駆動され、ハイレベルはハイインピーダンス) で使用することができます。

入力データレジスタ (GPIOx_IDR) は、AHB クロックサイクルごとに、I/O ピン上のデータをキャプチャします。

すべての GPIO ピンに、内部ウィークプルアップ抵抗とウィークプルダウン抵抗があり、GPIOx_PUPDR レジスタの値によってこれらを有効化/無効化できます。

9.3.2 I/O ピンオルタネート機能マルチプレクサと配置

デバイスの I/O ピンは、マルチプレクサを介してオンボードのペリフェラル／モジュールに接続され、一度に 1 つのペリフェラルオルタネート機能 (AF) のみが 1 つの I/O ピンに接続可能となっています。この方法により、同じ I/O ピンを共有するペリフェラル間での競合を無くすることができます。

各 I/O ピンは、最大 16 のオルタネート機能入力 (AF0～AF15) を持つマルチプレクサを内蔵しており、これらの入力、GPIOx_AFRL (ピン 0～7) と GPIOx_AFRH (ピン 8～15) レジスタを介して設定することができます。

- リセット後、すべての I/O はオルタネート機能 0 (AF0) に接続されます。
- 各ピンに固有のオルタネート機能割り当てについての詳細は、デバイスデータシートに記載されています。

この柔軟性の高い I/O 多重化アーキテクチャに加え、各ペリフェラルではオルタネート機能がそれぞれの I/O ピンに配置されており、さらに小型のパッケージで利用できるペリフェラルの数を最適化します。

I/O を任意の設定で使用するには、次の手順に従います。

- **デバッグ機能**： 各デバイスのリセット後、これらのピンはデバッグホストによってすぐに使用可能なオルタネート機能ピンとして割り当てられます。
- **GPIO**： 必要とする I/O を、GPIOx_MODER レジスタで出力、入力、またはアナログとして設定します。
- **ペリフェラルオルタネート機能**
 - I/O を GPIOx_AFRL または GPIOx_AFRH レジスタのどちらかで必要とする AFx に接続します。
 - タイプ、プルアップ／プルダウン、出力スピードをそれぞれ GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタで選択します。
 - 必要とする I/O を、GPIOx_MODER レジスタでオルタネート機能として設定します。
- **追加機能**：
 - ADC、DAC、COMP については、GPIOx_MODER レジスタのアナログモードで必要とする I/O を設定し、ADC、DAC、COMP レジスタで必要な機能を設定します。
 - RTC、WKUPx、オシレータなどの追加機能については、関連する RTC、PWR、および RCC レジスタで必要な機能を設定します。これらの機能は、標準の GPIO レジスタの設定よりも優先されます。

オルタネート機能 I/O ピンの配置に関する詳細は、デバイスデータシートの「オルタネート機能配置」表を参照してください。

9.3.3 I/O ポート制御レジスタ

各 GPIO ポートには 32 ビットメモリマップド制御レジスタが 4 つ (GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR) あり、最大 16 個の I/O を設定します。GPIOx_MODER レジスタは I/O モード (入力、出力、AF、アナログ) を選択するために使用されます。GPIOx_OTYPER および GPIOx_OSPEEDR レジスタは、出力タイプ (プッシュプルまたはオープンドレイン) およびスピードを選択するために使用されます。I/O の方向がどちらであっても、GPIOx_PUPDR レジスタは、プルアップ／プルダウンを選択するために使用されます。

9.3.4 I/O ポートデータレジスタ

各 GPIO には、16 ビットメモリマップドデータレジスタが 2 つあります。入力データレジスタ (GPIOx_IDR) と出力データレジスタ (GPIOx_ODR) です。GPIOx_ODR は出力されるデータを格納し、読み出し/書き込みアクセスが可能です。I/O から入力されるデータは読み出し専用の入力データレジスタ (GPIOx_IDR) に格納されます。

レジスタの説明は、[セクション 9.4.5 : GPIO ポート入力データレジスタ \(GPIOx_IDR\) \(x = A~D、H\)](#) および [セクション 9.4.6 : GPIO ポート出力データレジスタ \(GPIOx_ODR\) \(x = A~D、H\)](#) を参照してください。

9.3.5 I/O データのビット単位の操作

ビットセット/リセットレジスタ (GPIOx_BSRR) は 32 ビットレジスタで、アプリケーションによる出力データレジスタ (GPIOx_ODR) のビット単位のセット/リセットを可能にします。ビットセット/リセットレジスタは GPIOx_ODR の 2 倍のサイズです。

GPIOx_ODR の各ビットには GPIOx_BSRR の 2 つの制御ビット BS(i) と BR(i) が対応します。ビット BS(i) に 1 を書き込むと、対応する ODR(i) ビットが**セット**されます。ビット BR(i) に 1 を書き込むと、対応する ODR(i) ビットが**リセット**されます。

GPIOx_BSRR のいかなるビットに 0 を書き込んでも GPIOx_ODR の対応するビットには影響しません。仮に、GPIOx_BSRR のビットに対してセットおよびリセットの両方を実行しようとした場合、セット動作が優先されます。

GPIOx_BSRR レジスタを使用した GPIOx_ODR 内の個々のビットの変更には、1 回限りの効果しかなく、GPIOx_ODR ビットを固定するものではありません。GPIOx_ODR のビットは常に直接アクセスすることができます。GPIOx_BSRR レジスタによって、ビット単位の不可分操作を行うことができます。

GPIOx_ODR をビットレベルでプログラムする場合は、ソフトウェアで割り込みを無効にする必要はありません。1 回の不可分な AHB 書き込みアクセスで 1 ビットまたは複数ビットを変更することができます。

9.3.6 GPIO ロック機構

GPIOx_LCKR レジスタへ特定の書き込みシーケンスを行うことにより、GPIO 制御レジスタをロックすることができます。ロックされるレジスタは、GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH です。

GPIOx_LCKR レジスタに書き込むには、特定の書き込み/読み出しシーケンスを行う必要があります。このレジスタのビット 16 に適切な LOCK シーケンスを行う場合、LCKR[15:0] の値を使用して I/O の設定を固定します (この書き込みシーケンス中、LCKR[15:0] の値は同じである必要があります)。あるポートビットに LOCK シーケンスが行われると、次のリセットまで、そのポートビットの値を変更できなくなります。GPIOx_LCKR の各ビットによって、制御レジスタ (GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH) の対応するビットが停止されます。

GPIOx_LCKR ビット 16 を [15:0] ビットと同時に設定する必要があるため、この LOCK シーケンス ([セクション 9.4.8 : GPIO ポート設定ロックレジスタ \(GPIOx_LCKR\) \(x = A~D、H\)](#) を参照) は、GPIOx_LCKR レジスタへのワード (32 ビット長) アクセスを使用してのみ実行できます。

詳細については、[セクション 9.4.8 : GPIO ポート設定ロックレジスタ \(GPIOx_LCKR\) \(x = A~D、H\)](#) の LCKR レジスタの説明を参照してください。

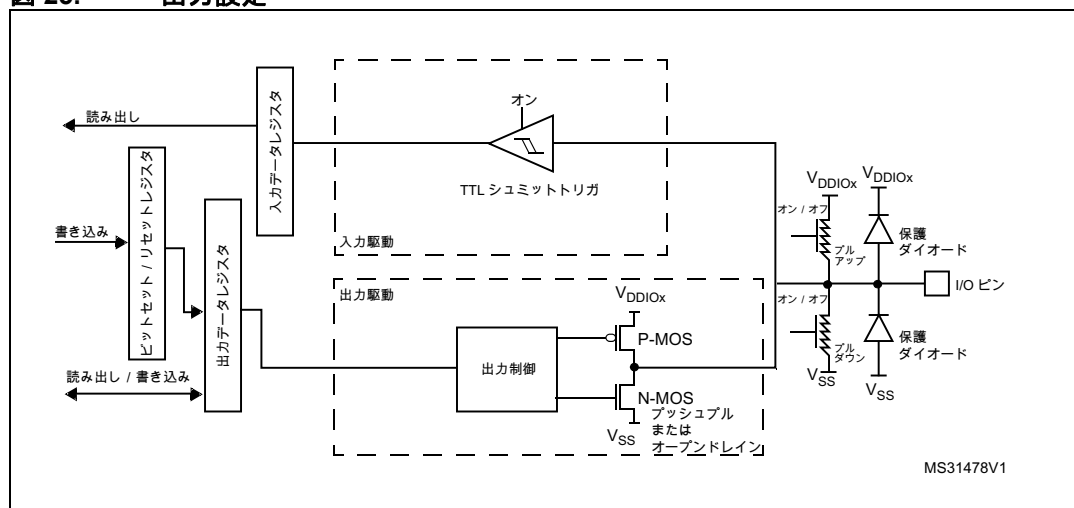
9.3.10 出力設定

I/O ポートが出力としてプログラムされた場合、

- 出力バッファが有効になります。
 - オープンドレインモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときはポートはハイインピーダンス状態のままです (P-MOS が有効になることはない)。
 - プッシュプルモード：出力レジスタが“0”のときは N-MOS が有効になり、“1”のときは P-MOS が有効になります。
- シュミットトリガ入力が有効になります。
- GPIOx_PUPDR レジスタの値に応じて、プルアップおよびプルダウン抵抗が有効になります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。
- 出力データレジスタの読み出しアクセスによって、最後に書き込まれたデータが得られます。

図 25 は、I/O ポートビットの出力設定を示しています。

図 25. 出力設定



9.3.11 オルタネート機能設定

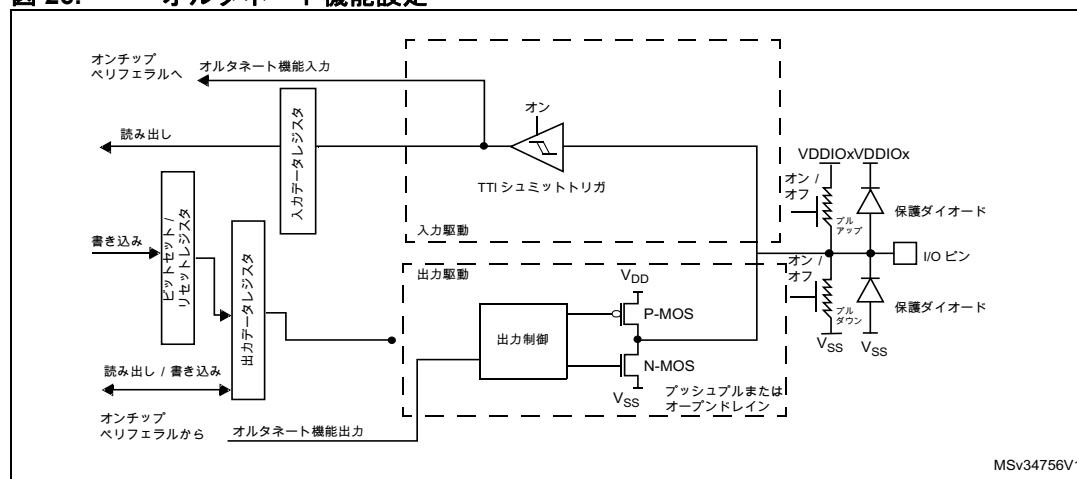
I/O ポートがオルタネート機能としてプログラムされた場合、

- 出力バッファをオープンドレインまたはプッシュプルモードに設定することができます。
- 出力バッファが、ペリフェラル (トランスミッタイネーブルおよびデータ) からの信号で駆動されます。
- シュミットトリガ入力があります。
- ウィークプルアップ抵抗およびプルダウン抵抗が有効になるか否かは、GPIOx_PUPDR レジスタの値によって決まります。
- I/O ピン上のデータは、AHB クロックサイクルごとに入力データレジスタにサンプリングされます。
- 入力データレジスタへの読み出しアクセスによって、I/O 状態が得られます。

注： 上述したオルタネート機能の設定は、選択されたオルタネート機能が LCD 機能である場合は行われません。この場合、I/O は、オルタネート機能出力としてプログラムされるので、アナログ設定で説明されているように設定されます。

図 26 は、I/O ポートビットのオルタネート機能設定を示しています。

図 26. オルタネート機能設定



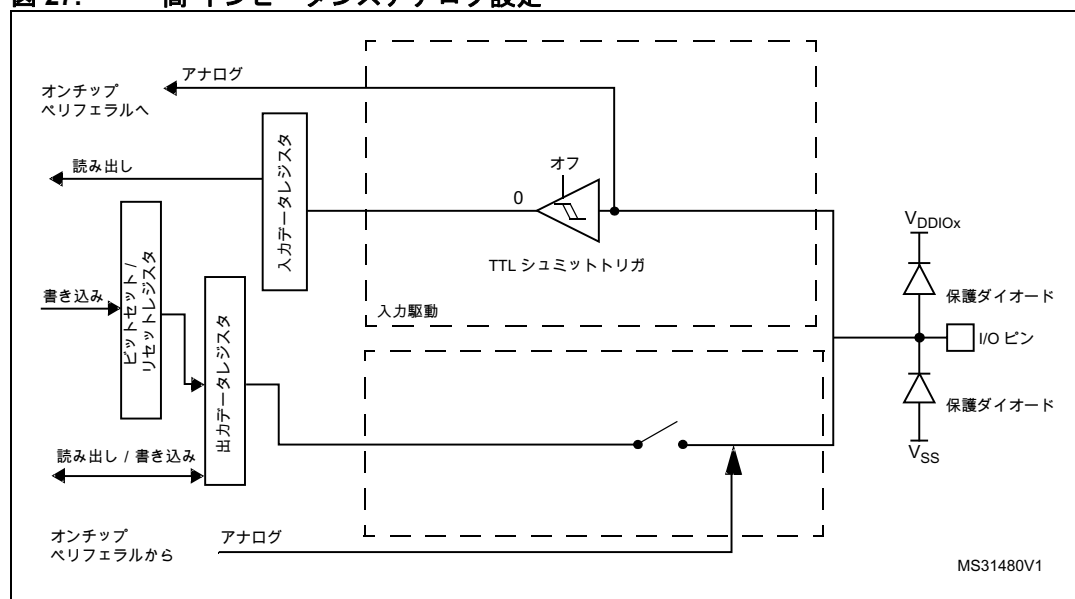
9.3.12 アナログ設定

I/O ポートがアナログとしてプログラムされた場合、

- 出力バッファが無効になります。
- シュミットトリガ入力は無効になり、I/O ピンのどのようなアナログ値に対しても消費電力をゼロに抑えます。シュミットトリガ回路の出力は、常に“0”になります。
- ウィークプルアップ抵抗およびプルダウン抵抗はハードウェアによって無効にされます。
- 入力データレジスタの読み出しアクセスを行うと、値“0”が得られます。

図 27 は、I/O ポートビットの高インピーダンスアナログ入力設定を示しています。

図 27. 高インピーダンスアナログ設定



9.3.13 HSEまたはLSE オシレータのピンを GPIO として使用

HSE または LSE オシレータがスイッチオフされた場合（リセット後のデフォルト状態）、関連のオシレータピンを 通常の GPIO として使用することができます。

HSE または LSE オシレータがスイッチオンされた場合（RCC_CSR レジスタの HSEON または LSEON ビットを設定することで）、オシレータは関連ピンを制御しますが、これらのピンの GPIO 設定は無効です。

オシレータがユーザ外部クロックモードに設定されている場合、OSC_IN または OSC32_IN ピンのみがクロック入力のために確保されますが、OSC_OUT または OSC32_OUT ピンは通常の GPIO として使用することができます。

9.3.14 GPIO ピンを RTC 供給ドメインで使用

コア供給ドメインの電源がオフになったとき（デバイスが STANDBY モードに移行したとき）、PC13/PC14/PC15 の GPIO が機能しなくなります。この場合、それらの GPIO 設定が RTC 設定によってバイパスされなければ、これらのピンはアナログ入力モードに設定されます。

RTC による I/O 制御の詳細は、[セクション 27.3:RTC の機能説明 \(575 ページ\)](#)を参照してください。

9.4 GPIO レジスタ

このセクションには、GPIO レジスタの詳細な説明が記載されています。

レジスタビット、レジスタアドレスオフセット、リセット値の概要は、[表 38](#)を参照してください。

ペリフェラルレジスタはワード、ハーフワード、バイトのいずれかのモードで書き込むことができます。

9.4.1 GPIO ポートモードレジスタ (GPIOx_MODER) (x = A~D、H)

アドレスオフセット : 0x00

リセット値 :

- ポート A 0xEBFF FCFF
- 他のポート 0xFFFF FFFF

| | | | | | | | | | | | | | | | |
|-------------|-----|-------------|-----|-------------|-----|-------------|-----|-------------|-----|-------------|-----|------------|-----|------------|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| MODE15[1:0] | | MODE14[1:0] | | MODE13[1:0] | | MODE12[1:0] | | MODE11[1:0] | | MODE10[1:0] | | MODE9[1:0] | | MODE8[1:0] | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MODE7[1:0] | | MODE6[1:0] | | MODE5[1:0] | | MODE4[1:0] | | MODE3[1:0] | | MODE2[1:0] | | MODE1[1:0] | | MODE0[1:0] | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 2y+1:2y **MODEy[1:0]** : ポート x 設定ビット (y = 0~15)

これらのビットは、I/O モードを設定するために、ソフトウェアによって書き込まれます。

00 : 入力モード

01 : 汎用出力モード

10 : オルタネート機能モード

11 : アナログモード (リセット状態)

9.4.2 GPIO ポート出力タイプレジスタ (GPIOx_OTYPER) (x = A~D、H)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OT15 | OT14 | OT13 | OT12 | OT11 | OT10 | OT9 | OT8 | OT7 | OT6 | OT5 | OT4 | OT3 | OT2 | OT1 | OT0 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 OTy : ポート x 設定ビット (y = 0~15)

これらのビットは、I/O 出力タイプを設定するために、ソフトウェアによって書き込まれます。

- 0 : 出力プッシュプル (リセット状態)
- 1 : 出力オープンドレイン

9.4.3 GPIO ポート出力スピードレジスタ (GPIOx_OSPEEDR) (x = A~D、H)

アドレスオフセット : 0x08

リセット値 :

- ポート A 0x0C00 0000
- 他のポート 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------------|----|-------------------|----|-------------------|----|-------------------|----|-------------------|----|-------------------|----|------------------|----|------------------|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| OSPEED15 [1:0] | | OSPEED14 [1:0] | | OSPEED13 [1:0] | | OSPEED12 [1:0] | | OSPEED11 [1:0] | | OSPEED10 [1:0] | | OSPEED9 [1:0] | | OSPEED8 [1:0] | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OSPEED7 [1:0] | | OSPEED6 [1:0] | | OSPEED5 [1:0] | | OSPEED4 [1:0] | | OSPEED3 [1:0] | | OSPEED2 [1:0] | | OSPEED1 [1:0] | | OSPEED0 [1:0] | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 2y+1:2y OSPEEDy[1:0] : ポート x 設定ビット (y = 0~15)

これらのビットは、I/O の出力スピードを設定するために、ソフトウェアによって書き込まれます。

- 00 : 超ロースピード
- 01 : ロースピード
- 10 : ミディアムスピード
- 11 : ハイスピード

注 : 周波数仕様、およびスピード別の電源や負荷条件については、デバイスデータシートを参照してください。

9.4.4 GPIO ポートプルアップ／プルダウンレジスタ (GPIOx_PUPDR) (x = A~D、H)

アドレスオフセット : 0x0C

リセット値 :

- ポート A 0x2400 0000
- 他のポート 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|-----|-------------|-----|-------------|-----|-------------|-----|-------------|-----|-------------|-----|------------|-----|------------|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| PUPD15[1:0] | | PUPD14[1:0] | | PUPD13[1:0] | | PUPD12[1:0] | | PUPD11[1:0] | | PUPD10[1:0] | | PUPD9[1:0] | | PUPD8[1:0] | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PUPD7[1:0] | | PUPD6[1:0] | | PUPD5[1:0] | | PUPD4[1:0] | | PUPD3[1:0] | | PUPD2[1:0] | | PUPD1[1:0] | | PUPD0[1:0] | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 2y+1:2y PUPDy[1:0] : ポート x 設定ビット (y = 0~15)

これらのビットは、I/O のプルアップまたはプルダウンを設定するために、ソフトウェアによって書き込まれます。

- 00 : プルアップ／プルダウンなし
- 01 : プルアップ
- 10 : プルダウン
- 11 : 予約済み

9.4.5 GPIO ポート入力データレジスタ (GPIOx_IDR) (x = A~D、H)

アドレスオフセット : 0x10

リセット値 : 0x0000 XXXX (X は未定義)

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ID15 | ID14 | ID13 | ID12 | ID11 | ID10 | ID9 | ID8 | ID7 | ID6 | ID5 | ID4 | ID3 | ID2 | ID1 | ID0 |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 IDy : ポート入力データビット (y = 0~15)

これらのビットは読み出し専用です。これらのビットには、対応する I/O ポートの入力値が格納されています。

9.4.6 GPIO ポート出力データレジスタ (GPIOx_ODR) (x = A~D、H)

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OD15 | OD14 | OD13 | OD12 | OD11 | OD10 | OD9 | OD8 | OD7 | OD6 | OD5 | OD4 | OD3 | OD2 | OD1 | OD0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **ODy** : ポート出入力データビット (y = 0~15)

これらのビットは、ソフトウェアによって読み出し／書き込みができます。

注 : 不可分なビット単位のセット／リセットのために、GPIOx_BSRR または GPIOx_BRR レジスタ (x = A~F) への書き込みによって、OD ビットを個々にセット／リセットすることができます。

9.4.7 GPIO ポートビットセット／リセットレジスタ (GPIOx_BSRR) (x = A~D、H)

アドレスオフセット : 0x18

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| BR15 | BR14 | BR13 | BR12 | BR11 | BR10 | BR9 | BR8 | BR7 | BR6 | BR5 | BR4 | BR3 | BR2 | BR1 | BR0 |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BS15 | BS14 | BS13 | BS12 | BS11 | BS10 | BS9 | BS8 | BS7 | BS6 | BS5 | BS4 | BS3 | BS2 | BS1 | BS0 |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

ビット 31:16 **BRy** : ポート x リセットビット y (y = 0~15)

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0 : 対応する ODx ビットの値は変化しません。

1 : 対応する ODx ビットをリセットします。

注 : BSx ビットと BRx ビットの両方がセットされた場合、BSx ビットが優先されます。

ビット 15:0 **BSy** : ポート x セットビット y (y = 0~15)

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0 : 対応する ODx ビットの値は変化しません。

1 : 対応する ODx ビットをセットします。

9.4.8 GPIO ポート設定ロックレジスタ (GPIOx_LCKR) (x = A~D、H)

このレジスタは、ビット 16 (LCKK) に正しい書き込みシーケンスが行われたときに、ポートビットの設定をロックするために使用されます。ビット [15:0] の値は、GPIO の設定をロックするために使用されます。書き込みシーケンスの間は、LCKR[15:0] の値を変更することはできません。ポートビットに LOCK シーケンスが適用されると、次のリセットまで、このポートビットの値を変更できなくなります。

注： GPIOx_LCKR レジスタへの書き込みには特定の書き込みシーケンスが使われます。このロックシーケンスではワードアクセス (32 ビット長) のみ可能です。

各ロックビットによって、特定の設定レジスタ (制御レジスタおよびオルタネート機能レジスタ) が固定されます。

アドレスオフセット : 0x1C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | LCKK |
| | | | | | | | | | | | | | | | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| LCK15 | LCK14 | LCK13 | LCK12 | LCK11 | LCK10 | LCK9 | LCK8 | LCK7 | LCK6 | LCK5 | LCK4 | LCK3 | LCK2 | LCK1 | LCK0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **LCKK** : ロックキー

このビットは常時読み出しができます。ロックキー書き込みシーケンスを使用しなければ変更できません。

- 0 : ポート設定ロックキーは無効です。
 - 1 : ポート設定ロックキーは有効です。GPIOx_LCKR レジスタは、MCU リセットが発生するまでロックされます。
- ロックキー書き込みシーケンス :
- 書き込み LCKR[16] = "1" + LCKR[15:0]
 - 書き込み LCKR[16] = "0" + LCKR[15:0]
 - 書き込み LCKR[16] = "1" + LCKR[15:0]
- 読み出し LCKR
- 読み出し LCKR[16] = "1" (この読み出し操作は任意だが、ロックが有効であることを確認できる。)

注： ロックキー書き込みシーケンスの間は、LCK[15:0] の値を変更することはできません。
ロックシーケンス中にエラーが発生すると、ロックは中止されます。
ポートの任意のビットの最初のロックシーケンスの後、次の CPU リセットまでは、LCKK ビットのいかなる読み出しアクセスに対しても、“1”が返されます。

ビット 15:0 **LCKy** : ポート x ロックビット y (y = 0~15)

これらのビットは読み出し／書き込みができますが、書き込めるのは LCKK ビットが“0”のときだけです。

- 0 : ポート設定はロックされません。
- 1 : ポート設定はロックされます。



9.4.9 GPIO オルタネート機能下位レジスタ (GPIOx_AFRL) (x = A~D、H)

アドレスオフセット : 0x20

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|-----|-----|-----|-------------|-----|-----|-----|-------------|-----|-----|-----|-------------|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| AFSEL7[3:0] | | | | AFSEL6[3:0] | | | | AFSEL5[3:0] | | | | AFSEL4[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| AFSEL3[3:0] | | | | AFSEL2[3:0] | | | | AFSEL1[3:0] | | | | AFSEL0[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:0 **AFSELy[3:0]** : ポート x ピン y (y = 0~7) のオルタネート機能選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択 :

| | |
|-----------------------------|-------------|
| 0000 : AF0 | 1000 : 予約済み |
| 0001 : AF1 (ポート A、B、C、D のみ) | 1001 : 予約済み |
| 0010 : AF2 (ポート A、B、C のみ) | 1010 : 予約済み |
| 0011 : AF3 (ポート A、B、C のみ) | 1011 : 予約済み |
| 0100 : AF4 (ポート A、B のみ) | 1100 : 予約済み |
| 0101 : AF5 (ポート A、B のみ) | 1101 : 予約済み |
| 0110 : AF6 (ポート A、B のみ) | 1110 : 予約済み |
| 0111 : AF7 (ポート A のみ) | 1111 : 予約済み |

9.4.10 GPIO オルタネート機能上位レジスタ (GPIOx_AFRH) (x = A~D、H)

アドレスオフセット : 0x24

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------------|-----|-----|-----|--------------|-----|-----|-----|--------------|-----|-----|-----|--------------|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| AFSEL15[3:0] | | | | AFSEL14[3:0] | | | | AFSEL13[3:0] | | | | AFSEL12[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| AFSEL11[3:0] | | | | AFSEL10[3:0] | | | | AFSEL9[3:0] | | | | AFSEL8[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:0 **AFSELy[3:0]** : ポート x ピン y (y = 8~15) のオルタネート機能選択

これらのビットは、オルタネート機能 I/O を設定するために、ソフトウェアによって書き込まれます。

AFSELy 選択 :

| | |
|-----------------------------|-------------|
| 0000 : AF0 | 1000 : 予約済み |
| 0001 : AF1 (ポート A、B、C、D のみ) | 1001 : 予約済み |
| 0010 : AF2 (ポート A、B、C のみ) | 1010 : 予約済み |
| 0011 : AF3 (ポート A、B、C のみ) | 1011 : 予約済み |
| 0100 : AF4 (ポート A、B のみ) | 1100 : 予約済み |
| 0101 : AF5 (ポート A、B のみ) | 1101 : 予約済み |
| 0110 : AF6 (ポート A、B のみ) | 1110 : 予約済み |
| 0111 : AF7 (ポート A のみ) | 1111 : 予約済み |



9.4.11 GPIO ポートビットリセットレジスタ (GPIOx_BRR) (x = A~D、H)

アドレスオフセット : 0x28
リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BR15 | BR14 | BR13 | BR12 | BR11 | BR10 | BR9 | BR8 | BR7 | BR6 | BR5 | BR4 | BR3 | BR2 | BR1 | BR0 |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

ビット 31:16 予約済み

ビット 15:0 **BRy** : ポート x リセットビット y (y= 0~15)

これらのビットは書き込み専用です。これらのビットを読み出すと、値 0x0000 が返されます。

0 : 対応する ODx ビットの値は変化しません。

1 : 対応する ODx ビットをリセットします。

9.4.12 GPIO レジスタマップ

次の表に、GPIO レジスタマップとリセット値を示します。

表 38. GPIO レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|------------------------------|---------------|------|------|---------------|--------------|---------------|------|---------------|--------------|---------------|------|---------------|--------------|--------------|------|--------------|--------------|--------------|-------|--------------|--------------|--------------|------|--------------|-------------|--------------|------|--------------|-------------|--------------|------|--------------|
| 0x00 | GPIOA_MODER | MODE15[1:0] | | | MODE14[1:0] | | MODE13[1:0] | | MODE12[1:0] | | MODE11[1:0] | | MODE10[1:0] | | MODE9[1:0] | | MODE8[1:0] | | MODE7[1:0] | | MODE6[1:0] | | MODE5[1:0] | | MODE4[1:0] | | MODE3[1:0] | | MODE2[1:0] | | MODE1[1:0] | | MODE0[1:0] |
| | リセット値 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0x00 | GPIOx_MODER (x = A、B、C、H) | MODE15[1:0] | | | MODE14[1:0] | | MODE13[1:0] | | MODE12[1:0] | | MODE11[1:0] | | MODE10[1:0] | | MODE9[1:0] | | MODE8[1:0] | | MODE7[1:0] | | MODE6[1:0] | | MODE5[1:0] | | MODE4[1:0] | | MODE3[1:0] | | MODE2[1:0] | | MODE1[1:0] | | MODE0[1:0] |
| | リセット値 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0x04 | GPIOx_OTYPER (x = A～D、H) | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OT15 | OT14 | OT13 | OT12 | OT11 | OT10 | OT9 | OT8 | OT7 | OT6 | OT5 | OT4 | OT3 | OT2 | OT1 | OT0 |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | GPIOA_OSPEEDR | OSPEED15[1:0] | | | OSPEED14[1:0] | | OSPEED13[1:0] | | OSPEED12[1:0] | | OSPEED11[1:0] | | OSPEED10[1:0] | | OSPEED9[1:0] | | OSPEED8[1:0] | | OSPEED7[1:0] | | OSPEED6[1:0] | | OSPEED5[1:0] | | OSPEED4[1:0] | | OSPEED3[1:0] | | OSPEED2[1:0] | | OSPEED1[1:0] | | OSPEED0[1:0] |
| | リセット値 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | GPIOx_OSPEEDR (x = B、C、H) | OSPEED15[1:0] | | | OSPEED14[1:0] | | OSPEED13[1:0] | | OSPEED12[1:0] | | OSPEED11[1:0] | | OSPEED10[1:0] | | OSPEED9[1:0] | | OSPEED8[1:0] | | OSPEED7[1:0] | | OSPEED6[1:0] | | OSPEED5[1:0] | | OSPEED4[1:0] | | OSPEED3[1:0] | | OSPEED2[1:0] | | OSPEED1[1:0] | | OSPEED0[1:0] |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | GPIOA_PUPDR | PUPD15[1:0] | | | PUPD14[1:0] | | PUPD13[1:0] | | PUPD12[1:0] | | PUPD11[1:0] | | PUPD10[1:0] | | PUPD9[1:0] | | PUPD8[1:0] | | PUPD7[1:0] | | PUPD6[1:0] | | PUPD5[1:0] | | PUPD4[1:0] | | PUPD3[1:0] | | PUPD2[1:0] | | PUPD1[1:0] | | PUPD0[1:0] |
| | リセット値 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | GPIOx_PUPDR (x = B、C、H) | PUPD15[1:0] | | | PUPD14[1:0] | | PUPD13[1:0] | | PUPD12[1:0] | | PUPD11[1:0] | | PUPD10[1:0] | | PUPD9[1:0] | | PUPD8[1:0] | | PUPD7[1:0] | | PUPD6[1:0] | | PUPD5[1:0] | | PUPD4[1:0] | | PUPD3[1:0] | | PUPD2[1:0] | | PUPD1[1:0] | | PUPD0[1:0] |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x10 | GPIOx_IDR (x = A～D、H) | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ID15 | ID14 | ID13 | ID12 | ID11 | ID10 | ID9 | ID8 | ID7 | ID6 | ID5 | ID4 | ID3 | ID2 | ID1 | ID0 |
| | リセット値 | | | | | | | | | | | | | | | | | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 0x14 | GPIOx_ODR (x = A～D、H) | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OD15 | OD14 | OD13 | OD12 | OD11 | OD10 | OD9 | OD8 | OD7 | OD6 | OD5 | OD4 | OD3 | OD2 | OD1 | OD0 |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x18 | GPIOx_BSRR (x = A～D、H) | BR15 | BR14 | BR13 | BR12 | BR11 | BR10 | BR9 | BR8 | BR7 | BR6 | BR5 | BR4 | BR3 | BR2 | BR1 | BR0 | BS15 | BS14 | BS13 | BS12 | BS11 | BS10 | BS9 | BS8 | BS7 | BS6 | BS5 | BS4 | BS3 | BS2 | BS1 | BS0 |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x1C | GPIOx_LCKR (x = A～D、H) | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | LCKK | LCK15 | LCK14 | LCK13 | LCK12 | LCK11 | LCK10 | LCK9 | LCK8 | LCK7 | LCK6 | LCK5 | LCK4 | LCK3 | LCK2 | LCK1 | LCK0 |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x20 | GPIOx_AFR1 (x = A～D、H) | AFSEL7[3:0] | | | | AFSEL6[3:0] | | | | AFSEL5[3:0] | | | | AFSEL4[3:0] | | | | AFSEL3[3:0] | | | | AFSEL2[3:0] | | | | AFSEL1[3:0] | | | | AFSEL0[3:0] | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x24 | GPIOx_AFRH (x = A～D、H) | AFSEL15[3:0] | | | | AFSEL14[3:0] | | | | AFSEL13[3:0] | | | | AFSEL12[3:0] | | | | AFSEL11[3:0] | | | | AFSEL10[3:0] | | | | AFSEL9[3:0] | | | | AFSEL8[3:0] | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

表 38. GPIO レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|--------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 0x28 | GPIOx_BRR (x = A~D、H) | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BR15 | BR14 | BR13 | BR12 | BR11 | BR10 | BR9 | BR8 | BR7 | BR6 | BR5 | BR4 | BR3 | BR2 | BR1 | BR0 |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

10 システム設定コントローラ (SYSCFG)

10.1 概要

デバイスは一連の設定レジスタを持っています。システム設定コントローラの主な目的は次の通りです。

- メモリを再配置する。
- タイマ入力キャプチャチャネルヘトリガソースを再配置する。
- 内部エッジ検出回路への外部割り込みラインの多重化を管理する。
- 入力キャプチャの多重化、オシレータピンの再配置などの専用機能を有効にする。
- I2C Fm+ モードを管理する。
- LCD 電源レールをデカップリングする。
- ファイアウォールを管理する。
- 温度センサおよび内部電圧基準を管理する (コンパレータ、48 MHz の HSI および ADC 用途を含む)。

Cortex[®]-M0+ は、入力信号の イベント上で遷移が発生した場合、WFE (Wait For Event) からウェイクアップすることができます。マルチプロセッサ環境においてセマフォ管理をサポートするために、コアが、SEV 命令の実行中に EVENTOUT 信号出力に関するイベントを出力することも可能です。

STM32L0x3 デバイスでは、外部割り込みラインまたは RTC アラーム割り込みによって、イベント入力を生成することができます。また、Cortex[®]-M0+ の EVENTOUT 信号に接続する出力ピンを選択することもできます。EVENTOUT の多重化は GPIO オルタネート機能によって管理されます ([セクション 9.4.9 : GPIO オルタネート機能下位レジスタ \(GPIOx_AFRL\) \(x = A ~ D、H\)](#) および [セクション 9.4.10 : GPIO オルタネート機能上位レジスタ \(GPIOx_AFRH\) \(x = A ~ D、H\)](#) を参照)。

注： *EVENTOUT は、すべての GPIO に配置されるわけではありません (PC13、PC14、PC15 など)。*

10.2 SYSCFG レジスタ

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスする必要があります。

10.2.1 SYSCFG メモリ再配置レジスタ (SYSCFG_CFGR1)

このレジスタは、メモリの再配置に関する特定の設定に使用されます。

注： このレジスタは、RCC_APB2RSTR レジスタの SYSCFGRST ビットからはリセットされません。

アドレスオフセット：0x00

リセット値：0x000x 000x (X は BOOT ピンで選択されたメモリモード)

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **BOOT_MODE** : BOOT ピンのステータスビットによって選択されたブートモード

これらのビットは読み出し専用です。これらのビットは、ブートモードが BOOT ピンによって選択されたことを示します。ビット 9 は、FLASH_OTPR レジスタの BOOT1 ビットの補数に対応します。その値はオプションバイトで指定されています (セクション 2.4 : ブート設定 (46 ページ) を参照)。ビット 8 は、BOOT0 ピンでサンプリングされた値に対応します。

- 00 : メインフラッシュメモリブートモード
- 01 : システムフラッシュメモリブートモード
- 10 : 予約済み
- 11 : 内蔵 SRAM はブートモード

ビット 7:2 予約済みであり、リセット値に保持する必要があります。

ビット 1:0 **MEM_MODE** : メモリ配置選択ビット

これらのビットは、ソフトウェアによってセット/クリアされます。このビットによって、アドレス 0x0000 0000 のメモリの内部配置を制御します。リセット後、これらのビットは、BOOT ピンで選択されたメモリ配置を行います。

- 00 : メインフラッシュメモリは 0x0000 0000 に配置されます。
- 01 : システムフラッシュメモリは 0x0000 0000 に配置されます。
- 10 : 予約済み
- 11 : SRAM は 0x0000 0000 に配置されます。



10.2.2 SYSCFG ペリフェラルモード設定レジスタ (SYSCFG_CFGR2)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|----------|----------|-------------|-------------|-------------|-------------|------|------|------|------|-----------|------|-------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | I2C2_FMP | I2C1_FMP | I2C_PB9_FMP | I2C_PB8_FMP | I2C_PB7_FMP | I2C_PB6_FMP | Res. | Res. | Res. | Res. | CAPA[2:0] | | FWDIS | |
| | | r/w | r/w | | | | | | | | | r/w | r/w | r/w | r/w |

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **I2C2 FMP** : I2C2 Fm+ 駆動機能イネーブルビット

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、IOPORT 制御レジスタの AF 選択ビットを介して選択された I2C2 のピン、PB13 と PB14 で Fm+ モードが有効になります。

ビット 12 **I2C1 FMP** : I2C1 Fm+ 駆動機能イネーブルビット

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、IOPORT 制御レジスタの AF 選択ビットを介して選択された I2C1 のピンで Fm+ モードが有効になります。このビットは、I2C_PBX_FMP ビットで論理和がとられます。

ビット 11 **I2C PB9 FMP** : PB9 イネーブルビットの Fm+ 駆動機能

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、PB9 で Fm+ 駆動機能が強制的に有効になります。

ビット 10 **I2C PB8 FMP** : PB8 イネーブルビットの Fm+ 駆動機能

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、PB8 で Fm+ 駆動機能が強制的に有効になります。

ビット 9 **I2C PB7 FMP** : PB7 イネーブルビットの Fm+ 駆動機能

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、PB7 で Fm+ 駆動機能が強制的に有効になります。

ビット 8 **I2C PB6 FMP** : PB6 イネーブルビットの Fm+ 駆動機能

このビットはソフトウェアによってセット／クリアされます。このビットがセットされると、PB6 で Fm+ 駆動機能が強制的に有効になります。

ビット 7:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:1 **CAPA[2:0]** : オプションの外付けコンデンサへの内部 V_{LCD} レールの接続設定

これらのビットは、ソフトウェアによってセット/クリアされます。これらのビットは、オプションの外付けコンデンサへの内部 V_{LCD} レール供給電圧 (LCD_VLCD1、LCD_VLCD2、LCD_VLCD3) の接続設定を制御します。ビット 1、2、および3 はそれぞれ PB2、PB12、PB0 を制御します。

0 : I/O ポートは V_{LCD} 供給電圧に接続されません。

1 : I/O ポートは V_{LCD} 供給電圧に接続されます。

詳細については、[セクション : CAPA\[2:0\] 管理の説明](#)を参照してください。

ビット 0 **FWDIS** : ファイアウォールディセーブルビット

このビットは、デフォルト設定されます (リセット後)。このビットは、メモリセグメントへのアクセスを保護するために、ファイアウォール設定に従ってソフトウェアによってクリアされます。いったんクリアされると、ソフトウェアによって設定することはできません。システムリセットによってのみ、ビットがセットされます。

0 : ファイアウォールアクセス有効

1 : ファイアウォールアクセス無効

注 : このビットは、APB リセットではセットできません。このビットをセットするにはシステムリセットが必要です。

CAPA[2:0] 管理の説明

この機能を使用することで、内部 V_{LCD} レールの供給電圧を、コンデンサをはんだ付けできるような外部ピンに接続することが可能になります (表 39)。

この機能を実行すると、3個のピンを使用して外付けコンデンサを $1/2 V_{LCD}$ ($1/3$ バイアスの場合は $2/3 V_{LCD}$)、 $1/4 V_{LCD}$ ($1/3$ バイアスの場合は $1/3 V_{LCD}$)、または $3/4 V_{LCD}$ に接続することができます。

表 39. VLCD レールの GPIO ピンへの接続

| | バイアス | | | CAPA[2:0] ビットによって選択されたピン |
|-----------|---------------|---------------|---------------|--------------------------|
| | 1/2 | 1/3 | 1/4 | |
| LCD_VLCD3 | 未使用 | 未使用 | $3/4 V_{LCD}$ | PB0 |
| LCD_VLCD2 | $1/2 V_{LCD}$ | $2/3 V_{LCD}$ | $1/2 V_{LCD}$ | PB2 |
| LCD_VLCD1 | 未使用 | $1/3 V_{LCD}$ | $1/4 V_{LCD}$ | PB12 |

10.2.3 リファレンス制御およびステータスレジスタ (REF_CFGR3)

REF_CFGR3 レジスタは、リファレンス制御／ステータスレジスタです。このレジスタは、VREFINT および温度センサに関連するすべてのビット／フラグを格納しています。

アドレスオフセット : 0x20

システムリセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------|--------------|-------------------|--------------------|-----------------|-------------------|------------------|-------------------|------|------|--------------|------|------|------|------|------------|
| REF_LOCK | VREFINT_RDYF | VREFINT_COMP_RDYF | VREFINT_ADC_RDYF | SENSOR_ADC_RDYF | REF_HSI48MHz_RDYF | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| rs | r | r | r | r | r | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | ENREF_HSI48MHz | ENBUF_VREFINT_COMP | Res. | Res. | ENBUF_SENSOR_ADC | ENBUF_VREFINT_ADC | Res. | Res. | SEL_VREF_OUT | Res. | Res. | Res. | Res. | EN_VREFINT |
| | | rw | rw | | | rw | rw | | | rw | rw | | | | rw |

ビット 31 REF_LOCK : REF_CFGR3 ロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、リファレンス制御／ステータスレジスタ REF_CFGR3[31:0] の内容全体をロックします。
0 : REF_CFGR3[31:0] ビットは読み出し／書き込み用です。
1 : REF_CFGR3[31:0] ビットは読み出し専用です。

ビット 30 VREFINT_RDYF : VREFINT レディフラグ

このビットは読み出し専用です。このビットは、内部電圧基準、VREFINT の状態を示します。このビットがセットされているとき、VREFINT は BOR、PVD、LCD で使用できます。
0 : VREFINT オフ状態
1 : VREFINT レディ

ビット 29 VREFINT_COMP_RDYF : コンパレータ用 VREFINT レディフラグ

このビットは読み出し専用です。このビットは、バッファされた内部電圧基準、VREFINT の状態を示します。このビットがセットされているとき、VREFINT はコンパレータで使用できます。
0 : コンパレータ用 VREFINT バッファはレディ状態ではありません。
1 : コンパレータ用 VREFINT はレディ状態です。

ビット 28 VREFINT_ADC_RDYF : ADC 用 VREFINT レディフラグ

このビットは読み出し専用です。このビットは、内部電圧基準、VREFINT の状態を示します。このビットがセットされているとき、VREFINT は ADC で使用できます。
0 : ADC 用 VREFINT バッファはレディ状態ではありません。
1 : ADC 用 VREFINT はレディ状態です。

ビット 27 SENSOR_ADC_RDYF : ADC 用温度センサレディフラグ

このビットは読み出し専用です。このビットは、ADC 用温度センサの状態を示します。このビットがセットされているとき、温度センサは ADC で使用できます。
0 : ADC 用温度センサバッファはレディ状態ではありません。
1 : ADC 用温度センサはレディ状態です。

ビット 26 REF_HSI48_RDYF : HSI48 用 VREFINT レディフラグ

このビットは読み出し専用です。このビットは、バッファされた内部電圧基準、VREFINT の状態を示します。このビットは読み出し専用です。このビットがセットされているとき、VREFINT は HSI48 オシレータで使用できます。
0 : HSI48 用 VREFINT バッファはレディ状態ではありません。
1 : VREFINT は HSI48 に対してレディ状態です。

ビット 25:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 ENREF_HSI48 : HSI48 オシレータ用 VREFINT 基準イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。

0 : HSI48 オシレータ用 VREFINT 基準の生成に使用するバッファはスイッチオフされています。

1 : HSI48 オシレータ用 VREFINT 基準の生成に使用するバッファはスイッチオンされています。

ビット 12 ENBUF_VREFINT_COMP : コンパレータ 2 用 VREFINT 基準イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。

0 : コンパレータ 2 用 VREFINT 基準の生成に使用するバッファはスイッチオフされています。

1 : コンパレータ 2 用 VREFINT 基準の生成に使用するバッファはスイッチオンされています。

ビット 11:10 予約済みであり、リセット値に保持する必要があります。

ビット 9 ENBUF_SENSOR_ADC : ADC 用温度センサ基準イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。

0 : ADC 用温度センサ 基準の生成に使用するバッファはスイッチオフされています。

1 : ADC 用温度センサ 基準の生成に使用するバッファはスイッチオンされています。

ビット 8 ENBUF_VREFINT_ADC : ADC 用 VREFINT 基準イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。

0 : ADC 用 VREFINT 基準の生成に使用するバッファはスイッチオフされています。

1 : ADC 用 VREFINT 基準の生成に使用するバッファはスイッチオンされています。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 SEL_VREF_OUT : VREFINT_ADC 接続ビット

これらのビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。これらのビットは、ENBUF_VREFINT_ADC がセットされている場合に、VREFINT_ADC に接続するパッドを選択します。

00 : パッドは接続されません。

01 : PB0 が接続されます。

10 : PB1 が接続されます。

11 : PB0 と PB1 が接続されます。

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 EN_VREFINT : VREFINT イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (REF_LOCK がセットされていない場合のみ)。このビットは、VREFINT 内部基準電圧および温度センサをオンにします。

0 : VREFINT はスイッチオフされています。

1 : VREFINT はスイッチオンされています。

このビットが STOP モードまたは SLEEP モードで 1 にロックされている場合、VREFINT は常にオンです。

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

アドレスオフセット : 0x08

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|------------|----|----|----|------------|----|----|----|------------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EXTI3[3:0] | | | | EXTI2[3:0] | | | | EXTI1[3:0] | | | | EXTI0[3:0] | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:16 予約済み

ビット 15:0 **EXTIx[3:0]** : EXTI x 設定 (x= 0~3)

これらのビットは、EXTIx 外部割り込みのソース入力を選択するために、ソフトウェアで書き込みます。
0000 : PA[x] ピン
0001 : PB[x] ピン
0010 : PC[x] ピン
0011 : PD[x] ピン (PD2 のみ)
0100 : 予約済み
0101 : PH[x] (PH[1:0] のみ)
他の設定は予約済みです。

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

アドレスオフセット : 0x0C

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|------------|----|----|----|------------|----|----|----|------------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EXTI7[3:0] | | | | EXTI6[3:0] | | | | EXTI5[3:0] | | | | EXTI4[3:0] | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:16 予約済み

ビット 15:0 **EXTIx[3:0]** : EXTI x 設定 (x= 4~7)

これらのビットは、EXTIx 外部割り込みのソース入力を選択するために、ソフトウェアで書き込みます。
0000 : PA[x] ピン
0001 : PB[x] ピン
0010 : PC[x] ピン
他の設定は予約済みです。

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

アドレスオフセット : 0x10

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|-------------|-----|-----|-----|-------------|-----|-----|-----|------------|-----|-----|-----|------------|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EXTI11[3:0] | | | | EXTI10[3:0] | | | | EXTI9[3:0] | | | | EXTI8[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:16 予約済み

ビット 15:0 **EXTIx[3:0]** : EXTI x 設定 (x= 8~11)

これらのビットは、EXTIx 外部割り込みのソース入力を選択するために、ソフトウェアで書き込みます。

0000 : PA[x] ピン

0001 : PB[x] ピン

0010 : PC[x] ピン

他の設定は予約済みです。

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

アドレスオフセット : 0x14

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|-------------|-----|-----|-----|-------------|-----|-----|-----|-------------|-----|-----|-----|-------------|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EXTI15[3:0] | | | | EXTI14[3:0] | | | | EXTI13[3:0] | | | | EXTI12[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:16 予約済み

ビット 15:0 **EXTIx[3:0]** : EXTI x 設定 (x= 12~15)

これらのビットは、EXTIx 外部割り込みのソース入力を選択するために、ソフトウェアで書き込みます。

0000 : PA[x] ピン

0001 : PB[x] ピン

0010 : PC[x] ピン

他の設定は予約済みです。

10.2.8 SYSCFG レジスタマップ

次の表に、SYSCFG レジスタマップとリセット値を示します。

表 40. SYSCFG レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
|-----------|----------------|-------------------------------------|--------------|-------------------|------------------|-----------------|----------------|------|------|------|------|------|------|------|------|------|------|------|------|-------------|--------------------|-------------|-------------|------------------|-------------------|------|-------------|--------------|------|------|-------------|---------|----------|------|------------|
| 0x00 | SYSCFG_CFGR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BOOT_MODE | | Res. | Res. | Res. | Res. | Res. | Res. | | MEM_MODE | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | x | x | | | | | | | x | x | | |
| 0x04 | SYSCFG_CFGR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | I2C2_FMP | I2C1_FMP | I2C_PB9_FMP | I2C_PB8_FMP | I2C_PB7_FMP | I2C_PB6_FMP | Res. | Res. | Res. | Res. | | CAPA[2:0] | FWDISEN | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 1 | | |
| 0x08 | SYSCFG_EXTICR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | | | EXTI3[3:0] | | | EXTI2[3:0] | | | | EXTI1[3:0] | | | | EXTI0[3:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x0C | SYSCFG_EXTICR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | | | EXTI7[3:0] | | | EXTI6[3:0] | | | | EXTI5[3:0] | | | | EXTI4[3:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x10 | SYSCFG_EXTICR3 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | | | EXTI11[3:0] | | | EXTI10[3:0] | | | | EXTI9[3:0] | | | | EXTI8[3:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x14 | SYSCFG_EXTICR4 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | | | EXTI15[3:0] | | | EXTI14[3:0] | | | | EXTI13[3:0] | | | | EXTI12[3:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x18 | COMP1_CTRL | セクション 16 : コンパレータ (COMP) を参照してください。 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x1C | COMP2_CTRL | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x20 | REF_CFGR3 | REF_LOCK | VREFINT_RDYF | VREFINT_COMP_RDYF | VREFINT_ADC_RDYF | SENSOR_ADC_RDYF | REF_HSI48_RDYF | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ENREF_HSI48 | ENBUF_VREFINT_COMP | Res. | Res. | ENBUF_SENSOR_ADC | ENBUF_VREFINT_ADC | Res. | Res. | SEL_VREF_OUT | | | | Res. | Res. | Res. | EN_VREFINT |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | 0 | 0 | | | 0 | 0 | | | 0 | 0 | | | | 0 | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

11 ダイレクトメモリアクセスコントローラ (DMA)

11.1 概要

DMA (Direct Memory Access : ダイレクトメモリアクセスコントローラ) は、ペリフェラルとメモリ間、およびメモリ間で高速データ転送を行うために使用します。DMA により、CPU を動作させずに、データを高速で移動することができます。これにより、CPU リソースを他の操作のためにあけておくことができます。

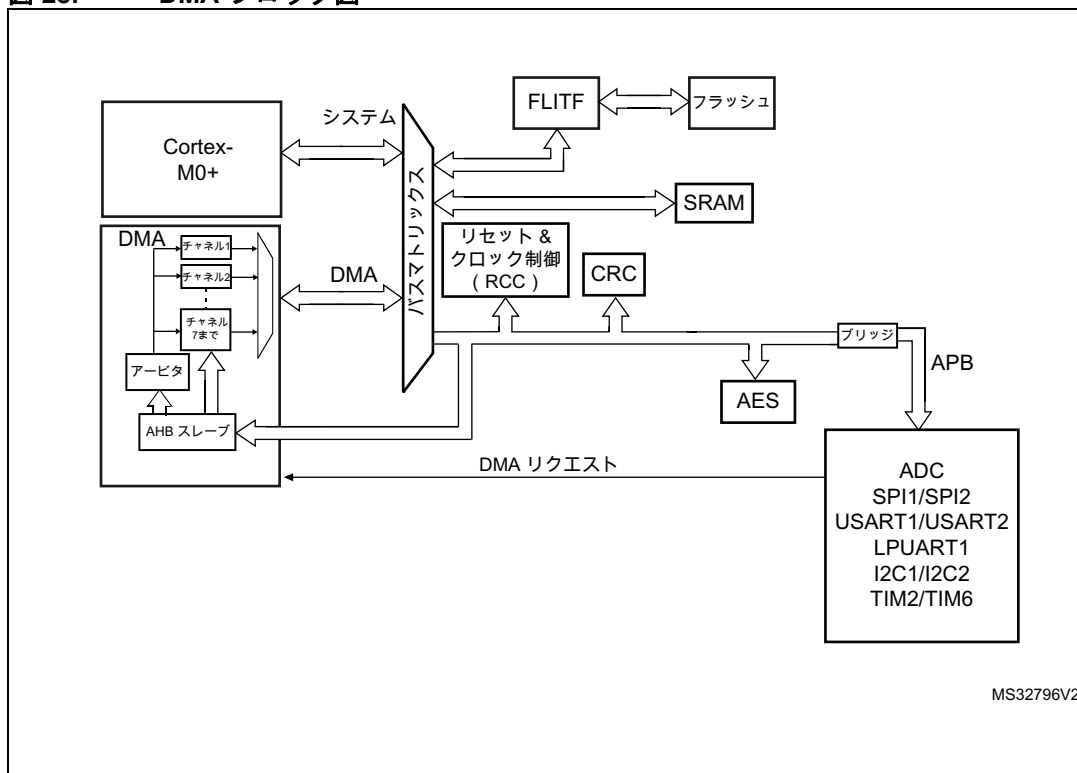
DMA コントローラは、合計 7 本までのチャンネルを搭載し、それぞれが 1 つ以上のペリフェラルからのメモリアセスリクエストを管理する役割を担っています。また、DMA リクエスト間の優先順位を操作するためのアービタも内蔵しています。

11.2 DMA の主な機能

- 最大 7 本の個別に設定可能なチャンネル (リクエスト) を内蔵しています。
- 各チャンネルは、専用のハードウェア DMA リクエストに接続されており、ソフトウェアトリガもサポートしています。この設定はソフトウェアで行われます。
- DMA チャンネルからのリクエスト間の優先順位はソフトウェアでプログラムできます (最高、高、中、低の 4 レベル)。またはレベルが等しい場合はハードウェアによって決まります (リクエスト 1 はリクエスト 2 よりも優先、など)。
- 転送元および転送先の転送サイズ (バイト、ハーフワード、ワード) は個別に設定され、パッキング/アンパッキングをエミュレートします。転送元/転送先のアドレスは、データサイズに基づいて整列させてください。
- サークュラバッファ管理をサポートします。
- 各チャンネルに対する単一割り込みリクエストにおいて、3 つのイベントフラグ (DMA 1/2 転送、DMA 転送完了、DMA 転送エラー) の論理和がとられます。
- メモリ間の転送
- ペリフェラルからメモリ、メモリからペリフェラル、およびペリフェラル間の転送
- フラッシュ、SRAM、APB、AHB ペリフェラルに対して、転送元および転送先としてアクセス
- プログラム可能な転送データ数 : 最大 65535

以下にブロック図を示します。

図 28. DMA ブロック図



MS32796V2

11.3 DMA の機能説明

DMA コントローラは、システムバスを Cortex®-M0+ コアと共有することでダイレクトメモリ転送を行います。DMA リクエストは、CPU と DMA のターゲット転送先が同じである場合に（メモリまたはペリフェラル）、多少のバスサイクルの間、CPU のシステムバスへのアクセスを停止することがあります。バスマトリックスはラウンドロビンスケジューリングを実装します。これにより、CPU のシステムバス帯域幅（メモリとペリフェラルの両方への）の少なくとも半分を確保できます。

11.3.1 DMA トランザクション

イベントの後、ペリフェラルは DMA コントローラにリクエスト信号を送信します。DMA コントローラは、チャンネルの優先順位に応じて、リクエストを処理します。DMA コントローラがペリフェラルにアクセスするとすぐに、DMA コントローラはペリフェラルに確認応答を送信します。ペリフェラルは、DMA コントローラからの確認応答を受け取るとすぐに、そのリクエストを解除します。ペリフェラルによってリクエストが無効にされると、DMA コントローラは確認応答を解除します。さらにリクエストがある場合、ペリフェラルは次のトランザクションを開始できます。

要約すると、各 DMA 転送は次の 3 つの動作で構成されています。

- ペリフェラルデータレジスタまたは現在の内部ペリフェラル/メモリアドレスレジスタを介してアドレス指定されたメモリ位置からのデータのロード。最初の転送に使用される開始アドレスは、DMA_CPARx または DMA_CMARx レジスタでプログラムされたペリフェラル/メモリのベースアドレスです。

- ペリフェラルデータレジスタまたは現在の内部ペリフェラル／メモリアドレスレジスタを介してアドレス指定されたメモリ位置にロードされたデータの格納。最初の転送に使用される開始アドレスは、DMA_CPARx または DMA_CMARx レジスタでプログラムされたペリフェラル／メモリのベースアドレスです。
- まだ実行される予定のトランザクションの数を保持している DMA_CNDTRx レジスタのポストデクリメント。

11.3.2 アービタ

アービタは、チャネルリクエストをその優先順位に基づいて管理し、ペリフェラル／メモリアクセスシーケンスを起動します。

優先順位は 2 段階で管理されます。

- ソフトウェア：各チャネルの優先順位は、DMA_CCRx レジスタで設定できます。4 つのレベルがあります。
 - － 最優先
 - － 高優先
 - － 中優先
 - － 低優先
- ハードウェア：2 つのリクエストのソフトウェア優先順位レベルが同じである場合、小さな番号のチャネルが大きな番号のチャネルよりも優先されます。たとえば、チャネル 2 はチャネル 4 よりも優先されます。

11.3.3 DMA チャネル

各チャネルは、ある固定アドレスにあるペリフェラルレジスタとメモリアドレスの間の DMA 転送を処理します。転送データ数（最大 65535）はプログラム可能です。転送データ項目数を格納しているレジスタは、トランザクションの終了ごとにデクリメントされます。

プログラム可能なデータサイズ

ペリフェラルおよびメモリの転送データサイズは、DMA_CCRx レジスタの PSIZE および MSIZE ビットを介して完全にプログラム可能です。

ポイントのインクリメント

ペリフェラルおよびメモリのポイントは、DMA_CCRx レジスタの PINC および MINC ビットに応じて、各転送後に任意で自動的にポストインクリメントすることができます。インクリメントモードを有効にした場合、次の転送アドレスは、選択されたデータサイズに応じて 1、2、または 4 だけインクリメントされた前回の転送アドレスとなります。最初の転送アドレスは、DMA_CPARx/DMA_CMARx レジスタでプログラムされたアドレスとなります。転送動作中は、これらのレジスタは最初にプログラムされた値を保持します。現在の転送アドレス（現在の内部ペリフェラル／メモリアドレスレジスタ内）に対するソフトウェアによるアクセスはできません。

チャネルがノンサーキュラモードに設定されている場合、転送終了後（転送すべきデータ項目数がゼロに達したとき）、DMA リクエストは処理されません。新たな転送すべきデータ項目数を DMA_CNDTRx レジスタに再ロードするには、DMA チャネルを無効にする必要があります。

注： *DMA チャネルが無効にされると、DMA レジスタはリセットされません。DMA チャネルレジスタ (DMA_CCRx、DMA_CPARx、DMA_CMARx) は、チャネル設定フェーズでプログラムされた初期値のままです。*

サーキュラモードでは、転送終了後、DMA_CNDTRx レジスタには最初にプログラムした値が自動的に再ロードされます。現在の内部アドレスレジスタには、DMA_CPARx/DMA_CMARx レジスタからのベースアドレス値が再ロードされます。

チャンネル設定手順

DMA チャンネル x (x はチャンネル番号) の設定は、以下のシーケンスで行います。

1. DMA_CPARx レジスタに、ペリフェラルレジスタアドレスをセットします。ペリフェラルイベント後は、メモリとの転送に使用されるこのアドレスとデータをやり取りします。
2. DMA_CMARx レジスタにメモリアドレスをセットします。ペリフェラルイベント後は、このメモリに対してデータを読み書きします。
3. 転送すべきデータの総数を DMA_CNDTRx レジスタに設定します。毎回のペリフェラルイベント後に、この値はデクリメントされます。
4. DMA_CCRx レジスタの PL[1:0] ビットを使用して、チャンネルの優先順位を設定します。
5. データ転送方向、サーキュラモード、ペリフェラルとメモリのインクリメントモード、ペリフェラルとメモリのデータサイズ、1/2 転送やフル転送後の割り込みを DMA_CCRx レジスタに設定します。
6. DMA_CCRx レジスタの ENABLE ビットをセットすることでチャンネルを有効にします。

チャンネルが有効になると、そのチャンネルに接続されているペリフェラルからの DMA リクエストを処理することができます。

バイトの半分が転送されると、1/2 転送フラグ (HTIF) がセットされ、1/2 転送割り込みイネーブルビット (HTIE) がセットされている場合には、割り込みが生成されます。転送が終了すると、転送完了フラグ (TCIF) がセットされ、転送完了割り込みイネーブルビット (TCIE) がセットされている場合には、割り込みが生成されます。

サーキュラモード

サーキュラモードを使用すると、サーキュラバッファや連続したデータフロー (ADC スキャンモードなど) を処理することができます。この機能は、DMA_CCRx レジスタの CIRC ビットを使用して有効にできます。サーキュラモードが有効になると、転送されるデータ項目数はチャンネル設定フェーズでプログラムされた初期値が自動的に再ロードされ、DMA リクエストの処理が続行されます。

メモリ間モード

DMA チャンネルは、ペリフェラルからのリクエストによってトリガされなくても機能します。このモードは、メモリ間モードと呼ばれます。

DMA_CCRx レジスタの MEM2MEM ビットがセットされると、チャンネルは、ソフトウェアが DMA_CCRx レジスタの Enable ビット (EN) をセットすることによってそのチャンネルを有効にするとすぐに、転送を開始します。DMA_CNDTRx レジスタがゼロに達すると、転送は停止します。メモリ間モードとサーキュラモードを同時に使用することはできません。

11.3.4 プログラム可能なデータ幅、データの整列、およびエンディアン

PSIZE と MSIZE が等しくない場合、DMA は、表 41 : プログラム可能なデータ幅およびエンディアンの動作 (ビット PINC = MINC = 1 の場合) に記載されているとおりに必要なデータの整列を行います。

表 41. プログラム可能なデータ幅およびエンディアンの動作 (ビット PINC = MINC = 1 の場合)

| 転送元 ポートの 幅 | 転送先 ポートの幅 | 転送する データ 項目の数 (NDT) | 転送元の内容: アドレス/データ | 転送動作 | 転送先の内容: アドレス/データ |
|------------------|--------------|------------------------------|--|--|--|
| 8 | 8 | 4 | @0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3 | 1 : B0[7:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2 : B1[7:0] を 0x1 で読み出し、次に B1[7:0] を 0x1 に書き込みます。 3 : B2[7:0] を 0x2 で読み出し、次に B2[7:0] を 0x2 に書き込みます。 4 : B3[7:0] を 0x3 で読み出し、次に B3[7:0] を 0x3 に書き込みます。 | @0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3 |
| 8 | 16 | 4 | @0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3 | 1 : B0[7:0] を 0x0 で読み出し、次に 00B0[15:0] を 0x0 に書き込みます。 2 : B1[7:0] を 0x1 で読み出し、次に 00B1[15:0] を 0x2 に書き込みます。 3 : B3[7:0] を 0x2 で読み出し、次に 00B2[15:0] を 0x4 に書き込みます。 4 : B4[7:0] を 0x3 で読み出し、次に 00B3[15:0] を 0x6 に書き込みます。 | @0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3 |
| 8 | 32 | 4 | @0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3 | 1 : B0[7:0] を 0x0 で読み出し、次に 000000B0[31:0] を 0x0 に書き込みます。 2 : B1[7:0] を 0x1 で読み出し、次に 000000B1[31:0] を 0x4 に書き込みます。 3 : B3[7:0] を 0x2 で読み出し、次に 000000B2[31:0] を 0x8 に書き込みます。 4 : B4[7:0] を 0x3 で読み出し、次に 000000B3[31:0] を 0xC に書き込みます。 | @0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3 |
| 16 | 8 | 4 | @0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6 | 1 : B1B0[15:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2 : B3B2[15:0] を 0x2 で読み出し、次に B2[7:0] を 0x1 に書き込みます。 3 : B5B4[15:0] を 0x4 で読み出し、次に B4[7:0] を 0x2 に書き込みます。 4 : B7B6[15:0] を 0x6 で読み出し、次に B6[7:0] を 0x3 に書き込みます。 | @0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6 |
| 16 | 16 | 4 | @0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6 | 1 : B1B0[15:0] を 0x0 で読み出し、次に B1B0[15:0] を 0x0 に書き込みます。 2 : B3B2[15:0] を 0x2 で読み出し、次に B3B2[15:0] を 0x2 に書き込みます。 3 : B5B4[15:0] を 0x4 で読み出し、次に B5B4[15:0] を 0x4 に書き込みます。 4 : B7B6[15:0] を 0x6 で読み出し、次に B7B6[15:0] を 0x6 に書き込みます。 | @0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6 |
| 16 | 32 | 4 | @0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6 | 1 : B1B0[15:0] を 0x0 で読み出し、次に 0000B1B0[31:0] を 0x0 に書き込みます。 2 : B3B2[15:0] を 0x2 で読み出し、次に 0000B3B2[31:0] を 0x4 に書き込みます。 3 : B5B4[15:0] を 0x4 で読み出し、次に 0000B5B4[31:0] を 0x8 に書き込みます。 4 : B7B6[15:0] を 0x6 で読み出し、次に 0000B7B6[31:0] を 0xC に書き込みます。 | @0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6 |
| 32 | 8 | 4 | @0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC | 1 : B3B2B1B0[31:0] を 0x0 で読み出し、次に B0[7:0] を 0x0 に書き込みます。 2 : B7B6B5B4[31:0] を 0x4 で読み出し、次に B4[7:0] を 0x1 に書き込みます。 3 : BBBAB9B8[31:0] を 0x8 で読み出し、次に B8[7:0] を 0x2 に書き込みます。 4 : BFBEBDBC[31:0] を 0xC で読み出し、次に BC[7:0] を 0x3 に書き込みます。 | @0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC |
| 32 | 16 | 4 | @0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC | 1 : B3B2B1B0[31:0] を 0x0 で読み出し、次に B1B0[0:] を 0x0 に書き込みます。 2 : B7B6B5B4[31:0] を 0x4 で読み出し、次に B5B4[0:] を 0x1 に書き込みます。 3 : BBBAB9B8[31:0] を 0x8 で読み出し、次に B9B8[0:] を 0x2 に書き込みます。 4 : BFBEBDBC[31:0] を 0xC で読み出し、次に BDBC[0:] を 0x3 に書き込みます。 | @0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC |
| 32 | 32 | 4 | @0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC | 1 : B3B2B1B0[31:0] を 0x0 で読み出し、次に B3B2B1B0[31:0] を 0x0 に書き込みます。 2 : B7B6B5B4[31:0] を 0x4 で読み出し、次に B7B6B5B4[31:0] を 0x4 に書き込みます。 3 : BBBAB9B8[31:0] を 0x8 で読み出し、次に BBBAB9B8[31:0] を 0x8 に書き込みます。 4 : BFBEBDBC[31:0] を 0xC で読み出し、次に BFBEBDBC[31:0] を 0xC に書き込みます。 | @0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC |

バイトまたはハーフワード書き込み動作をサポートしていない AHB ペリフェラルへの対応

DMA が AHB のバイトまたはハーフワード書き込み動作を起動すると、データは HWDATA[31:0] バスの未使用のレーンに複製されます。そのため、使用中の AHB スレーブペリフェラルがバイト/ハーフワード書き込み動作をサポートしておらず (HSIZE がペリフェラルで使用されていない)、かつエラーが 1 件も出ていない場合は、以下の2つの例に示されるように、DMA は 32 HWDATA ビットを書き込みます。

- ハーフワード “0xABCD” を書き込むには、DMA は HWDATA バスを “0xABCDABCD” (HSIZE = ハーフワード) にセットします。
- バイト “0xAB” を書き込むには、DMA は HWDATA バスを “0xABABABAB” (HSIZE = バイト) にセットします。

AHB/APB ブリッジが AHB 32 ビットスレーブペリフェラルである (HSIZE の値は考慮せず) と仮定した場合、以下のようなやり方で、AHB バイトまたはハーフワードのあらゆる動作を 32 ビット APB 動作に変換します。

- データ“0xB0”を 0x0 (または 0x1、0x2、0x3) に書き込む AHB バイト書き込み動作は、データ“0xB0B0B0B0”を 0x0 に書き込む APB ワード書き込み動作に変換されます。
- データ“0xB1B0”を 0x0 (または 0x2) に書き込む AHB ハーフワード書き込み動作は、データ“0xB1B0B1B0”を 0x0 に書き込む APB ワード書き込み動作に変換されます。

たとえば、APB バックアップレジスタ (32 ビットアドレス境界に整列された 16 ビットレジスタ) に書き込みたい場合は、転送元のメモリサイズ (MSIZE) を 16 ビットに、転送先のペリフェラルサイズ (PSIZE) を 32 ビットに設定する必要があります。

11.3.5 エラー管理

DMA 転送エラーは、予約済みアドレス空間に対する読み出しや書き込みによって発生する可能性があります。DMA 読み出しまたは書き込みアクセス中に DMA 転送エラーが発生した場合、障害のあるチャンネルは、ハードウェアが対応するチャンネル設定レジスタ (DMA_CCRx) の該当する EN ビットをクリアすることにより、自動的に無効化されます。DMA_IFR レジスタのチャンネルの転送エラー割り込みフラグ (TEIF) がセットされ、DMA_CCRx レジスタの転送エラー割り込みイネーブルビット (TEIE) がセットされている場合には、割り込みが生成されます。

11.3.6 DMA 割り込み

割り込みは、DMA チャンネルごとの 1/2 転送、転送完了、転送エラーに対して生成されます。高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 42. DMA 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|----------|---------|------------|
| 1/2 転送 | HTIF | HTIE |
| 転送完了 | TCIF | TCIE |
| 転送エラー | TEIF | TEIE |

11.3.7 DMA リクエストマッピング

DMA コントローラ

ペリフェラル (TIM2/6、ADC、DAC、SPI1/2、I2C1/2、AES (AES付きカテゴリ 2 でのみ使用可能)、USART1/2、LPUART1) からのハードウェアリクエストは、DMA チャンネル選択レジスタ (複数可) を介して DMA チャンネル (1~7) に配置されます。1 つのチャンネルでは、一度に 1 つのリクエストのみ有効にしてください。図 29 : DMA リクエストマッピングを参照してください。

ペリフェラル DMA リクエストは、対応するペリフェラルのレジスタの DMA 制御ビットをプログラムすることにより、個別に有効/無効にできます。

図 29. DMA リクエストマッピング

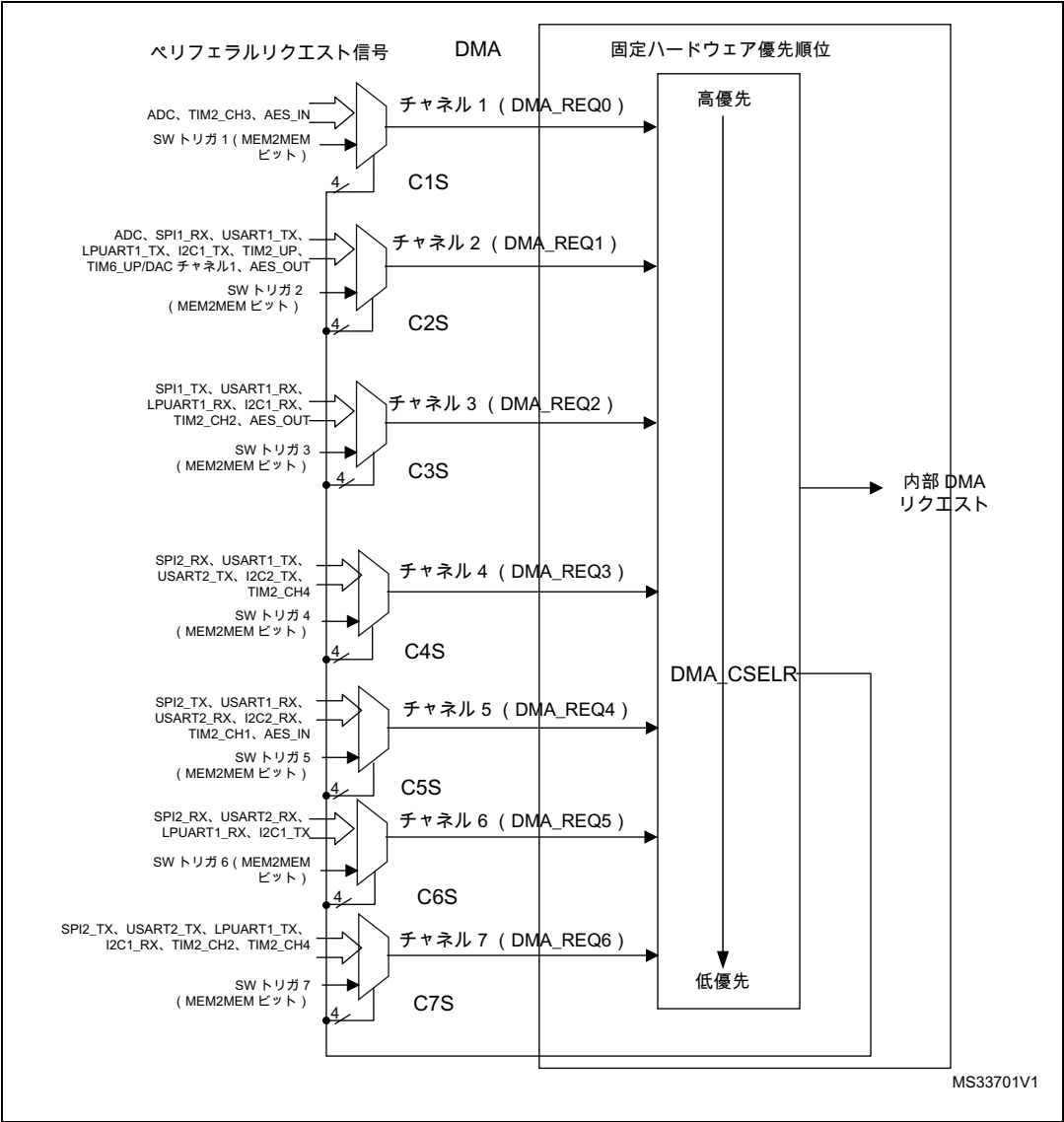


表 43 に、チャンネルごとの DMA リクエストを示します。

表 43. チャンネルごとの DMA リクエストの概要

| リクエスト 番号 | ペリフェラル | チャンネル 1 | チャンネル 2 | チャンネル 3 | チャンネル 4 | チャンネル 5 | チャンネル 6 | チャンネル 7 |
|-------------|------------------------------|--------------|-----------------------|----------------|---------------|---------------|----------------|----------------------|
| 0 | ADC | ADC | ADC | | | | | |
| 1 | SPI1 | | SPI1_RX | SPI1_TX | | | | |
| 2 | SPI2 | | | | SPI2_RX | SPI2_TX | SPI2_RX | SPI2_TX |
| 3 | USART1 | | USART1_TX | USART1_RX | USART1_T X | USART1_R X | | |
| 4 | USART2 | | | | USART2_T X | USART2_R X | USART2_R X | USART2_ TX |
| 5 | LPUART1 | | LPUART1_ TX | LPUART1_ RX | | | LPUART1_ RX | LPUART1_T X |
| 6 | I2C1 | | I2C1_TX | I2C1_RX | | | I2C1_TX | I2C1_RX |
| 7 | I2C2 | | | | I2C2_TX | I2C2_RX | | |
| 8 | TIM2 | TIM2_ CH3 | TIM2_UP | TIM2_CH2 | TIM2_CH4 | TIM2_CH1 | | TIM2_CH2 TIM2_CH4 |
| 9 | TIM6_UP /DAC_ channel1 | | TIM6/DAC_ channel1 | | | | | |
| 11 | AES ⁽¹⁾ | AES_IN | AES_OUT | AES_OUT | | AES_IN | | |

1. AES付きカテゴリ 2 でのみ使用可能。

11.4 DMA レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、バイト (8 ビット)、ハーフワード (16 ビット)、またはワード (32 ビット) 単位でアクセスできます。

11.4.1 DMA 割り込みステータスレジスタ (DMA_ISR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|-------|-------|-------|------|-------|-------|-------|------|-------|-------|-------|------|-------|-------|-------|------|
| Res. | Res. | Res. | Res. | TEIF7 | HTIF7 | TCIF7 | GIF7 | TEIF6 | HTIF6 | TCIF6 | GIF6 | TEIF5 | HTIF5 | TCIF5 | GIF5 |
| | | | | r | r | r | r | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TEIF4 | HTIF4 | TCIF4 | GIF4 | TEIF3 | HTIF3 | TCIF3 | GIF3 | TEIF2 | HTIF2 | TCIF2 | GIF2 | TEIF1 | HTIF1 | TCIF1 | GIF1 |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27、23、19、**TEIFx** : チャネル x 転送エラーフラグ (x = 1~7)

15、11、7、3 このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

0 : チャネル x に転送エラーは発生していません。

1 : チャネル x に転送エラー (TE) が発生しました。

ビット 26、22、18、**HTIFx** : チャネル x 1/2 転送フラグ (x = 1~7)

14、10、6、2 このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

0 : チャネル x に 1/2 転送 (HT) イベントは発生していません。

1 : チャネル x に 1/2 転送 (HT) イベントが発生しました。

ビット 25、21、17、**TCIFx** : チャネル x 転送完了フラグ (x = 1~7)

13、9、5、1 このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

0 : チャネル x の転送完了 (TC) イベントは発生していません。

1 : チャネル x に転送完了 (TC) イベントが発生しました。

ビット 24、20、16、**GIFx** : チャネル x グローバル割り込みフラグ (x = 1~7)

12、8、4、0 このビットは、ハードウェアによってセットされます。ソフトウェアで DMA_IFCR レジスタの対応するビットに 1 を書き込むことによってクリアされます。

0 : チャネル x に TE、HT、または TC イベントは発生していません。

1 : チャネル x に TE、HT、または TC イベントが発生しました。

11.4.2 DMA 割り込みフラグクリアレジスタ (DMA_IFCR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|--------|--------|--------|-------|--------|--------|--------|-------|--------|--------|--------|-------|--------|--------|--------|-------|
| Res. | Res. | Res. | Res. | CTEIF7 | CHTIF7 | CTCIF7 | CGIF7 | CTEIF6 | CHTIF6 | CTCIF6 | CGIF6 | CTEIF5 | CHTIF5 | CTCIF5 | CGIF5 |
| | | | | w | w | w | w | w | w | w | w | w | w | w | w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CTEIF4 | CHTIF4 | CTCIF4 | CGIF4 | CTEIF3 | CHTIF3 | CTCIF3 | CGIF3 | CTEIF2 | CHTIF2 | CTCIF2 | CGIF2 | CTEIF1 | CHTIF1 | CTCIF1 | CGIF1 |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27、23、19、**CTEIFx** : チャネル x 転送エラークリア (x = 1~7)

15、11、7、3 このビットはソフトウェアによってセット/クリアされます。

0 : 影響なし

1 : DMA_ISR レジスタの対応する TEIF フラグをクリアします。

ビット 26、22、18、**CHTIFx** : チャネル x 1/2 転送クリア (x = 1~7)

14、10、6、2 このビットはソフトウェアによってセット/クリアされます。

0 : 影響なし

1 : DMA_ISR レジスタの対応する HTIF フラグをクリアします。

ビット 25、21、17、**CTCIFx** : チャネル x 転送完了クリア (x = 1~7)

13、9、5、1 このビットはソフトウェアによってセット/クリアされます。

0 : 影響なし

1 : DMA_ISR レジスタの対応する TCIF フラグをクリアします。

ビット 24、20、16、**CGIFx** : チャネル x グローバル割り込みクリア (x = 1~7)

12、8、4、0 このビットはソフトウェアによってセット/クリアされます。

0 : 影響なし

1 : DMA_ISR レジスタの GIF、TEIF、HTIF、および TCIF フラグをクリアします。

11.4.3 DMA チャネル x 設定レジスタ (DMA_CCRx) (x = 1~7、x = チャネル番号)

アドレスオフセット : 0x08 + 0d20 x (チャネル番号 - 1)

リセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|---------|---------|------|------------|------|------------|------|------|------|------|------|------|------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | MEM2MEM | PL[1:0] | | MSIZE[1:0] | | PSIZE[1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **MEM2MEM** : メモリ間モード

このビットはソフトウェアによってセット/クリアされます。

- 0 : メモリ間モードは無効です。
- 1 : メモリ間モードは有効です。

ビット 13:12 **PL[1:0]** : チャネル優先順位レベル

これらのビットは、ソフトウェアによってセット/クリアされます。

- 00 : 低優先
- 01 : 中優先
- 10 : 高優先
- 11 : 最優先

ビット 11:10 **MSIZE[1:0]** : メモリサイズ

これらのビットは、ソフトウェアによってセット/クリアされます。

- 00 : 8 ビット
- 01 : 16 ビット
- 10 : 32 ビット
- 11 : 予約済み

ビット 9:8 **PSIZE[1:0]** : ペリフェラルサイズ

これらのビットは、ソフトウェアによってセット/クリアされます。

- 00 : 8 ビット
- 01 : 16 ビット
- 10 : 32 ビット
- 11 : 予約済み

ビット 7 **MINC** : メモリインクリメントモード

このビットはソフトウェアによってセット/クリアされます。

- 0 : メモリインクリメントモードは無効です。
- 1 : メモリインクリメントモードは有効です。

ビット 6 **PINC** : ペリフェラルインクリメントモード

このビットはソフトウェアによってセット/クリアされます。

- 0 : ペリフェラルインクリメントモードは無効です。
- 1 : ペリフェラルインクリメントモードは有効です。

ビット 5 **CIRC** : サーキュラモード

このビットはソフトウェアによってセット/クリアされます。

- 0 : サーキュラモードは無効です。
- 1 : サーキュラモードは有効です。

ビット 4 DIR : データ転送方向

このビットはソフトウェアによってセット／クリアされます。

- 0 : ペリフェラルから読み出します。
- 1 : メモリから読み出します。

ビット 3 TEIE : 転送エラー割り込みイネーブル

このビットはソフトウェアによってセット／クリアされます。

- 0 : TE 割り込みは無効です。
- 1 : TE 割り込みは有効です。

ビット 2 HTIE : 1/2 転送割り込みイネーブル

このビットはソフトウェアによってセット／クリアされます。

- 0 : HT 割り込みは無効です。
- 1 : HT 割り込みは有効です。

ビット 1 TCIE : 転送完了割り込みイネーブル

このビットはソフトウェアによってセット／クリアされます。

- 0 : TC 割り込みは無効です。
- 1 : TC 割り込みは有効です。

ビット 0 EN : チャネルイネーブル

このビットはソフトウェアによってセット／クリアされます。

- 0 : チャネルは無効です。
- 1 : チャネルは有効です。

11.4.4 DMA チャンネル x データ数レジスタ (DMA_CNDTRx) (x = 1~7、x = チャンネル番号)

アドレスオフセット : 0x0C + 0d20 x (チャンネル番号 - 1)

リセット値 : 0x0000 0000

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **NDT[15:0]** : 転送データ項目の数

転送データ数 : 0~65535このレジスタは、チャンネルが無効な場合にのみ書き込みできます。チャンネルが有効になると、このレジスタは読み出し専用になり、送信されるべき残りのバイトを指し示します。このレジスタは、各 DMA 転送後にデクリメントされます。

転送が完了すると、このレジスタは、ゼロのままにすることもできれば、チャンネルが自動再ロードモードに設定されている場合には、事前にプログラムされた値を自動的に再ロードすることもできます。

このレジスタの値が 0 の場合、チャンネルが有効であるか否かにかかわらず、トランザクションは処理できません。

11.4.5 DMA チャンネル x ペリフェラルアドレスレジスタ (DMA_CPARx) (x = 1~7、x = チャンネル番号)

アドレスオフセット : 0x10 + 0d20 x (チャンネル番号 - 1)

リセット値 : 0x0000 0000

チャンネルが有効なときには、このレジスタへは書き込みを *行わない* てください。

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

ビット 31:0 **PA[31:0]** : ペリフェラルアドレス

データの読み出し／書き込みが行われるペリフェラルデータレジスタのベースアドレス。

PSIZE が 01 (16 ビット) のとき、PA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

PSIZE が 10 (32ビット) のとき、PA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

11.4.6 DMA チャンネル x メモリアドレスレジスタ (DMA_CMARx)
(x = 1~7、x = チャンネル番号)

アドレスオフセット : 0x14 + 0d20 x (チャンネル番号 - 1)

リセット値 : 0x0000 0000

チャンネルが有効なときには、このレジスタへは書き込みを *行わない* てください。

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

ビット 31:0 **MA[31:0]** : メモリアドレス

データの読み出し／書き込みが行われるメモリ領域のベースアドレス。

MSIZE が 01 (16 ビット) のとき、MA[0] ビットは無視されます。アクセスは、自動的にハーフワードアドレスに整列されます。

MSIZE が 10 (32ビット) のとき、MA[1:0] は無視されます。アクセスは、自動的にワードアドレスに整列されます。

11.4.7 DMA チャンネル選択レジスタ (DMA_CSELR)

アドレスオフセット : 0xA8

リセット値 : 0x0000 0000

このレジスタは、DMA チャンネルの再配置を管理するために使用されます (図 29 を参照)。

| | | | | | | | | | | | | | | | |
|-----------|------|------|------|-----------|----|----|----|-----------|----|----|----|-----------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | C7S [3:0] | | | | C6S [3:0] | | | | C5S [3:0] | | | |
| | | | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| C4S [3:0] | | | | C3S [3:0] | | | | C2S [3:0] | | | | C1S [3:0] | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **C7S[3:0]** : DMA チャンネル 7 選択

- 0010 : DMA チャンネル 7 は SPI2_TX に再配置されます。
- 0100 : DMA チャンネル 7 は USART2_TX に再配置されます。
- 0101 : DMA チャンネル 7 は LPUART1_TX に再配置されます。
- 0110 : DMA チャンネル 7 は I2C1_RX に再配置されます。
- 1000 : DMA チャンネル 7 は TIM2_CH2/TIM2_CH4 に再配置されます。
- その他の設定 : DMA チャンネル 7 は再配置されません。

ビット 23:20 **C6S[3:0]** : DMA チャンネル 6 選択

- 0010 : DMA チャンネル 6 は SPI2_RX に再配置されます。
- 0100 : DMA チャンネル 6 は USART2_RX に再配置されます。
- 0101 : DMA チャンネル 6 は LPUART1_RX に再配置されます。
- 0110 : DMA チャンネル 6 は I2C1_TX に再配置されます。
- その他の設定 : DMA チャンネル 6 は再配置されません。

ビット 19:16 **C5S[3:0]** : DMA チャンネル 5 選択

- 0010 : DMA チャンネル 5 は SPI2_TX に再配置されます。
- 0011 : DMA チャンネル 5 は USART1_RX に再配置されます。
- 0100 : DMA チャンネル 5 は USART2_RX に再配置されます。
- 0111 : DMA チャンネル 5 は I2C2_RX に再配置されます。
- 1000 : DMA チャンネル 5 は TIM2_CH1 に再配置されます。
- 1011 : DMA チャンネル 5 は AES_IN に再配置されます (AES付きカテゴリ 2 でのみ使用可能, そうしないと、再配置されません)。
- その他の設定 : DMA チャンネル 5 は再配置されません。

ビット 15:12 **C4S[3:0]** : DMA チャンネル 4 選択

- 0010 : DMA チャンネル 4 は SPI2_RX に再配置されます。
- 0011 : DMA チャンネル 4 は USART1_TX に再配置されます。
- 0100 : DMA チャンネル 4 は USART2_TX に再配置されます。
- 0111 : DMA チャンネル 4 は I2C2_TX に再配置されます。
- 1000 : DMA チャンネル 4 は TIM2_CH4 に再配置されます。
- その他の設定 : DMA チャンネル 4 は再配置されません。

ビット 11:8 C3S[3:0] : DMA チャンネル 3 選択

0001 : DMA チャンネル 3 は SPI1_TX に再配置されます。
0011 : DMA チャンネル 3 は USART1_RX に再配置されます。
0101 : DMA チャンネル 3 は LPUART1_RX に再配置されます。
0110 : DMA チャンネル 3 は I2C1_RX に再配置されます。
1000 : DMA チャンネル 3 は TIM2_CH2 に再配置されます。
1011 : DMA チャンネル 3 は AES_OUT に再配置されます (AES 付きカテゴリ 2 でのみ使用可能, そうしないと、再配置されません)。
その他の設定 : DMA チャンネル 3 は再配置されません。

ビット 7:4 C2S[3:0] : DMA チャンネル 2 選択

0000 : DMA チャンネル 2 は ADC に再配置されます。
0001 : DMA チャンネル 2 は SPI1_RX に再配置されます。
0011 : DMA チャンネル 2 は USART1_TX に再配置されます。
0101 : DMA チャンネル 2 は LPUART1_TX に再配置されます。
0110 : DMA チャンネル 2 は I2C1_TX に再配置されます。
1000 : DMA チャンネル 2 は TIM2_UP に再配置されます。
1001 : DMA チャンネル 2 は TIM6_UPDAC チャンネル 1 に再配置されます。
1011 : DMA チャンネル 2 は AES_OUT に再配置されます (AES 付きカテゴリ 2 でのみ使用可能, そうしないと、再配置されません)。
その他の設定 : DMA チャンネル 2 は再配置されません。

ビット 3:0 C1S[3:0] : DMA チャンネル 1 選択

0000 : DMA チャンネル 1 は ADC に再配置されます。
1000 : DMA チャンネル 1 は TIM2_CH3 に再配置されます。
1011 : DMA チャンネル 1 は AES_IN に再配置されます (AES 付きカテゴリ 2 でのみ使用可能, そうしないと、再配置されません)。
その他の設定 : DMA チャンネル 1 は再配置されません。

次の表に、DMA レジスタマップとリセット値を示します。

表 44. DMA レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | | |
|-----------|------------|----------|------|------|------|----------|------|------|------|------|----------|------|------|------|------|----------|------|-----------|---------|----------|----------|-------------|----|-------------|---|----------|------|------|-----|------|----------|------|----|---|---|----------|--|--|--|--|
| 0x58 | DMA_CCR5 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MEM2MEM | PL[1:0] | | MSIZE[1:0] | | PSIZE[1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x5C | DMA_CNDTR5 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | NDT[15:0] | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x60 | DMA_CPAR5 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x64 | DMA_CMAR5 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x068 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x06C | DMA_CCR6 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x070 | DMA_CNDTR6 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | NDT[15:0] | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x074 | DMA_CPAR6 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x078 | DMA_CMAR6 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x07C | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x080 | DMA_CCR7 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | |
| 0x084 | DMA_CNDTR7 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | NDT[15:0] | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x088 | DMA_CPAR7 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x08C | DMA_CMAR7 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x090 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0A8 | DMA_CSELR | Res. | Res. | Res. | Res. | C7S[3:0] | | | | | C6S[3:0] | | | | | C5S[3:0] | | | | | C4S[3:0] | | | | | C3S[3:0] | | | | | C2S[3:0] | | | | | C1S[3:0] | | | | |
| | リセット値 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

12 ネスト化されたベクタ割り込みコントローラ (NVIC)

12.1 主な機能

- 39 本のマスク可能な割り込みチャネルを内蔵(表 45 を参照)。Cortex®-M0+の 16 本の割り込みラインは数に含まれていません。
- 16 のプログラム可能な優先レベル (4 ビットの割り込み優先順位を使用)
- 遅延時間の少ない例外および割り込み処理
- 電源管理制御
- システム制御レジスタの実装

NVIC とプロセッサコアのインタフェースを密に結合することで、割り込み処理の遅延を低減し、後着割り込みを効率的に処理することができます。

コア例外を含むすべての割り込みは、NVIC によって管理されます。例外と NVIC プログラミングの詳細については、プログラミングマニュアル PM0056 を参照してください。

12.2 SysTick 較正值レジスタ

SysTick 較正值は 4000 に固定されますので、SysTick クロックを 4 MHz (最大 HCLK/8) に設定した状態での基準タイムベースは 1 ms になります。

12.3 割り込みベクタと例外ベクタ

表 45 は、STM32L0x3 デバイスのベクタテーブルです。

表 45. ベクタテーブル

| 位置 | 優先順位 | 優先種別 | 項目 (略称) | 説明 | アドレス |
|----|------|------|--------------------|--|---------------------------|
| | - | - | - | 予約済み | 0x0000_0000 |
| | -3 | 固定 | リセット | リセット | 0x0000_0004 |
| | -2 | 固定 | NMI_Handler | ノンマスクابل割り込み。RCC クロックセキュリティシステム (CSS) は NMI ベクタにリンクされます。 | 0x0000_0008 |
| | -1 | 固定 | HardFault_Handler | あらゆる種類の異常 | 0x0000_000C |
| | 0 | 設定可能 | MemManage_Handler | メモリ管理 | 0x0000_0010 |
| | 1 | 設定可能 | BusFault_Handler | プリフェッチ異常、メモリアクセス異常 | 0x0000_0014 |
| | 2 | 設定可能 | UsageFault_Handler | 未定義命令または無効状態 | 0x0000_0018 |
| | - | - | - | 予約済み | 0x0000_001C - 0x0000_002B |
| | 3 | 設定可能 | SVC_Handler | SWI 命令によるシステムサービスコール | 0x0000_002C |
| | 4 | 設定可能 | DebugMon_Handler | デバッグモニタ | 0x0000_0030 |
| | - | - | - | 予約済み | 0x0000_0034 |
| | 5 | 設定可能 | PendSV_Handler | ペンディング可能なシステムサービスリクエスト | 0x0000_0038 |

表 45. ベクタテーブル (続き)

| 位置 | 優先順位 | 優先種別 | 項目 (略称) | 説明 | アドレス |
|----|------|------|------------------------------------|---|-------------|
| | 6 | 設定可能 | SysTick_Handler | システムティックタイマ | 0x0000_003C |
| 0 | 7 | 設定可能 | WWDG | ウィンドウ型ウォッチドッグ割り込み | 0x0000_0040 |
| 1 | 8 | 設定可能 | PVD | PVD 検出割り込み (EXTI ライン経由) | 0x0000_0044 |
| 2 | 9 | 設定可能 | RTC | RTC グローバル割り込み (EXTI17/19/20 ライン経由) | 0x0000_0048 |
| 3 | 10 | 設定可能 | FLASH | フラッシュメモリおよびデータ EEPROM グローバル割り込み | 0x0000_004C |
| 4 | 11 | 設定可能 | RCC_CR | RCC および CRS グローバル割り込み | 0x0000_0050 |
| 5 | 12 | 設定可能 | EXTI[1:0] | EXTI ライン 0 および 1 割り込み | 0x0000_0054 |
| 6 | 13 | 設定可能 | EXTI[3:2] | EXTI ライン 2 および 3 割り込み | 0x0000_0058 |
| 7 | 14 | 設定可能 | EXTI[15:4] | EXTI ライン 4~15 割り込み | 0x0000_005C |
| 8 | 15 | 設定可能 | TSC | タッチ検出コントローラ割り込み | 0x0000_0060 |
| 9 | 16 | 設定可能 | DMA1_Channel1 | DMA1 チャンネル 1 グローバル割り込み | 0x0000_0064 |
| 10 | 17 | 設定可能 | DMA1_Channel[3:2] | DMA1 チャンネル 2 および 3 割り込み | 0x0000_0068 |
| 11 | 18 | 設定可能 | DMA1_Channel[7:4] | DMA1 チャンネル 4~7 割り込み | 0x0000_006C |
| 12 | 19 | 設定可能 | ADC_COMP | ADC およびコンパレータ割り込み (EXTI21 および EXTI22 経由) | 0x0000_0070 |
| 13 | 20 | 設定可能 | LPTIM1 | LPTIMER1 割り込み (EXTI29 経由) | 0x0000_0074 |
| 14 | 21 | 設定可能 | - | 予約済み | 0x0000_0078 |
| 15 | 22 | 設定可能 | TIM2 | TIMER2 グローバル割り込み | 0x0000_007C |
| 16 | 23 | 設定可能 | - | 予約済み | 0x0000_0080 |
| 17 | 24 | 設定可能 | TIM6_DAC | TIMER6 グローバル割り込みおよび DAC 割り込み | 0x0000_0084 |
| 18 | 25 | 設定可能 | - | 予約済み | 0x0000_0088 |
| 19 | 26 | 設定可能 | - | 予約済み | 0x0000_008C |
| 20 | 27 | 設定可能 | TIM21 | TIMER21 グローバル割り込み | 0x0000_0090 |
| 21 | 28 | 設定可能 | - | 予約済み | 0x0000_0094 |
| 22 | 29 | 設定可能 | TIM22 | TIMER22 グローバル割り込み | 0x0000_0098 |
| 23 | 30 | 設定可能 | I2C1 | I2C1 グローバル割り込み (EXTI23 経由) | 0x0000_009C |
| 24 | 31 | 設定可能 | I2C2 | I2C2 グローバル割り込み | 0x0000_00A0 |
| 25 | 32 | 設定可能 | SPI1 | SPI1 グローバル割り込み | 0x0000_00A4 |
| 26 | 33 | 設定可能 | SPI2 | SPI2 グローバル割り込み | 0x0000_00A8 |
| 27 | 34 | 設定可能 | USART1 | USART1 グローバル割り込み (EXTI25 経由) | 0x0000_00AC |
| 28 | 35 | 設定可能 | USART2 | USART2 グローバル割り込み (EXTI26 経由) | 0x0000_00B0 |
| 29 | 36 | 設定可能 | LPUART1 + AES ⁽¹⁾ + RNG | LPUART1 グローバル割り込み (EXTI28 経由) + AES グローバル割り込み + RNG グローバル割り込み | 0x0000_00B4 |

表 45. ベクタテーブル (続き)

| 位置 | 優先順位 | 優先種別 | 項目 (略称) | 説明 | アドレス |
|----|------|------|---------|--------------------------|-------------|
| 30 | 37 | 設定可能 | LCD | LCD グローバル割り込み | 0x0000_00B8 |
| 31 | 38 | 設定可能 | USB | USB イベント割り込み (EXTI18 経由) | 0x0000_00BC |

1. AES付きカテゴリ 2 のみ使用可能。

13 拡張割り込み／イベントコントローラ (EXTI)

13.1 概要

拡張割り込み／イベントコントローラ (EXTI) は、外部および内部で非同期的に発生するイベント／割り込みを管理し、CPU／割り込みコントローラへのイベントリクエスト、およびパワーコントローラへのウェイクアップリクエストを生成します。

EXTI を使用することで、デバイスを STOP モードからウェイクアップできるイベントラインを最大 30 本まで管理することが可能です。

中には設定が可能なラインがあります。この場合、アクティブエッジは個別に選択することができ、ステータスフラグは割り込みのソースを示します。設定可能なラインを使用するのは、外部 I/O 割り込みと、いくつかのペリフェラルです。また、ダイレクトなラインもあり、それらを使用するのは、STOP イベントや割り込みからのウェイクアップを生成するいくつかのペリフェラルです。この場合、ステータスフラグはペリフェラルから供給されます。

割り込みやイベントの生成のために、各ラインを個別にマスクすることができます。

また、EXTI コントローラは、専用レジスタへのプログラミングを行うことによって、対応するハードウェアイベントラインと多重化されているソフトウェアによるイベントや割り込みのエミュレーションを可能にします。

13.2 EXTI の主な機能

EXTI の主な機能は以下のとおりです。

- 最大 30 のイベント／割り込みリクエストの生成
 - － 22 本の設定可能なライン
 - － 6 本のダイレクトライン
- 各イベント／割り込みラインの個別マスク
- 設定可能な立ち上がりまたは立ち下がりエッジ (設定可能なラインのみ)
- 専用のステータスビット (設定可能なラインのみ)
- イベント／割り込みリクエストのエミュレーション (設定可能なラインのみ)

13.3 EXTI の機能説明

設定可能な割り込みラインの場合、割り込みを生成するためには、それらを設定し、有効化にする必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、割り込みマスクレジスタの対応するビットに"1"を書き込むことによって割り込みリクエストを有効にします。選択されたエッジが外部割り込みラインで発生すると、割り込みリクエストが生成されます。割り込みラインに対応するペンディングビットもセットされます。このリクエストは、ペンディングレジスタに"1"を書き込むことによってリセットされます。

ダイレクト割り込みラインの場合、割り込みは割り込みマスクレジスタのデフォルト値によって有効化され、ペンディングレジスタには対応するペンディングビットはセットされていません。

イベントを生成するには、イベントラインを設定し、有効にする必要があります。それには、2 つのトリガレジスタに希望するエッジ検出をプログラムし、イベントマスクレジスタの対応するビットに"1"を書き込むことによってイベントリクエストを有効にします。選択されたエッジがイベントライン

で発生すると、イベントパルスが生成されます。イベントラインに対応するペンディングビットはセットされません。

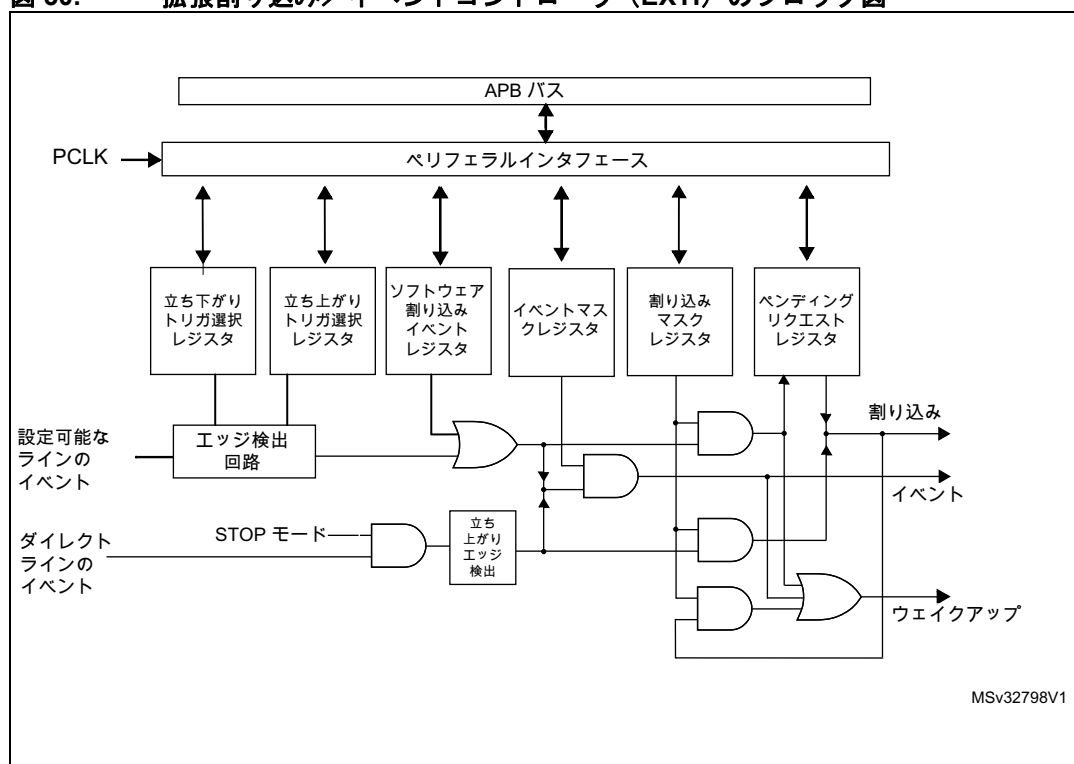
設定可能なラインの場合、割り込み／イベントリクエストは、ソフトウェア割り込み／イベントレジスタに"1"を書き込むことによって、ソフトウェアによって生成することもできます。

注： **ダイレクトラインに関連する割り込みやイベントは、システムがSTOP モードのときのみトリガされます。システムがまだ動作している場合は、EXTI は割り込み／イベントを生成しません。**

13.3.1 EXTI ブロック図

図 30 にブロック図を示します。

図 30. 拡張割り込み／イベントコントローラ (EXTI) のブロック図



13.3.2 ウェイクアップイベント管理

STM32L0x3 マイクロコントローラは、外部イベントや内部イベントを処理することによって、コアをウェイクアップすることができます (WFE)。ウェイクアップイベントは、次のいずれかによって生成できます。

- ペリフェラル制御レジスタで割り込みを有効にし、NVIC では有効にせず、Cortex®-M0+ システム制御レジスタの SEVONPEND ビットを有効にします。MCU が WFE からリスタートするときには、ペリフェラル割り込みペンディングビットと (NVIC 割り込みクリアペンディングレジスタの) ペリフェラル NVIC IRQ チャンネルペンディングビットをクリアする必要があります。
- あるいは、EXTI ラインをイベントモードに設定します。CPU が WFE からリスタートするときには、イベントラインに対応するペンディングビットはセットされていないので、ペリフェラル割り込みペンディングビットや NVIC IRQ チャンネルペンディングビットをクリアする必要はありません。

13.3.3 ペリフェラル非同期割り込み

一部のペリフェラルは、システムが RUN モードまたは STOP モードにあるときにイベントを生成することができるので、システムを STOP モードからウェイクアップさせることが可能になります。

そのためには、ペリフェラルは、(APB クロックなどのシステムクロックに) 同期されたイベントと非同期のイベントの両方を生成します。この非同期イベントは、EXTI のダイレクトラインに接続されます。

注： **STOP モードからのウェイクアップ機能を持つ数少ないペリフェラルは、EXTI の設定可能なラインに接続されます。この場合、STOP モードからのウェイクアップを可能にするには EXTI の設定が必要です。**

13.3.4 ハードウェア割り込みの選択

ラインを割り込みソースとして設定するには、次の手順を使用します。

1. 割り込みラインのマスクビットを設定します (EXTI_IMR)。
2. 割り込みラインのトリガ選択ビットを設定します (EXTI_RTISR および EXTI_FTISR)。
3. ラインのいずれかからの割り込みに対して正しく確認応答できるように、外部割り込みコントローラ (EXTI) に配置された NVIC IRQ チャンネルを制御するイネーブルビットとマスクビットを設定します。

ダイレクトラインには EXTI の設定は必要ありません。

13.3.5 ハードウェアイベントの選択

ラインを割り込みソースとして設定するには、次の手順を使用します。

1. イベントラインのマスクビットを設定します (EXTI_EMR)。
2. イベントラインのトリガ選択ビットを設定します (EXTI_RTISR および EXTI_FTISR)。

13.3.6 ソフトウェア割り込み／イベントの選択

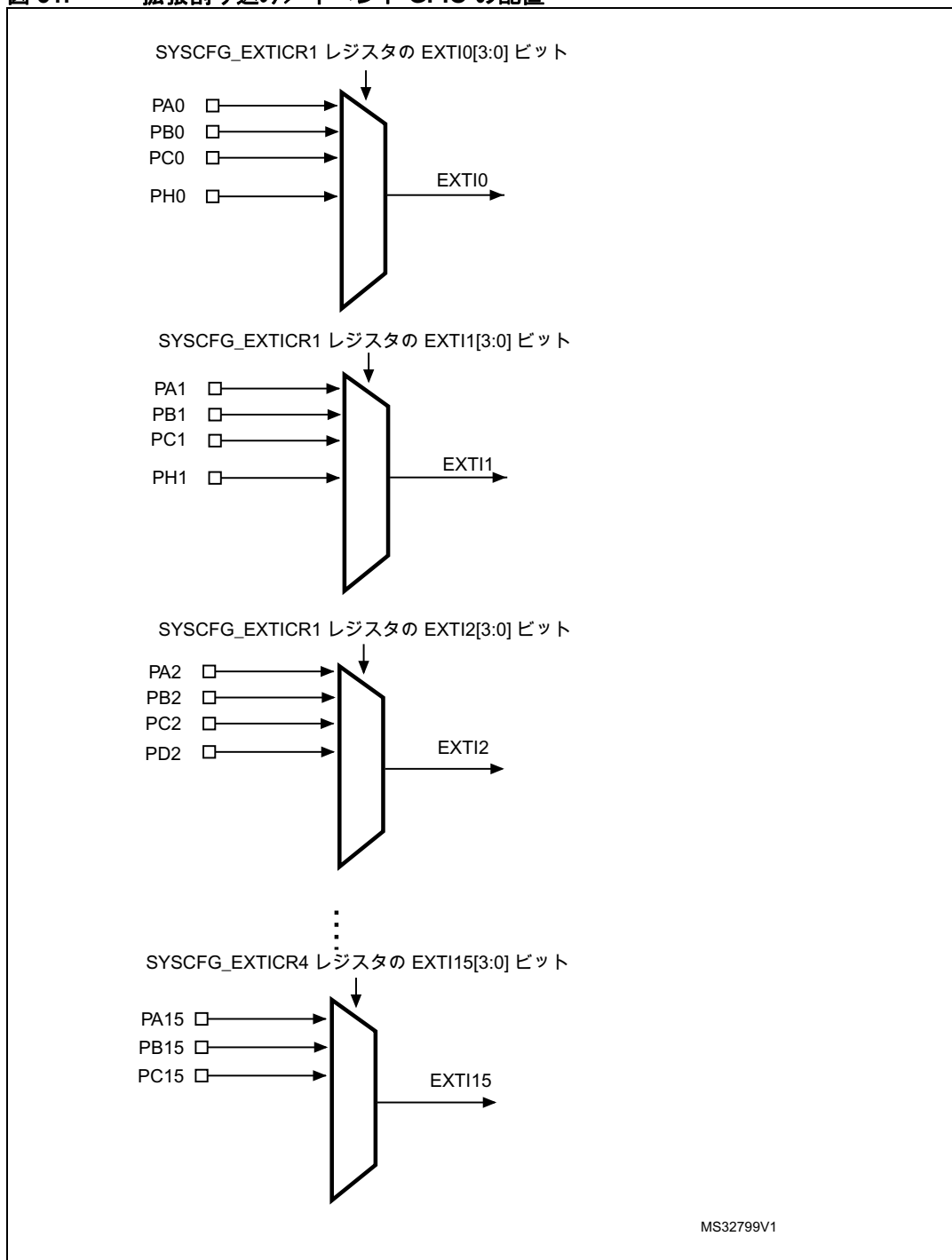
設定可能なラインのいずれかを、ソフトウェア割り込み／イベントラインとして設定できます。ソフトウェア割り込みを生成するには、以下の手順に従ってください。

1. 割り込み／イベントラインのマスクビットを設定します (EXTI_IMR、EXTI_EMR)。
2. ソフトウェア割り込みレジスタの必要なビットをセットします (EXTI_SWIER)。

13.4 EXTI 割り込み／イベントラインの配置

STM32L0x3では、30本の割り込み／イベントラインが使用できます。[図 31](#)に示すように、GPIOは16本の設定可能な割り込み／イベントラインに接続されます。

図 31. 拡張割り込み／イベント GPIO の配置



注： 利用可能なI/Oポートのリストについては、データシートを参照してください。

表 46 : EXTI のラインの接続に示すように、30 本のラインが接続されます。

表 46. EXTI のラインの接続

| EXTI のライン | ラインソース | ラインタイプ |
|-----------|---------------------------------|--------|
| 0~15 | GPIO | 設定可能 |
| 16 | PVD | 設定可能 |
| 17 | RTC アラーム | 設定可能 |
| 18 | USB ウェイクアップイベント | ダイレクト |
| 19 | RTC タンパ、タイムスタンプ、 または CSS_LSE | 設定可能 |
| 20 | RTC ウェイクアップタイマ | 設定可能 |
| 21 | COMP1 出力 | 設定可能 |
| 22 | COMP2 出力 | 設定可能 |
| 23 | I2C1 ウェイクアップ | ダイレクト |
| 24 | 予約済み | |
| 25 | USART 1 ウェイクアップ | ダイレクト |
| 26 | USART2 ウェイクアップ | ダイレクト |
| 27 | 予約済み | |
| 28 | LPUART1 ウェイクアップ | ダイレクト |
| 29 | LPTIM1 ウェイクアップ | ダイレクト |

13.5 EXTI レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1](#) を参照してください。

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスする必要があります。

13.5.1 EXTI 割り込みマスクレジスタ (EXTI_IMR)

アドレスオフセット : 0x00
リセット値 : 0xFF84 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | IM29 | IM28 | Res. | IM26 | IM25 | Res. | IM23 | IM22 | IM21 | IM20 | IM19 | IM18 | IM17 | IM16 |
| | | rw | rw | | rw | rw | | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IM15 | IM14 | IM13 | IM12 | IM11 | IM10 | IM9 | IM8 | IM7 | IM6 | IM5 | IM4 | IM3 | IM2 | IM1 | IM0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **IMx** : ライン x の割り込みマスク (x = 29~28)

- 0 : ライン x からの割り込みリクエストはマスクされます。
- 1 : ライン x からの割り込みリクエストはマスクされません。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:25 **IMx** : ライン x の割り込みマスク (x = 26~25)

- 0 : ライン x からの割り込みリクエストはマスクされます。
- 1 : ライン x からの割り込みリクエストはマスクされません。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23:0 **IMx** : ライン x の割り込みマスク (x = 23~0)

- 0 : ライン x からの割り込みリクエストはマスクされます。
- 1 : ライン x からの割り込みリクエストはマスクされません。

13.5.2 EXTI イベントマスクレジスタ (EXTI_EMR)

アドレスオフセット : 0x04
リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | EM29 | EM28 | Res. | EM26 | EM25 | Res. | EM23 | EM22 | EM21 | EM20 | EM19 | EM18 | EM17 | EM16 |
| | | r/w | r/w | | r/w | r/w | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| EM15 | EM14 | EM13 | EM12 | EM11 | EM10 | EM9 | EM8 | EM7 | EM6 | EM5 | EM4 | EM3 | EM2 | EM1 | EM0 |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:30 予約済みであり、リセット値に保持する必要があります。

ビット 29:28 **EMx** : ライン x のイベントマスク (x = 29~28)

- 0 : ライン x からのイベントリクエストはマスクされます。
- 1 : ライン x からのイベントリクエストはマスクされません。

ビット 27 予約済みであり、リセット値に保持する必要があります。

ビット 26:25 **EMx** : ライン x のイベントマスク (x = 26~25)

- 0 : ライン x からのイベントリクエストはマスクされます。
- 1 : ライン x からのイベントリクエストはマスクされません。

ビット 24 予約済みであり、リセット値に保持する必要があります。

ビット 23:0 **EMx** : ライン x のイベントマスク (x = 23~0)

- 0 : ライン x からのイベントリクエストはマスクされます。
- 1 : ライン x からのイベントリクエストはマスクされません。

13.5.3 EXTI 立ち上がりエッジトリガ選択レジスタ (EXTI_RTSR)

アドレスオフセット : 0x08
リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RT22 | RT21 | RT20 | RT19 | Res. | RT17 | RT16 |
| | | | | | | | | | r/w | r/w | r/w | r/w | | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | RT16 | 2 | 1 | 0 |
| RT15 | RT14 | RT13 | RT12 | RT11 | RT10 | RT9 | RT8 | RT7 | RT6 | RT5 | RT4 | RT3 | RT2 | RT1 | RT0 |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **RTx** : ライン x の立ち上がりトリガイイベント設定ビット (x = 22~19)

- 0 : 入力ライン x の (イベントと割り込みについて) 立ち上がりトリガは無効です。
- 1 : 入力ライン x の (イベントと割り込みについて) 立ち上がりトリガは有効です。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **RTx** : ライン x の立ち上がりトリガイイベント設定ビット (x = 17~0)

- 0 : 入力ライン x の (イベントと割り込みについて) 立ち上がりトリガは無効です。
- 1 : 入力ライン x の (イベントと割り込みについて) 立ち上がりトリガは有効です。



注： 設定可能なウェイクアップラインはエッジトリガであるため、これらのライン上でグリッチが生成されないようにする必要があります。

EXTI_RTISR レジスタへの書き込み中に設定可能な割り込みラインで立ち上がりエッジが発生した場合、ペンディングビットはセットされません。

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この設定では、両方のエッジでトリガ条件が生成されます。

13.5.4 立ち下がりエッジトリガ選択レジスタ（EXTI_FTSR）

アドレスオフセット：0x0C
リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FT22 | FT21 | FT20 | FT19 | Res. | FT17 | FT16 |
| | | | | | | | | | rw | rw | rw | rw | | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| FT15 | FT14 | FT13 | FT12 | FT11 | FT10 | FT9 | FT8 | FT7 | FT6 | FT5 | FT4 | FT3 | FT2 | FT1 | FT0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 FTx：ライン x の立ち下がりトリガイベント設定ビット（x = 22~19）
0：入力ライン x の（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ライン x の（イベントと割り込みについて）立ち下がりトリガは有効です

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 FTx：ライン x の立ち下がりトリガイベント設定ビット（x = 17~0）
0：入力ライン x の（イベントと割り込みについて）立ち下がりトリガは無効です
1：入力ライン x の（イベントと割り込みについて）立ち下がりトリガは有効です

注： 設定可能なウェイクアップラインはエッジトリガであるため、これらのライン上でグリッチが生成されないようにする必要があります。

EXTI_RTISR レジスタへの書き込み中に設定可能な割り込みラインで立ち下がりエッジが発生した場合、ペンディングビットはセットされません。

同じ割り込みラインに対して、立ち上がりエッジトリガと立ち下がりエッジトリガを設定することができます。この設定では、両方のエッジでトリガ条件が生成されます。

13.5.5 EXTI ソフトウェア割り込みイベントレジスタ (EXTI_SWIER)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|------|------|------|-------|-------|-------|-------|------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SWI22 | SWI21 | SWI20 | SWI19 | Res. | SWI17 | SWI16 |
| | | | | | | | | | rw | rw | rw | rw | | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SWI15 | SWI14 | SWI13 | SWI12 | SWI11 | SWI10 | SWI9 | SWI8 | SWI7 | SWI6 | SWI5 | SWI4 | SWI3 | SWI2 | SWI1 | SWI0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **SWIx** : ライン x のソフトウェア割り込み (x = 22~19)

このビットが“0”のときに“1”をセットすると、EXTI_PR レジスタに対応するペンディングビットがセットされます。EXTI_IMR および EXTI_EMR のこのラインで割り込みが有効になっていれば、割り込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする (対応ビットに“1”を書き込む) と、このビットはクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **SWIx** : ライン x のソフトウェア割り込み (x = 17~0)

このビットが“0”のときに“1”をセットすると、EXTI_PR レジスタに対応するペンディングビットがセットされます。EXTI_IMR および EXTI_EMR のこのラインで割り込みが有効になっていれば、割り込みリクエストが生成されます。

EXTI_PR レジスタの対応するビットをクリアする (対応ビットに“1”を書き込む) と、このビットはクリアされます。

13.5.6 EXTI ペンディングレジスタ (EXTI_PR)

アドレスオフセット : 0x14

リセット値 : 定義されていません。

| | | | | | | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PIF22 | PIF21 | PIF20 | PIF19 | Res. | PIF17 | PIF16 |
| | | | | | | | | | rc_w1 | rc_w1 | rc_w1 | rc_w1 | | rc_w1 | rc_w1 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PIF15 | PIF14 | PIF13 | PIF12 | PIF11 | PIF10 | PIF9 | PIF8 | PIF7 | PIF6 | PIF5 | PIF4 | PIF3 | PIF2 | PIF1 | PIF0 |
| rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rc_w1 |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:19 **PIF_x** : ライン x のペンディング割り込みフラグ (x = 22~19)

0 : トリガリクエストは発生していません。

1 : 選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが割り込みラインに到着したときにセットされます。このビットに“1”を書き込むか、またはエッジ検出回路の感度を変更すると、このビットはクリアされます。

ビット 18 予約済みであり、リセット値に保持する必要があります。

ビット 17:0 **PIF_x** : ライン x のペンディング割り込みフラグ (x = 17~0)

0 : トリガリクエストは発生していません。

1 : 選択されたトリガリクエストが発生しました。

このビットは、選択されたエッジイベントが割り込みラインに到着したときにセットされます。このビットに“1”を書き込むか、またはエッジ検出回路の感度を変更すると、このビットはクリアされます。

13.5.7 EXTI レジスタマップ

次の表に、EXTI レジスタマップとリセット値を示します。

表 47. 拡張割り込み／イベントコントローラのレジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|------------|------|------|-----------|------|------|-----------|------|------|----------|----------------|----|----|------|-----------|----|----|----|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0x00 | EXTI_IMR | Res. | Res. | IM[29:28] | | Res. | IM[26:25] | | Res. | IM[23:0] | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | 1 | 1 | | 1 | 1 | | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | EXTI_EMR | Res. | Res. | EM[29:28] | | Res. | EM[26:25] | | Res. | EM[23:0] | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | EXTI_RTISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RT[22:19] | | | Res. | RT[17:0] | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | EXTI_FTSR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FT[22:19] | | | Res. | FT[17:0] | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x10 | EXTI_SWIER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SWI [22:19] | | | Res. | SWI[17:0] | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x14 | EXTI_PR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PIF [22:19] | | | Res. | PIF[17:0] | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

14 アナログデジタルコンバータ (ADC)

14.1 概要

この 12 ビット ADC は、逐次比較型アナログデジタルコンバータです。最大 19 の多重化チャネルを持ち、16 の外部および 3 内部ソースからの信号を測定することができます。さまざまなチャネルの A/D 変換は、シングル、連続、スキャン、または不連続モードで行うことができます。ADC の結果は、左詰めまたは右詰め 16 ビットのデータレジスタに格納されます。

アナログウォッチドッグ機能により、入力電圧が、ユーザ定義の上限値または下限値から逸脱していないかを、アプリケーションで検出することができます。

低周波数で非常に低い消費電力を可能にするために、効率的な低電力モードが実装されています。

組み込みのオーバーサンプリング回路により、CPU の計算負荷を軽減しながらアナログ性能を高めることができます。

14.2 ADC の主な機能

- 高性能
 - 12 ビット、10 ビット、8 ビット、または 6 ビットの設定可能な分解能
 - ADC 変換時間：12 ビット分解能（1.14 MHz）で 0.87 μ s、10 ビット分解能で 0.81 μ s。分解能を下げることによって、より高速な変換時間を達成可能。
 - 自動較正
 - プログラム可能なサンプリング時間
 - 組み込みのデーター貫性によるデータ配置
 - DMA サポート
- 低電力
 - アプリケーションは最適な ADC 性能を維持しながら、低電力動作の PCLK 周波数を低減できます。たとえば、PCLK の周波数にかかわらず、1.0 μ s の変換時間が維持されます。
 - ウェイトモード：低周波数 PCLK のアプリケーションの ADC オーバーランを防止します。
 - オートオフモード：アクティブ変換フェーズ中以外は、ADC の電源は自動的に切れます。これは ADC の消費電力を大幅に低減します。
- アナログ入力チャンネル
 - 16 の外部アナログ入力
 - 内部温度センサ用の 1 つのチャンネル (V_{SENSE})
 - 内部基準電圧用の 1 つのチャンネル (V_{REFINT})
 - 外部 V_{LCD} 電源ピンを監視するための 1 つのチャンネル
- 変換開始は、次のように開始できます。
 - ソフトウェアによって
 - 設定可能な極性によるハードウェアトリガによって (TIM2、TIM6、TIM21、TIM22、または GPIO 入力イベントからの内部タイマイイベント)
- 変換モード
 - 単一チャンネルを変換でき、または一連のチャンネルをスキャンできます。
 - シングルモードは、選択された入力をトリガごとに 1 回変換します。
 - 連続モードは、選択された入力を連続的に変換します。
 - 不連続モード
- サンプリング終了時、変換終了時、シーケンス変換終了時、およびアナログウォッチドッグイベントまたはオーバーランイベント時に割り込みを生成します。
- アナログウォッチドッグ
- オーバーサンプリング回路
 - 16 ビットデーターレジスタ
 - 2 ~ 256 倍までオーバーサンプリング比を調整可能
 - 最大 8 ビットまでプログラム可能なデータシフト
- ADC 電源仕様：1.8 ~ 3.6 V
- ADC 入力電圧範囲： $V_{SSA} \leq V_{IN} \leq V_{DDA}$

 32 に、ADC のブロック図を示します。

14.3 ADC ピンと内部信号

表 48. ADC 内部信号

| 内部信号名 | 信号タイプ | 説明 |
|---------------------|-------|--|
| TRGx | 入力 | ADC 変換トリガ |
| V _{SENSE} | 入力 | 内部温度センサ出力電圧 |
| V _{REFINT} | 入力 | 内部電圧基準出力電圧 |
| LCD_VLCD1 | 入力 | LCD が 1/3Bias に設定されたとき、1/3 VLCD LCD が 1/4Bias または 1/2Bias に設定されたとき、1/4 VLCD |

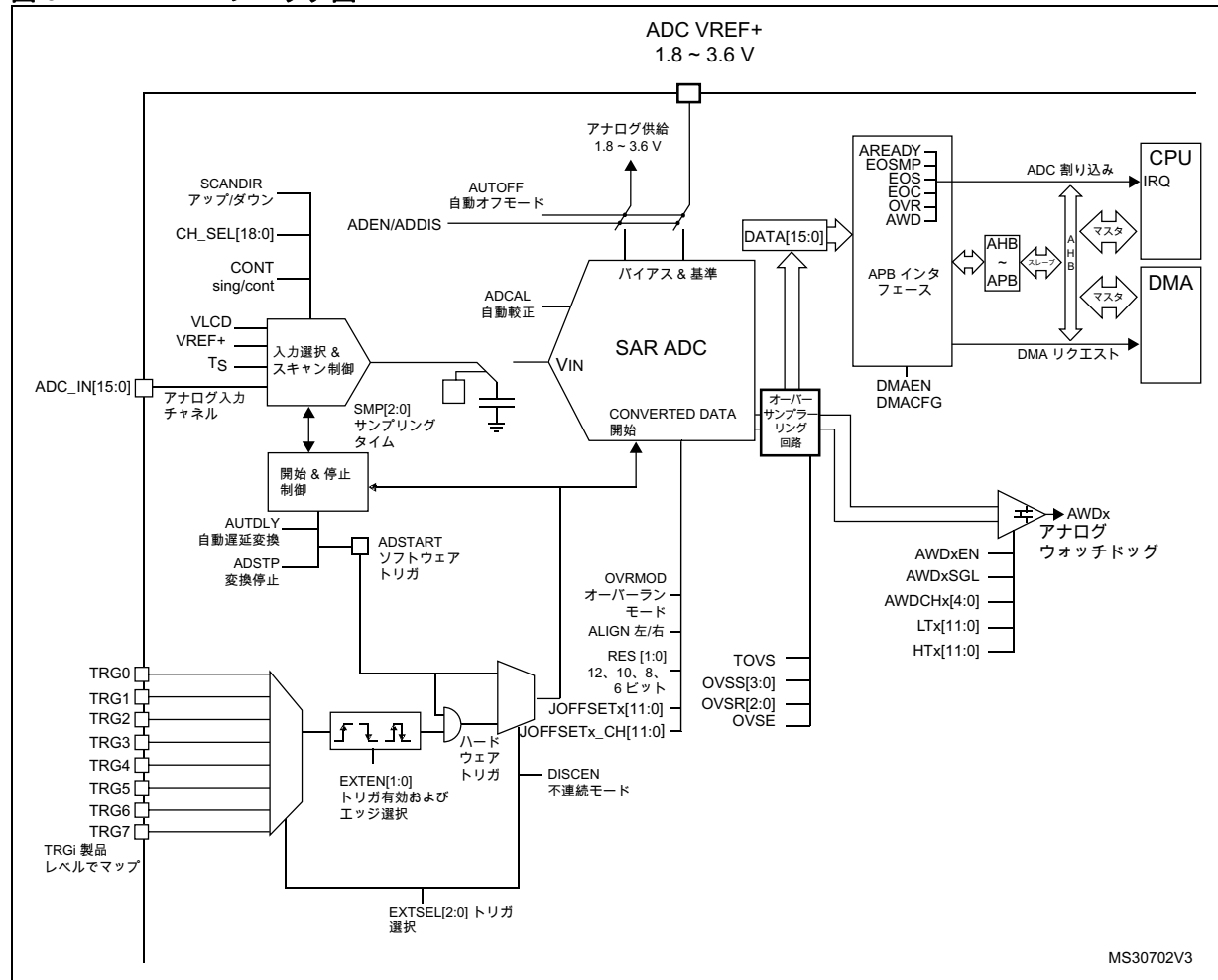
表 49. ADC ピン

| 名前 | 信号タイプ | 説明 |
|------------------|----------------|--|
| V _{DDA} | 入力、アナログ電源 | ADC、V _{DDA} ≥ V _{DD} のアナログ電源と正基準電圧 |
| V _{SSA} | 入力、アナログ供給グラウンド | アナログ電源のグラウンド。V _{SS} 電位である必要がある |
| ADC_IN[15:0] | アナログ入力信号 | 16 チャンネルのアナログ入力 |

14.4 ADC の機能詳細

図 32 に ADC ブロック図を、表 49 に ADC ピンの説明を示します。

図 32. ADC ブロック図



1. TRGi マッピングについては、表 52 : 外部トリガを参照してください。

14.4.1 ADC 電圧レギュレータ (ADVREGEN)

ADC には特定の内部電圧レギュレータがあり、ADC を使用する前に有効にして、安定させる必要があります。

ADC 電圧レギュレータの安定時間は、完全にハードウェアによって管理されるため、ソフトウェアで対処する必要はありません。

ADC の動作が完了すると、ADC は無効化されます (ADEN=0)。その後、ADC 電圧レギュレータを無効にすることによって、さらに節電できます (ADC 電圧レギュレータの無効化シーケンスを参照してください)。

注： 内部電圧レギュレータが無効なときには、内部アナログ較正が保持されます。

ADC 内部電圧レギュレータのアナログ基準

内部 ADC 電圧レギュレータは、 V_{REFINT} 内部電圧基準のバッファされたコピーを使用します。このバッファは、メイン電圧レギュレータが通常の RUN モード（デバイスが RUN モードまたは SLEEP モードで動作する MR モード）のときには、常に有効です。メイン電圧レギュレータが低電力実行モード（デバイスが LOW-POWER RUN、LOW-POWER SLEEP、または STOP モードで動作する LPR モード）のときには、このバッファが無効になることがあり、ソフトウェアは以下に説明されている手順に従って ADC を使用する必要があります。

1. 低電力実行モードに入ります（ADC 電圧レギュレータと内部 ADC 電圧レギュレータの両方とも無効である必要があります）。
2. V_{REFINT} 内部電圧基準を有効にするために、REF_CTRL レジスタの EN_VREFINT ビットをセットします。電圧基準の準備ができるまで、VREFINT_RDYF ビットをポーリングする必要があります。
3. REF_CTRL レジスタの ENBUF_EN_VREFINT_ADC ビットをセットすることによって、バッファを有効にします。バッファの準備ができるまで、VREFINT_ADC_RDYF ビットをポーリングする必要があります。バッファの消費電力は、約 8 μA です。
4. ADVREGEN ビットをセットすることによって、内部 ADC 電圧レギュレータを有効にします。これで、ADC の使用準備ができます。

ADVREG 有効化シーケンス

電圧レギュレータを有効にするには、3 つの方法があります。

- ADVREGEN=1 を書き込むことによって
- ADCAL=1 を書き込んで較正を起動することによって（ADVREGEN ビットは自動的に 1 にセットされます）
- ADEN=1 を書き込んで ADC を有効にすることによって

ADVREG 無効化シーケンス

ADC 電圧レギュレータを無効にするには、以下のシーケンスを実行します。

1. ADC が無効であることを確認します（ADEN=0）。
2. ADVREGEN=0 を書き込みます。

14.4.2 較正 (ADCAL)

ADC は較正機能を備えています。この手順時、ADC は、次の ADC の電源オフまで ADC に内部で適用される較正係数を計算します。アプリケーションは、較正中は ADC を使用してはならず、較正が完了するまで待つ必要があります。

較正は、A/D 変換を開始する前に行ってください。較正は、プロセスのばらつきによりチップごとに異なることがあるオフセットエラーを削除します。

較正は、ビット ADCAL=1 をセットすることによってソフトウェアによって開始されます。較正は、ADC が無効のとき（ADEN=0 のとき）だけ開始できます。すべての較正シーケンス時、ADCAL ビットは 1 のままです。較正が完了すると、ハードウェアによってクリアされます。この後、較正係数を ADC_DR レジスタ（ビット 6 ~ 0）から読み出すことができます。

ADC が無効の場合（ADEN=0）または ADC 電圧基準が無効の場合（ADVREGEN = 0）、内部アナログ較正が保持されます。ADC 動作条件が変化したときには（ V_{DDA} の変化が ADC オフセットのばらつきの主因であり、温度変化はそれほど影響しません）、較正サイクルを再実行することが推奨されます。

以下の場合、較正係数は失われます。

- 製品が STANDBY モードのとき (ADC に電源が供給されていないとき)。
- ADC ペリフェラルがリセットされたとき。

以下の低電力モードでは、較正係数は保持されます。LOW-POWER RUN、LOW-POWER SLEEP、および STOP。

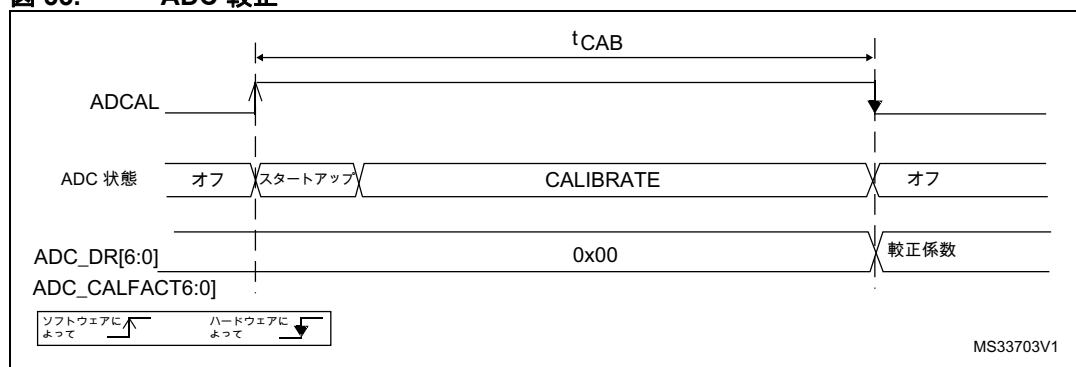
ADC を再起動するときの時間を短縮するために、ソフトウェアによって較正係数を保存して復元することも可能です (ADC パワーダウン中の温度と電圧が安定している限り)。

ADC が有効であり、変換中でない場合 (ADEN=1 かつ ADSTART=0)、較正係数を書き込むことができます。その場合、次の変換開始時に、較正係数がアナログ ADC に自動的にインジェクトされます。このローディングは透過性であり、変換開始のサイクル遅延は増加しません。

較正ソフトウェア手順：

1. ADEN=0 であることを確認します。
2. ADCAL=1 にセットします。
3. ADCAL=0 (または EOCAL=1) になるまで待ちます。これは、ADC_IER レジスタの EOCALIE ビットをセットすることによって割り込みを有効にした場合、割り込みによって処理できます。
4. 較正係数を ADC_DR または ADC_CALFACT レジスタのビット 6:0 から読み出すことができます。

図 33. ADC 較正



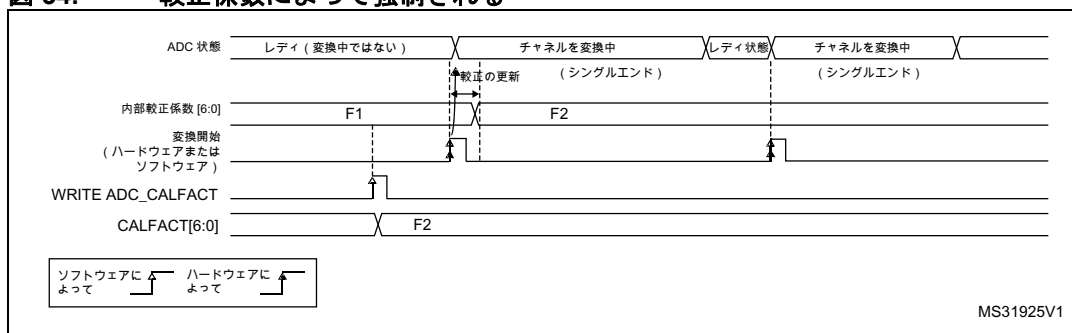
ADC 電圧レギュレータが以前にセットされていなかった場合は、ADCAL=1 にセットすると、自動的に有効になります (ビット ADVREGEN はハードウェアによって自動的にセットされます)。この場合、ADC 較正時間は、ADC 電圧レギュレータの安定時間を考慮して、長めになります。

較正の終了時、ADC 電圧レギュレータは有効のままです。

較正係数によって強制されるソフトウェア手順

1. ADEN= 1 かつ ADSTART =0 である (ADC が起動していて、変換中でない) ことを確認します。
2. 保存済みの較正係数とともに ADC_CALFACT を書き込みます。
3. 較正係数は、新しい変換が開始されるときに使用されます。

図 34. 較正係数によって強制される



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

MCU のパワーアップ時、ADC は無効になり、パワーダウンモードになります (ADEN=0)。

図 35 に示すように、ADC は正確な変換を開始する前に、安定時間 t_{STAB} を必要とします。

ADC を有効化または無効化するには、2 つの制御ビットが使用されます。

- ADC を有効にするには、ADEN=1 をセットします。ADC の動作準備ができると、ADRDY フラグがセットされます。
- ADC を無効にして、ADC をパワーダウンモードにするには、ADDIS=1 をセットします。ADC が完全に無効になると、ADEN および ADDIS ビットはハードウェアによって自動的にクリアされます。

ADC 電圧レギュレータが以前にセットされていなかった場合は、ADEN=1 をセットすると、自動的に有効になります (ビット ADVREGEN はハードウェアによって自動的にセットされます)。この場合、ADC 安定時間 t_{STAB} は、ADC 電圧レギュレータの安定時間を考慮して、長めになります。

その後、SWSTART=1 をセットすることによって (セクション 14.5 : 外部トリガおよびトリガ極性での変換 (EXTSEL、EXTEN) (277 ページ) を参照)、またはトリガが有効な場合は外部トリガイベントが発生したときに、変換を開始できます。

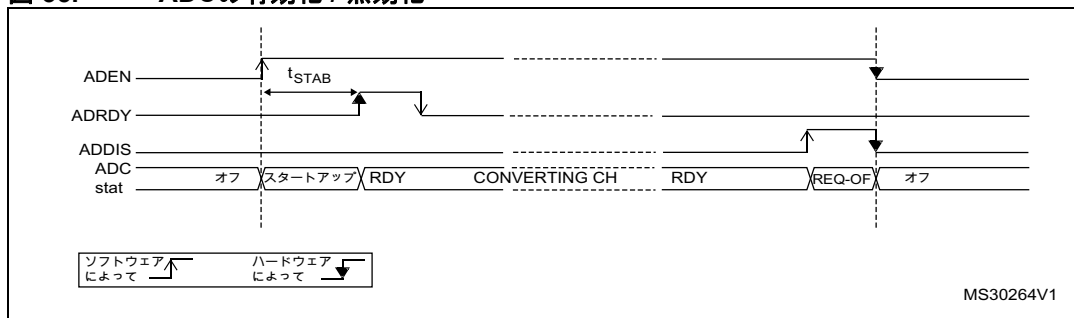
ADC を有効にするには、次の手順を実行します。

1. ADC_CR レジスタの ADEN=1 をセットします。
2. ADC_ISR レジスタの ADRDY=1 になるまで待ちます (ADRDY は ADC 起動時間後にセットされます)。これは、ADC_IER レジスタの ADRDYIE ビットをセットすることによって割り込みを有効にした場合、割り込みによって処理できます。

ADC を無効にするには、次の手順を実行します。

1. ADC_CR レジスタの ADSTART=0 を確認して、変換が実行中でないことを確認します。必要な場合は、ADC_CR レジスタの ADSTP ビットに 1 を書き込み、このビットが 0 として読み出されるまで待つことによって、実行中の変換を停止します。
2. ADC_CR レジスタの ADDIS=1 をセットします。
3. アプリケーションによって必要とされる場合、ADC_CR レジスタの ADEN=0 になり、ADC が完全に無効であることを示すまで待ちます (ADEN=0 になると、ADDIS は自動的にリセットされます)。

図 35. ADCの有効化 / 無効化

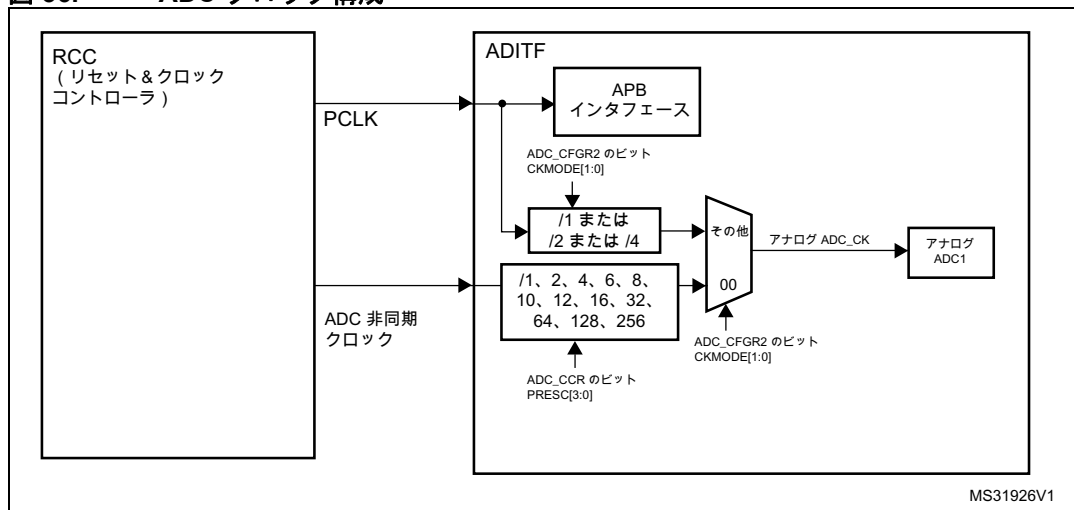


注：オートオフモード (AUTOFF=1) の場合、パワーオン/オフフェーズはハードウェアによって自動的に実行され、ADRDY フラグはセットされません。

14.4.4 ADC クロック (CKMODE、PRESC[3:0]、LFMEN)

ADC はデュアルクロックドメインアーキテクチャを持つので、ADC に APB クロック (PCLK) から独立したクロック (ADC 非同期クロック) を与えることができます。

図 36. ADC クロック構成



1. PCLK および ADC 非同期クロックを有効にする方法については、[セクション 7：リセットおよびクロック制御 \(RCC\) \(143 ページ\)](#) を参照してください。

アナログ ADC の入力クロックは、2 つのクロックソースから選択できます (PCLK および ADC 非同期クロックを有効にする方法については、[図 36 : ADC クロック構成](#)を参照してください)。

- a) ADC クロックは、APB クロックから独立し、非同期の「ADC 非同期クロック」という名前の特定のクロックソースにできます。
- このクロックソースの生成の詳細については、RCC のセクションを参照してください。
- この構成を選択するには、ADC_CFGR2 レジスタのビット CKMODE[1:0] をリセットする必要があります。
- b) ADC クロックは ADC バスインタフェースの APB クロックから取得して、ビット CKMODE[1:0] に従ってプログラム可能な係数 (2 または 4) で分周することができます。
- この構成を選択するには、ADC_CFGR2 レジスタのビット CKMODE[1:0] が "00" 以外の値である必要があります。

オプション a) では、ADC_CFGR2 レジスタのビット PRESC[3:0] をプログラムするとき、生成された ADC クロックを最終的にプリスケラ (1、2、4、6、8、12、16、32、64、128、256) で分周できます。

オプション a) には、選択された APB クロック構成にかかわらず、最大の ADC クロック周波数に達するという利点があります。

オプション b) には、クロックドメインの再同期を迂回するという利点があります。これは、ADC がタイマによってトリガされるときと、アプリケーションが ADC の確実で精密なトリガを必要とする場合に便利です (そうしないと、トリガインスタンスの不確実性は、2 つのクロックドメイン間の再同期によって高まります)。

表 50. トリガから変換開始までの遅延

| ADC クロックソース | CKMODE[1:0] | トリガイベントから変換開始までの遅延 |
|------------------|-------------|---|
| HSI16 MHz クロック | 00 | 遅延は決定的ではありません (ジッタ)。 |
| PCLK は 2 分周されます。 | 01 | 遅延は決定的であり (ジッタではない)、4.25 ADC クロックサイクルに等しい。 |
| PCLK は 4 分周されます。 | 10 | 遅延は決定的であり (ジッタではない)、4.125 ADC クロックサイクルに等しい。 |
| PCLK は 1 分周されます。 | 11 | 遅延は決定的であり (ジッタではない)、4.5 ADC クロックサイクルに等しい。 |

注意 : CKMODE[1:0]=11 (PCLK は 1 分周されます) を選択するときには、ユーザは PCLK が 50% のデューティサイクルを持つことを確認する必要があります。このためには、RCC 内部で、ユーザは 50% デューティサイクルを持つシステムクロックを選択する必要があり、RCC 内部の APB プリスケラをバイパスモードで設定する必要があります (RCC セクションを参照してください)。

低周波数

2.8MHz 未満のアナログ ADC クロック周波数を選択するときには、まず、ADC_CCR レジスタのビット LFMEN=1 をセットすることによって低周波数モードを有効にする必要があります。

14.4.5 ADC の設定

ADC が無効な場合、ソフトウェアは ADC_CR レジスタの ADCAL および ADEN ビットに書き込む必要があります (ADEN は 0 である必要があります)。

ADC が有効であり、ADC を無効にするための保留中のリクエストがない場合のみ (ADEN=1 かつ ADDIS=0)、ソフトウェアは ADC_CR レジスタの ADSTART および ADDIS ビットに書き込む必要があります。

ADC_IER、ADC_CFGRi、ADC_SMPR、ADC_TR、ADC_CHSELR、および ADC_CCR レジスタのその他のすべての制御ビットについては、ADC が有効な場合 (ADEN = 1) と変換中でない場合 (ADSTART = 0) のみ、ソフトウェアは設定制御ビットに書き込む必要があります。

ADC が有効であり (おそらく変換中)、ADC を無効にするための保留中のリクエストがない場合のみ (ADSTART=1 かつ ADDIS=0)、ソフトウェアは ADC_CR レジスタの ADSTP ビットに書き込む必要があります。

注： *上記のルールによって禁じられた書き込み操作をソフトウェアが行わないようにするハードウェア保護はありません。そのような禁止された書き込みアクセスが発生した場合、ADC は未定義の状態になることがあります。この場合に正しい動作を回復するには、ADC を無効にする必要があります (ADEN=0 と ADC_CR レジスタのすべてのビットをクリアします)。*

14.4.6 チャネル選択 (CHSEL、SCANDIR)

最大 19 の多重化チャネルがあります。

- GPIO ピンからの 16 のアナログ入力 (ADC_IN0...ADC_IN15)
- 3 つの内部アナログ入力 (温度センサ、内部基準電圧、LCD_VLCD1 チャネル)

単一チャネルを変換でき、または一連のチャネルを自動的にスキャンできます。

変換されるチャネルのシーケンスは、ADC_CHSELR チャネル選択レジスタでプログラムする必要があります。各アナログ入力チャネルに専用の選択ビットがあります (CHSEL0...CHSEL18)。

チャネルがスキャンされる順序は、ADC_CFGR1 レジスタの SCANDIR ビットをプログラムすることによって設定できます。

- SCANDIR=0：チャネル 0 からチャネル 18 へ前方スキャン
- SCANDIR=1：チャネル 18 からチャネル 0 へ後方スキャン

温度センサ、V_{REFINT}、および LCD_VLCD1 内部チャネル

温度センサは、チャネル ADC_IN18 に接続されます。内部電圧基準 V_{REFINT} は、チャネル ADC_IN17 に接続されます。LCD_VLCD1 チャネルは、チャネル ADC_IN16 に接続されます。

14.4.7 プログラム可能なサンプリング時間 (SMP)

変換を開始する前に、ADC は測定する電圧ソースと ADC の内蔵サンプリングコンデンサの間の直接接続を確立する必要があります。このサンプリング時間は、入力電圧ソースがサンプルをチャージし、コンデンサが入力電圧レベルを保持できるだけの十分な長さが必要です。

プログラム可能なサンプリング時間によって、入力電圧ソースの入力抵抗に従って変換速度を微調整することができます。

ADC は、一定の ADC クロックサイクル数だけ入力電圧をサンプリングしますが、この時間は、ADC_SMPR レジスタの SMP[2:0] ビットを使用して変更できます。

このプログラム可能なサンプリング時間は、すべてのチャネルに共通です。アプリケーションによって必要とされる場合、ソフトウェアは変換ごとにこのサンプリング時間を変更して適応できます。



合計変換時間は、次のように計算されます。

$$t_{\text{CONV}} = \text{サンプリング時間} + 12.5 \times \text{ADC クロックサイクル}$$

例：

ADC_CLK = 16 MHz、サンプリング時間 = 1.5 ADC クロックサイクル：

$$t_{\text{CONV}} = 1.5 + 12.5 = 14 \text{ ADC クロックサイクル} = 0.875 \mu\text{s}$$

ADC は、EOSMP フラグをセットすることによって、サンプリングフェーズの終了を示します。

14.4.8 シングル変換モード (CONT=0)

シングル変換モードでは、ADC は単一シーケンスの変換を実行して、すべてのチャンネルを一度変換します。このモードは ADC_CFGR1 レジスタの CONT=0 のときに選択されます。変換は、次のいずれかによって開始されます。

- ADC_CR レジスタの ADSTART ビットのセット
- ハードウェアトリガイベント

シーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC (end of conversion) フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

変換シーケンスの完了後、

- EOSEQ (end of sequence) フラグがセットされます。
- EOSEQIE ビットがセットされている場合、割り込みが生成されます。

次に、ADC は、新しい外部トリガイベントが発生するか、ADSTART ビットが再びセットされるまで停止します。

注： 単一チャンネルを変換するには、長さが1のシーケンスをプログラムします。

14.4.9 連続変換モード (CONT=1)

連続変換モードでは、ソフトウェアまたはハードウェアトリガイベントが発生すると、ADC は一連の変換を実行して、すべてのチャンネルを一度変換した後、自動的に再起動して、同じ変換シーケンスを連続的に実行します。このモードは ADC_CFGR1 レジスタの CONT=1 のときに選択されます。変換は、次のいずれかによって開始されます。

- ADC_CR レジスタの ADSTART ビットのセット
- ハードウェアトリガイベント

シーケンス内で、各変換の完了後、

- 変換されたデータは 16 ビットの ADC_DR レジスタに格納されます。
- EOC (end of conversion) フラグがセットされます。
- EOCIE ビットがセットされている場合、割り込みが生成されます。

変換シーケンスの完了後、

- EOSEQ (end of sequence) フラグがセットされます。
- EOSEQIE ビットがセットされている場合、割り込みが生成されます。

次に、新しいシーケンスをすぐに再開して、ADC は変換シーケンスを連続的に繰り返します。

注: 単一チャネルを変換するには、長さが1のシーケンスをプログラムします。
不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1とCONT=1の両方のビットをセットすることは禁じられています。

14.4.10 変換の開始 (ADSTART)

ソフトウェアは、ADSTART=1をセットすることによってADC変換を開始します。

ADSTARTがセットされると、変換は、

- EXTEN = 0x0 (ソフトウェアトリガ) の場合、すぐに開始します。
- EXTEN¹ 0x0 の場合、選択されたハードウェアトリガの次のアクティブエッジで開始します。

ADSTART ビットは、ADC の動作が実行中かどうかを示すためにも使用されます。ADSTART=0 であり、ADC がアイドルであることを示しているときには、ADC を再設定できます。

ADSTART ビットは、ハードウェアによってクリアされます。

- ソフトウェアトリガによるシングルモードのとき (CONT=0、EXTSEL=0x0)
 - 連続シーケンスの終了時に (EOSEQ=1)
- すべての場合に (CONT=x、EXTSEL=x)
 - ソフトウェアによって起動された ADSTP 手順の実行後 (セクション 14.4.12: 実行中の変換の停止 (ADSTP) (276 ページ) を参照)

注: 連続モード (CONT=1) では、EOSEQ フラグがセットされたとき、シーケンスは自動的に再起動されるので、ADSTART ビットはハードウェアによってクリアされません。

シングルモードでハードウェアトリガが選択されたとき (CONT=0 かつ EXTSEL¹ 0x00)、EOSEQ フラグがセットされたとき、ADSTART はハードウェアによってクリアされません。このため、ハードウェアはADSTART ビットを再びセットする必要がなく、次のトリガイベントを見逃す恐れがありません。

14.4.11 タイミング

変換の開始から変換の終了までの経過時間は、設定されたサンプリング時間に逐次比較時間 (データ分解能に依存) を加えた合計です。

$$t_{ADC} = t_{SMPL} + t_{SAR} = [1.5 I_{min} + 12.5 I_{12bit}] \times t_{ADC_CLK}$$

$$t_{ADC} = t_{SMPL} + t_{SAR} = 93.8 \text{ ns } I_{min} + 781.3 \text{ ns } I_{12bit} = 0.875 \text{ } \mu\text{s } I_{min} \text{ (for } f_{ADC_CLK} = 16 \text{ MHz)}$$

図 37. アナログ / デジタル変換時間

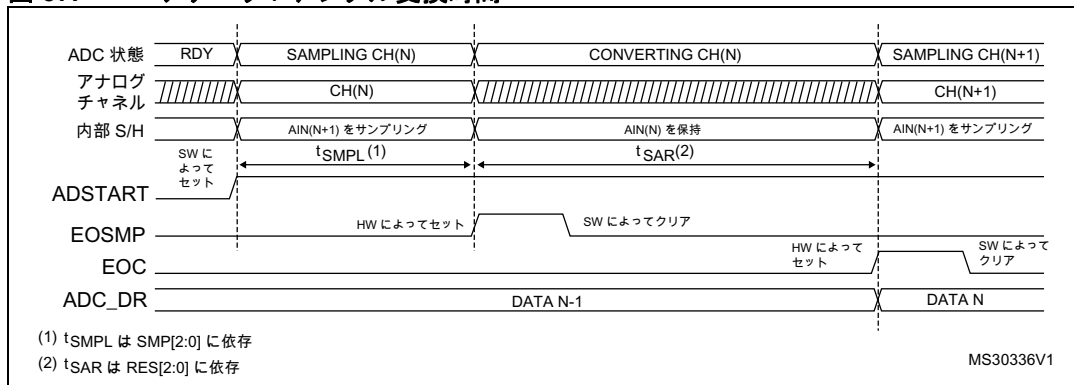
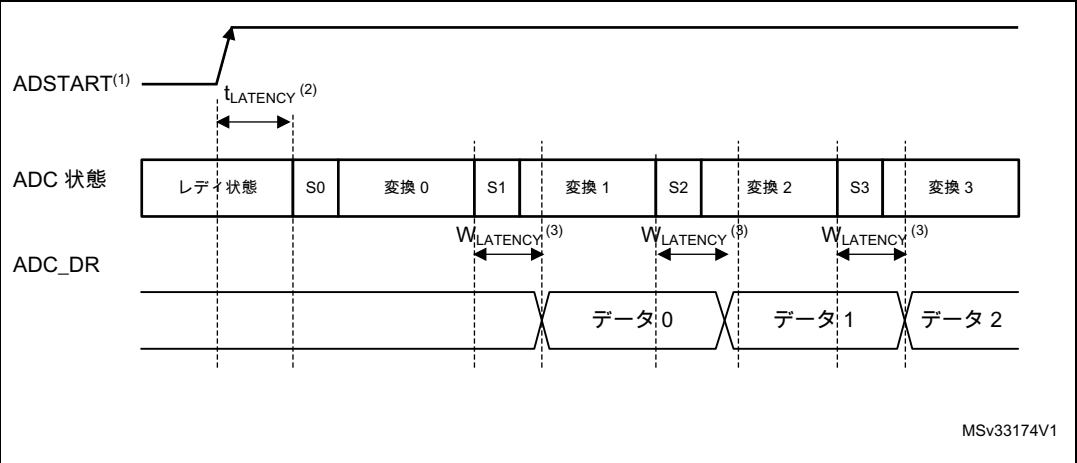


図 38. ADC 変換タイミング



1. EXTEN = 00 または EXTEN ≠ 00
2. トリガ遅延 (詳細についてはデータシートを参照)
3. ADC_DR レジスタ書き込み遅延 (詳細についてはデータシートを参照)

14.4.12 実行中の変換の停止 (ADSTP)

ソフトウェアは、ADC_CR レジスタの ADSTP=1 をセットすることによって、実行中の変換を停止することができます。

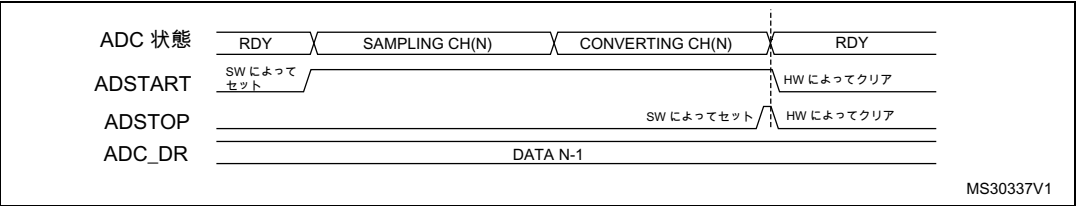
これによって ADC 動作がリセットされ、ADC はアイドルになり、新しい動作の準備ができます。

ADSTP ビットがソフトウェアによってセットされると、実行中の変換は中止され、結果は破棄されます (ADC_DR レジスタは現在の変換で更新されません)。

スキャンシーケンスも中止され、リセットされます (ADC を再起動すると、新しいシーケンスが再開されることを意味します)。

この手順が完了すると、ADSTP および ADSTART ビットの両方がハードウェアによってクリアされ、ソフトウェアは新しい変換を開始する前に ADSTART=0 になるまで待つ必要があります。

図 39. 実行中の変換の停止



14.5 外部トリガおよびトリガ極性での変換 (EXTSEL、EXTEN)

変換または変換シーケンスは、ソフトウェアによって、または外部イベント（タイマキャプチャなど）によってトリガされます。EXTEN[1:0] 制御ビットが “0b00” に等しくない場合、外部イベントは選択された極性で変換をトリガできます。トリガ選択は、ソフトウェアがビット ADSTART=1 をセットすると有効になります。

変換中に発生したハードウェアトリガは無視されます。

ビット ADSTART=0 の場合、発生したハードウェアトリガは無視されます。

表 51 にEXTEN[1:0] の値とトリガ極性の対応を示します。

表 51. トリガ極性の設定

| 転送元 | EXTEN[1:0] |
|-----------------------------|------------|
| トリガ検出は無効です。 | 00 |
| 立ち上がりエッジで検出します。 | 01 |
| 立ち下がりエッジで検出します。 | 10 |
| 立ち上がりエッジと立ち下がりエッジの両方で検出します。 | 11 |

注： 外部トリガの極性は、ADC が変換中でない (ADSTART=0) のときだけ変更できます。

EXTSEL[2:0] 制御ビットは、8 つの可能なイベントのうち、変換をトリガするイベントを選択するために使用されます。

表 52 に、レギュラ変換に使用できる外部トリガを示します。

ソフトウェアソーストリガイベントは、ADC_CR レジスタの ADSTART ビットをセットすることによって生成できます。

表 52. 外部トリガ

| 名前 | 転送元 | EXTSEL[2:0] |
|------|-------------|-------------|
| TRG0 | TIM6_TRGO | 000 |
| TRG1 | TIM21_CH2 | 001 |
| TRG2 | TIM2_TRGO | 010 |
| TRG3 | TIM2_CH4 | 011 |
| TRG4 | TIM22_TRGO | 100 |
| TRG5 | 予約済み | 101 |
| TRG6 | 予約済み | 110 |
| TRG7 | EXTI ライン 11 | 111 |

注： トリガ選択は、ADC が変換中でない (ADSTART=0) のときだけ変更できます。

14.5.1 不連続モード (DISCEN)

このモードは、ADC_CFGR1 レジスタの DISCEN ビットをセットすることによって有効になります。

このモード (DISCEN=1) では、シーケンスで定義された各変換を開始するには、ハードウェアまたはソフトウェアトリガイベントが必要です。逆に、DISCEN=0 の場合は、単一のハードウェアまたはソフトウェアトリガイベントが、シーケンス内で定義されたすべての変換を連続的に開始します。

例：

- DISCEN=1、変換されるチャネル = 0、3、7、10
 - 最初のトリガ：チャネル 0 が変換され、EOC イベントが生成されます。
 - 2 番目のトリガ：チャネル 3 が変換され、EOC イベントが生成されます。
 - 3 番目のトリガ：チャネル 7 が変換され、EOC イベントが生成されます。
 - 4 番目のトリガ：チャネル 10 が変換され、EOC イベントと EOSEQ イベントの両方が生成されます。
 - 5 番目のトリガ：チャネル 0 が変換され、EOC イベントが生成されます。
 - 6 番目のトリガ：チャネル 3 が変換され、EOC イベントが生成されます。
 - ...
- DISCEN=0、変換されるチャネル = 0、3、7、10
 - 最初のトリガ：シーケンス全体、すなわち、チャネル 0、3、7、および 10 が変換されます。各変換後に EOC イベントが生成され、最後の変換後には EOSEQ イベントも生成されます。
 - 後続のトリガイベントがあると、シーケンス全体が再開されます。

注： 不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方のビットをセットすることは禁じられています。

14.5.2 プログラム可能な分解能 (RES) - 高速変換モード

ADC の分解能を下げることによって、高速な変換時間 (t_{SAR}) が可能になります。

分解能は、ADC_CFGR1 レジスタの RES[1:0] ビットをプログラムすることによって、12、10、8、または 6 ビットに設定できます。分解能を下げることによって、高いデータ精度を必要としないアプリケーションの変換時間を高速にできます。

注： RES[1:0] ビットは、ADEN ビットがリセットされたときだけ変更する必要があります。

変換結果は常に 12 ビット幅であり、未使用の LSB ビットはゼロとして読み出されます。

分解能を下げると、表 53 に示すように、逐次比較ステップに必要な変換時間が短縮されます。

表 53. t_{SAR} タイミングは分解能に依存

| RES[1:0] ビット | t_{SAR} (ADC クロック サイクル) | t_{SAR} (ns) ($f_{\text{ADC}} =$ 16 MHz 時) | t_{SMPL} (min) (ADC クロック サイクル) | t_{ADC} (ADC クロック サイクル) (最小 t_{SMPL}) | t_{ADC} (μ s) ($f_{\text{ADC}} =$ 16 MHz 時) |
|-----------------|---|--|---|---|--|
| 12 | 12.5 | 781 ns | 1.5 | 14 | 875 ns |
| 10 | 11.5 | 719 ns | 1.5 | 13 | 812 ns |
| 8 | 9.5 | 594 ns | 1.5 | 11 | 688 ns |
| 6 | 7.5 | 469 ns | 1.5 | 9 | 562 ns |

14.5.3 変換の終了、サンプリングフェーズの終了 (EOC、EOSMP フラグ)

ADC は、各変換の終了 (EOC) イベントを示します。

ADC は、新しい変換データ結果が ADC_DR レジスタで使用可能になるとすぐに、ADC_ISR レジスタの EOC フラグをセットします。ADC_IER レジスタの EOCIE ビットがセットされている場合、割り込みが生成されます。EOC フラグは、ソフトウェアによって 1 を書き込むことによって、または ADC_DR レジスタを読み出すことによってクリアされます。

ADC は、ADC_ISR レジスタの EOSMP フラグをセットすることによって、サンプリングフェーズの終了も示します。EOSMP フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。ADC_IER レジスタの EOSMPIE ビットがセットされている場合、割り込みが生成されます。

この割り込みの目的は、処理を変換と同期するためです。一般に、アナログマルチプレクサには変換フェーズ中の隠された時間でアクセスできるため、マルチプレクサは次のサンプリング開始時に位置づけられます。

注： サンプリングの終了から変換の終了までは非常に短時間しか残されていないため、割り込みと WFI 命令ではなく、ポーリングまたは WFE 命令を使用することが推奨されます。

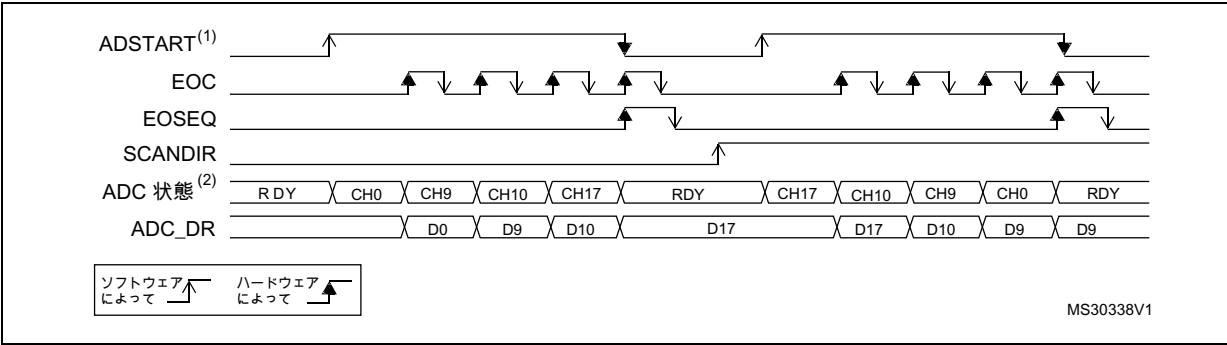
14.5.4 変換シーケンスの終了 (EOSEQ フラグ)

ADC は各シーケンスの終了 (EOSEQ) イベントをアプリケーションに通知します。

ADC は、変換の最後のデータ結果が ADC_DR レジスタで使用可能になるとすぐに、ADC_ISR レジスタの EOSEQ フラグをセットします。ADC_IER レジスタの EOSEQIE ビットがセットされている場合、割り込みが生成されます。EOSEQ フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

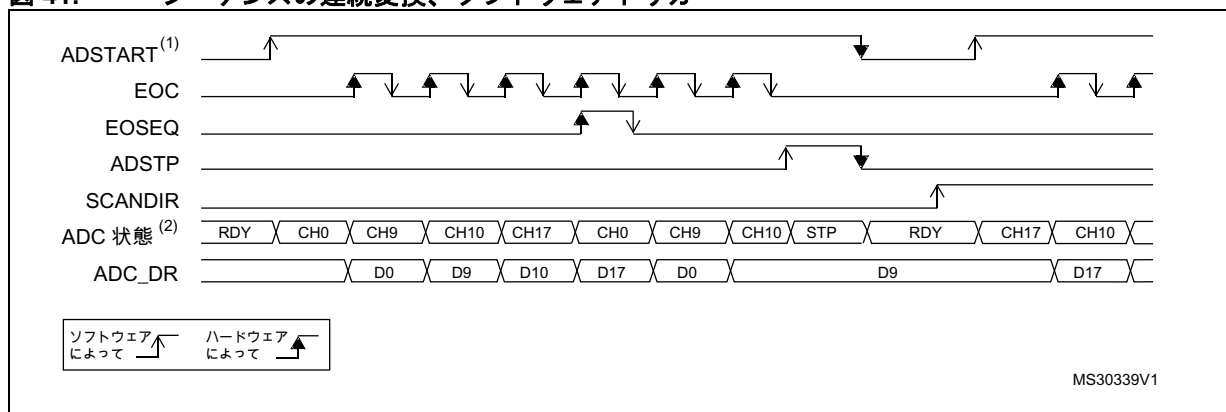
14.5.5 タイミング図の例 (シングル/連続モードハードウェア/ソフトウェアトリガ)

図 40. シーケンスのシングル変換、ソフトウェアトリガ



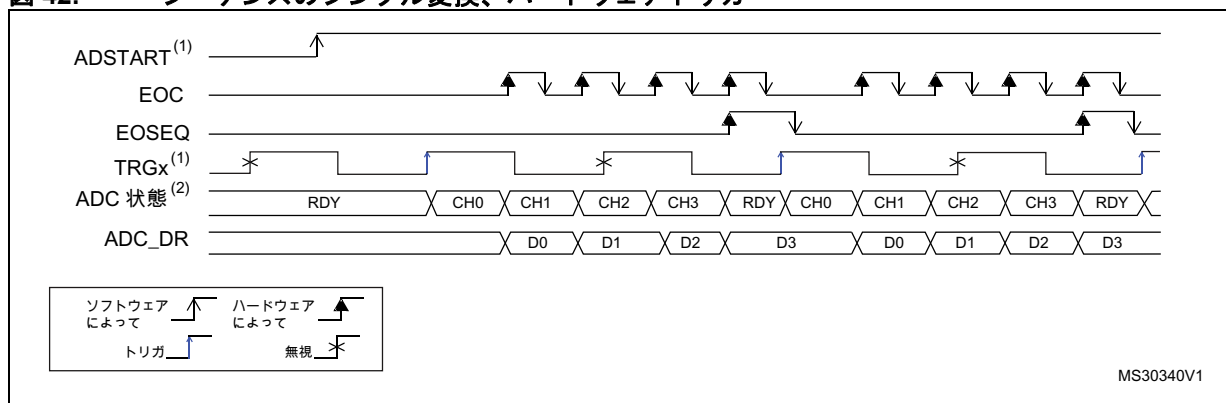
1. EXTEN=0x0, CONT=0
2. CHSEL=0x20601, WAIT=0, AUTOFF=0

図 41. シーケンスの連続変換、ソフトウェアトリガ



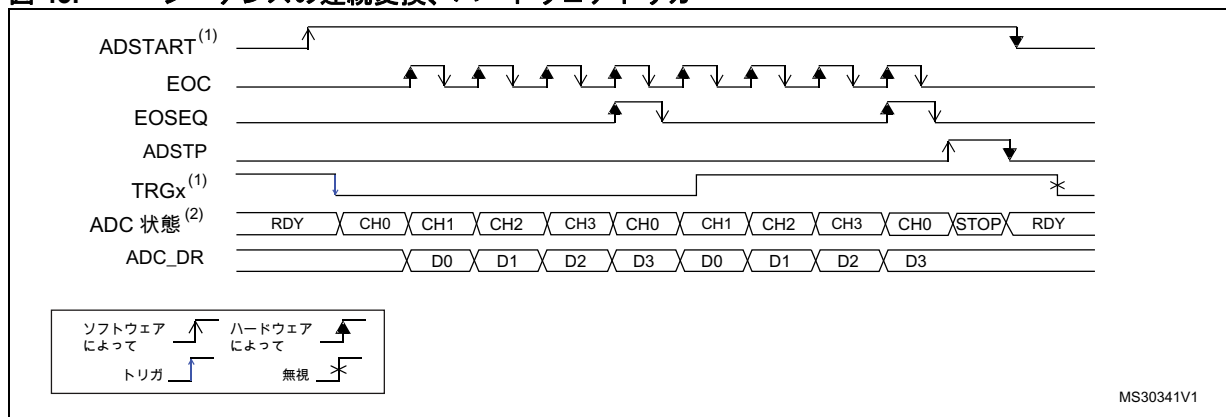
- EXTEN=0x0、CONT=1、
- CHSEL=0x20601、WAIT=0、AUTOFF=0

図 42. シーケンスのシングル変換、ハードウェアトリガ



- EXTSEL=TRGx (オーバー周波数)、EXTEN=0x1 (立ち上がりエッジ)、CONT=0
- CHSEL=0xF、SCANDIR=0、WAIT=0、AUTOFF=0

図 43. シーケンスの連続変換、ハードウェアトリガ



- EXTSEL=TRGx、EXTEN=0x2 (立ち下がりエッジ)、CONT=1
- CHSEL=0xF、SCANDIR=0、WAIT=0、AUTOFF=0

14.6 データ管理

14.6.1 データレジスタおよびデータの配置 (ADC_DR、ALIGN)

各変換の終了時 (EOC イベントの発生時)、変換されたデータの結果は 16 ビット幅の ADC_DR データレジスタに格納されます。

ADC_DR のフォーマットは、設定されたデータ配置と分解能に依存します。

ADC_CFGR1 レジスタの ALIGN ビットは、変換後に格納されるデータの配置を選択します。図 44 に示すように、データは右詰め (ALIGN=0) または左詰め (ALIGN=1) にできます。

図 44. データの配置と分解能 (オーバーサンプリング無効 : OVSE = 0)

| ALIGN | RES | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|-----|----------|----|----|----|----|----|---|---------|---|------|---|----------|-----|---|-----|---|
| 0 | 0x0 | 0x0 | | | | | | | | | | | DR[11:0] | | | | |
| | 0x1 | 0x00 | | | | | | | DR[9:0] | | | | | | | | |
| | 0x2 | 0x00 | | | | | | | DR[7:0] | | | | | | | | |
| | 0x3 | 0x00 | | | | | | | DR[5:0] | | | | | | | | |
| 1 | 0x0 | DR[11:0] | | | | | | | | | | | | 0x0 | | | |
| | 0x1 | DR[9:0] | | | | | | | | | 0x00 | | | | | | |
| | 0x2 | DR[7:0] | | | | | | | 0x00 | | | | | | | | |
| | 0x3 | 0x00 | | | | | | | DR[5:0] | | | | | | | 0x0 | |

MS30342V1

14.6.2 ADC オーバーラン (OVR、OVRMOD)

オーバーランフラグ (OVR) は、新しい変換からのデータが使用可能になる前に、変換されたデータが CPU または DMA によって時間内に読み出されなかったときに、データオーバーランを示します。

OVR フラグは、新しい変換が完了した時点で EOC フラグが '1' のままであった場合に、ADC_ISR レジスタでセットされます。ADC_IER レジスタの OVR1E ビットがセットされている場合、割り込みが生成されます。

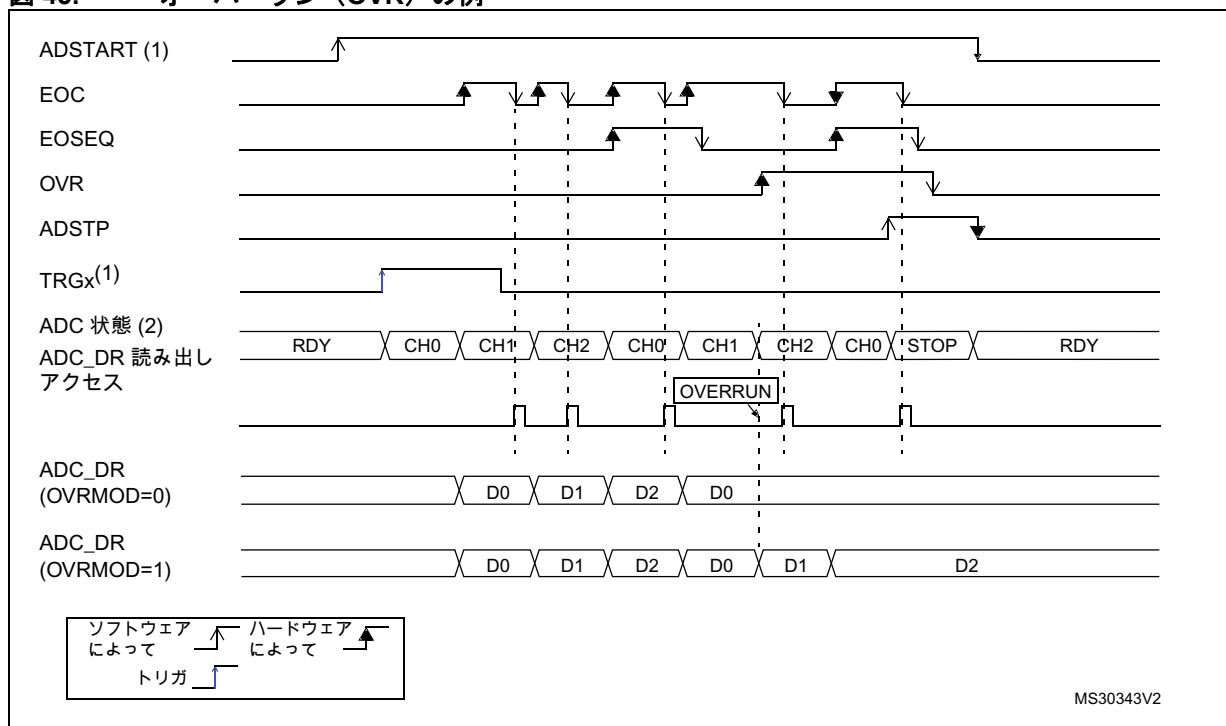
オーバーラン条件が発生すると、ADC は動作し続け、ソフトウェアが ADC_CR レジスタの ADSTP ビットをセットすることによってシーケンスの停止とリセットを決めるまで、変換を続行できます。

OVR フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

ADC_CFGR1 レジスタの OVRMOD ビットをプログラムすることによって、オーバーランイベントが発生したときにデータが保存されるか上書きされるかを設定できます。

- OVRMOD=0
 - － オーバーランイベントが発生しても、データレジスタは上書きされません。古いデータは保持され、新しい変換は破棄されます。OVR が 1 のままの場合、さらに変換を実行できませんが、結果データは破棄されます。
- OVRMOD=1
 - － データレジスタは最後の変換結果で上書きされ、以前の未読データは失われます。OVR が 1 のままの場合、さらに変換を実行でき、ADC_DR レジスタは常に最新の変換からのデータを含みます。

図 45. オーバーラン (OVR) の例



MS30343V2

14.6.3 DMA を使用しない変換データシーケンスの管理

変換が十分に遅い場合、ソフトウェアで変換シーケンスを処理することができます。この場合、ソフトウェアは EOC フラグと関連の割り込みを使用して、各データ結果を処理する必要があります。変換が完了するたびに、ADC_ISR レジスタの EOC ビットがセットされ、ADC_DR レジスタを読み出すことができます。オーバーラインイベントをエラーとして管理するには、ADC_CFGR1 レジスタの OVRMOD ビットを 0 に設定する必要があります。

14.6.4 オーバーランなしでの DMA を使用しない変換データの管理

変換のたびにデータの読み出しをせずに ADC に 1 つまたは複数のチャンネルを変換させると便利な場合があります。この場合、OVRMOD ビットを 1 に設定する必要があります。ソフトウェアは OVR フラグを無視する必要があります。OVRMOD=1 のとき、オーバーランイベントが発生しても ADC は変換を続行し、ADC_DR レジスタは常に最新の変換データを含みます。

14.6.5 DMA を使用した変換データの管理

変換されたレギュラチャンネルの値はすべて、単一のデータレジスタに格納されるので、複数のチャンネルを変換するときには DMA を使用すると効率的です。これによって、ADC_DR レジスタに格納されている変換データ結果が失われるのを避けることができます。

DMA モードが有効なとき (ADC_CFGR1 レジスタの DMAEN ビットが 1 にセットされている)、各チャンネルの変換後、DMA リクエストが生成されます。これにより、変換されたデータを ADC_DR レジスタからソフトウェアで選択した場所へ転送することができます。

注: ADC_CFGR1 レジスタの DMAEN ビットは、ADC 較正フェーズ後にセットする必要があります。

これにもかかわらず、DMA が DMA 転送リクエストを時間内に処理できなかったためにオーバーランが発生した場合 (OVR=1)、ADC は DMA リクエストの生成を停止し、新しい変換に対応するデータは DMA によって転送されません。これは、RAM に転送されるすべてのデータを有効とみなすことができることを意味します。

OVRMOD ビットの設定に応じて、データは保存または上書きされます ([セクション 14.6.2 : ADC オーバーラン \(OVR、OVRMOD\) \(281 ページ\)](#) を参照してください)。

DMA 転送リクエストは、ソフトウェアが OVR ビットをクリアするまでブロックされます。

アプリケーションの用途に応じて 2 つの DMA モードがあり、ADC_CFGR1 レジスタのビット DMACFG で設定されます。

- DMA ワンショットモード (DMACFG=0)。
DMA が固定数のデータワードを転送するようにプログラムされたときには、このモードを選択してください。
- DMA サークュラモード (DMACFG=1)
DMA をサーキュラモードまたはダブルバッファモードでプログラムするときには、このモードを選択してください。

DMA ワンショットモード (DMACFG=0)

このモードでは、ADC は新しい変換データワードが使用可能になるたびに DMA 転送リクエストを生成し、変換が再び開始された場合でも、DMA が最後の DMA 転送に達すると (DMA_EOT 割り込みが発生すると) ([セクション 11 : ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#) を参照)、DMA リクエストの生成を停止します。

DMA 転送が完了すると (DMA コントローラで設定されたすべての転送が終了すると)、

- ADC データレジスタの内容が停止されます。
- 実行中の変換は中止され、その部分的な結果は破棄されます。
- DMA コントローラに対する新しいDMA リクエストは発行されません。これによって、開始された変換がある場合のオーバーランエラーの生成を回避します。
- スキャンシーケンスは中止され、リセットされます。
- DMA は停止します。

DMA サークュラモード (DMACFG=1)

このモードでは、DMA が最後の DMA 転送に達した場合でも、ADC は新しい変換データワードがデータレジスタで使用可能になるたびに DMA 転送リクエストを生成します。これにより、DMA を連続的なアナログ入力データストリームを処理するようにサーキュラモードで設定できます。

14.7 低電力機能

14.7.1 ウェイトモード変換

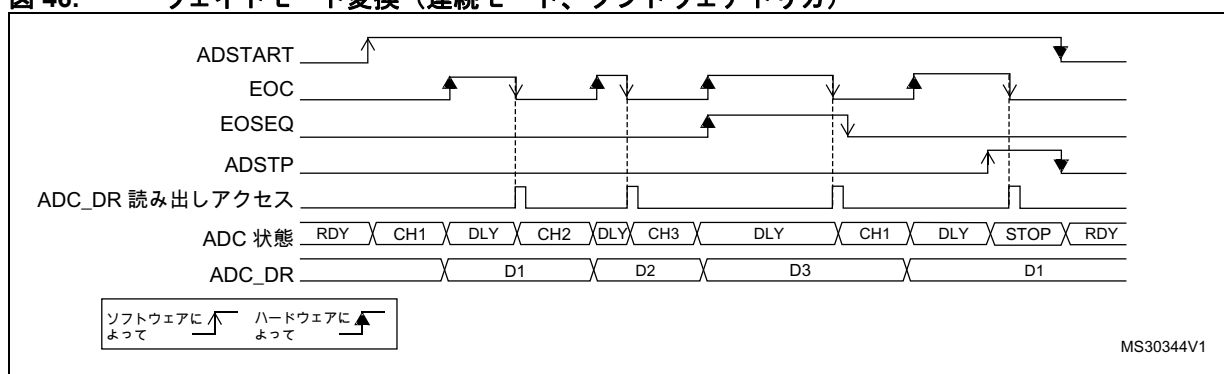
ウェイトモード変換を使用すると、ソフトウェアを単純化するだけでなく、ADC オーバーランが発生するリスクのある低周波数のクロックで動作しているアプリケーションのパフォーマンスを最適化できます。

ADC_CFGR1 レジスタの WAIT ビットが 1 にセットされているとき、新しい変換は、前のデータが処理された場合、ADC_DR レジスタが読み出された場合、または EOC ビットがクリアされた場合のみ開始できます。

これは、ADC の速度をデータを読み出すシステムの速度に自動的に適応させる方法です。

注： 変換中または読み出しアクセス前のウェイト時間中に発生したハードウェアトリガは無視されます。

図 46. ウェイトモード変換（連続モード、ソフトウェアトリガ）



1. EXTEN=1x0、CONT=0
2. CHSEL=0x3、SCANDIR=0、WAIT=1、AUTOFF=0

14.7.2 オートオフモード (AUTOFF)

ADC にはオートオフモードと呼ばれる自動電源管理機能があり、ADC_CFGR1 レジスタの AUTOFF=1 をセットすることによって有効になります。

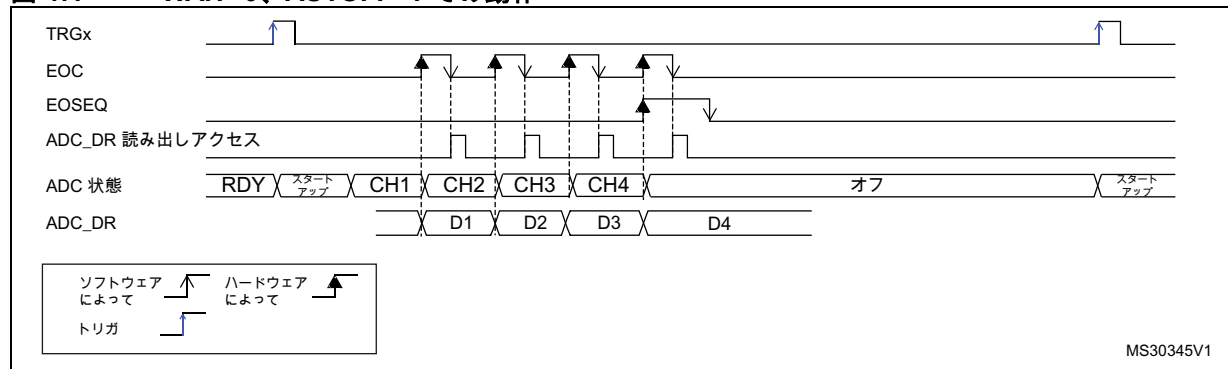
AUTOFF=1 のとき、ADC は、変換中でないときには常に電源がオフであり、（ソフトウェアまたはハードウェアトリガによって）変換が開始されると自動的にウェイクアップします。変換を開始するトリガイイベントと ADC のサンプリング時間の間に、スタートアップ時間が自動的に挿入されます。変換シーケンスが完了すると、ADC は自動的に無効になります。

オートオフモードは、アプリケーションが比較的少ない変換しか必要としないとき、または変換リクエストの間隔が十分に離れているとき（低周波数のハードウェアトリガによる場合など）に消費電力を大幅に削減できるため、ADC のオンとオフを切り替えるために余分な電力と余分な時間がかかって引き合います。

低周波数のクロックで動作するアプリケーションの場合、オートオフモードとウェイトモード変換 (WAIT=1) を組み合わせることができます。この組み合わせは、ADC がウェイトフェーズで自動的に電源オフされ、ADC_DR レジスタがアプリケーションによって読み出されるとすぐに再起動する場合、大幅な節電を可能にします（図 48 : WAIT=1、AUTOFF=1 での動作を参照してください）。

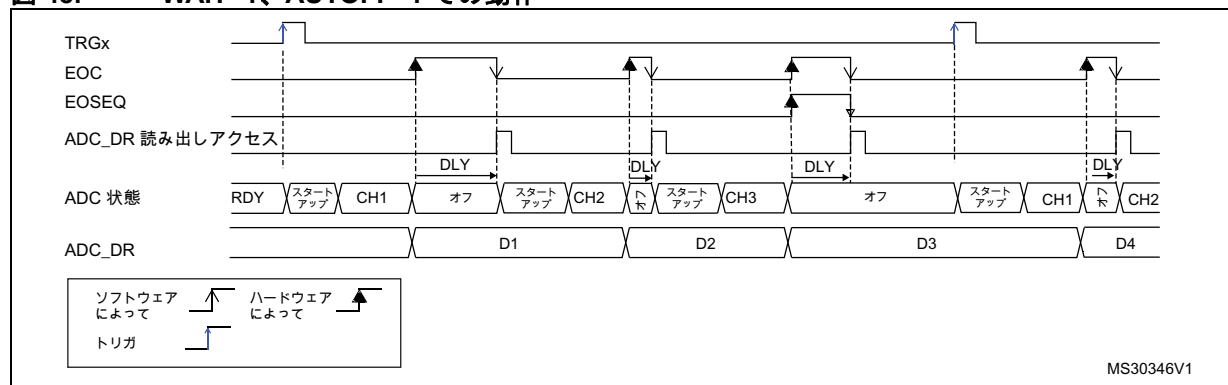
注： 専用 14 MHz 内部オシレータを管理する方法については、[セクション 7：リセットおよびクロック制御 \(RCC\) \(143 ページ\)](#) を参照してください。ADC インタフェースは、節電のために、14 MHz 内部オシレータの ON / OFF を自動的に切り替えることができます。

図 47. WAIT=0、AUTOFF=1 での動作



- EXTSEL=TRGx, EXTEN=0x1 (立ち上がりエッジ)、CONT=x, ADSTART=1, CHSEL=0xF, SCANDIR=0, WAIT=1, AUTOFF=1

図 48. WAIT=1、AUTOFF=1 での動作



- EXTSEL=TRGx, EXTEN=0x1 (立ち上がりエッジ)、CONT=x, ADSTART=1, CHSEL=0xF, SCANDIR=0, WAIT=1, AUTOFF=1

14.8 アナログウィンドウウォッチドッグ (AWDEN、AWDSGL、AWDCH、AWD_HTR/LTR、AWD)

AWD アナログウォッチドッグ機能は、ADC_CFGR1 レジスタの AWDEN ビットをセットすることによって有効になります。図 49 に示すように、選択された 1 つのチャンネルまたはすべての有効チャンネル (表 55: アナログウォッチドッグチャンネル選択を参照) が設定された電圧範囲 (ウィンドウ) 内にとどまっているかどうかを監視するために使用されます。

アナログウォッチドッグ (AWD) ステータスビットは、ADC によって変換されたアナログ電圧が低閾値を下回るか、高閾値を上回る場合にセットされます。これらの閾値は、ADC_HTR および ADC_LTR 16 ビットレジスタの下位 12 ビットに設定します。ADC_IER レジスタの AWDIE ビットをセットすることによって、割り込みを有効にできます。

AWD フラグは、ソフトウェアによって 1 を書き込むことによってクリアされます。

12 ビット未満の分解能でデータを変換するときには (ビット DRES[1:0] に従って)、内部比較は常に 12 ビット全体の 元の変換データ (左詰め) に対して実行されるため、プログラムされた閾値の LSB はクリアされたままである必要があります。

表 54 に、可能なすべての分解能での比較方法を示します。

表 54. アナログウォッチドッグ比較

| 分解能ビット RES[1:0] | アナログウォッチドッグ比較: | | コメント |
|--------------------|--------------------------------|---------------------|---|
| | 元の変換データ、 左詰め ⁽¹⁾ | 閾値 | |
| 00 : 12 ビット | DATA[11:0] | LT[11:0] と HT[11:0] | - |
| 01 : 10 ビット | DATA[11:2],00 | LT[11:0] と HT[11:0] | ユーザは LT1[1:0] と HT1[1:0] を "00" に設定する必要があります。 |
| 10 : 8 ビット | DATA[11:4],0000 | LT[11:0] と HT[11:0] | ユーザは LT1[3:0] と HT1[3:0] を "0000" に設定する必要があります。 |
| 11 : 6 ビット | DATA[11:6],000000 | LT[11:0] と HT[11:0] | ユーザは LT1[5:0] と HT1[5:0] を "000000" に設定する必要があります。 |

1. 配置計算の前に、元の変換データに対するウォッチドッグ比較が行われます。

表 55 に、ADC_CFGR1 レジスタの AWDSGL および AWDEN ビットを設定して、1 つ以上のチャンネルに対してアナログウォッチドッグを有効にする方法を示します。

図 49. アナログウォッチドッグによって保護される領域

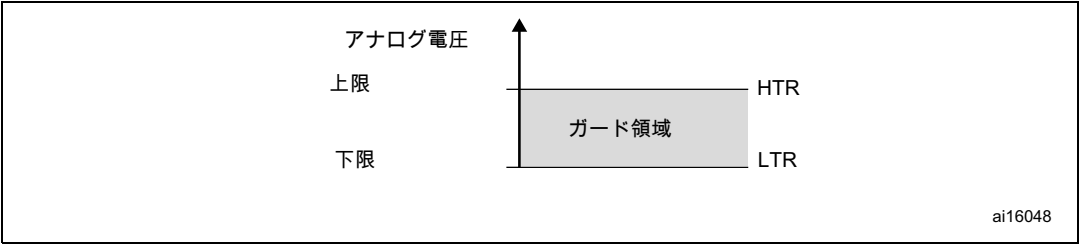


表 55. アナログウォッチドッグチャンネル選択

| アナログウォッチドッグによって保護されるチャンネル | AWDSGL ビット | AWDEN ビット |
|---------------------------|------------|-----------|
| なし | x | 0 |
| すべてのチャンネル | 0 | 1 |
| 単一の ⁽¹⁾ チャンネル | 1 | 1 |

1. AWDC[4:0] ビットによって選択

14.9 オーバーサンプリング回路

オーバーサンプリングユニットは、データの前処理を実行して、CPU の負荷を軽減します。複数の変換を処理して、最大 16 ビット幅の単一データに平均化できます。

以下の形式で結果を提供します。N および M は調整可能です。

$$\text{Result} = \frac{1}{M} \times \sum_{n=0}^{n=N-1} \text{Conversion}(t_n)$$

平均化、データレートの削減、SNR の向上、基本的フィルタリングをハードウェアによって実行できます。

オーバーサンプリング比 N は、ADC_CFGR2 レジスタの OVFS[2:0] ビットによって定義されます。2x から 256x までの範囲にできます。分周係数 M は、最大 8 ビットの右ビットシフトから成ります。これは、ADC_CFGR2 レジスタの OVSS[3:0] ビットによって設定されます。

合計ユニットは最大 20 ビット (256 x 12 ビット) の結果をもたらし、最初に右へシフトされます。結果の上位ビットは切り詰められ、最下位 16 ビットのみが保持されて、シフトによって残った最下位ビットを使用して最も近い値に丸められてから ADC_DR データレジスタに転送されます。

注： シフト後の中間結果が 16 ビットを超える場合、結果の上位ビットは単純に切り詰められます。

図 50. 20 ビットから 16 ビットへの結果の切り詰め

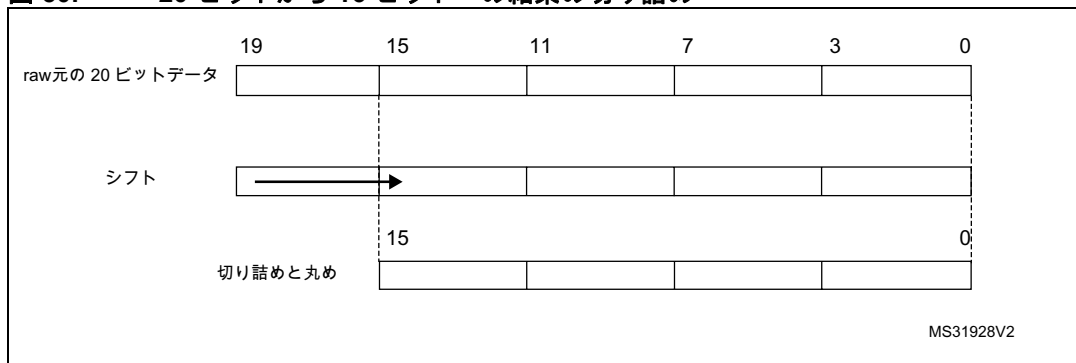
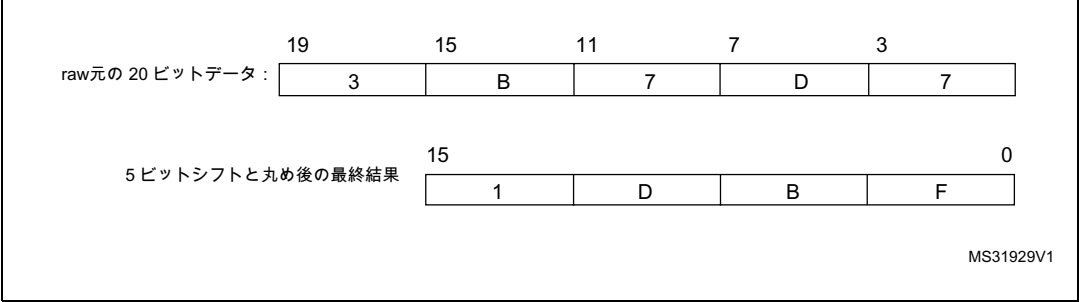


図 51 に、元の 20 ビットの累積データから最終的な 16 ビットの結果への処理の数値例を示します。

図 51. 5 ビットシフトと丸めの数値例



下の 表 56 に、元の 変換データが 0 x FFF の場合の、さまざまな N と M の組み合わせでのデータフォーマットを示します。

表 56. 最大出力結果対 N と M。グレイの値は切り詰めを示す

| オーバー サンプ リング比 | 最大 元のデータ | シフト なし OVSS = 0000 | 1 ビット シフト OVSS = 0001 | 2 ビット シフト OVSS = 0010 | 3 ビット シフト OVSS = 0011 | 4 ビット シフト OVSS = 0100 | 5 ビット シフト OVSS = 0101 | 6 ビット シフト OVSS = 0110 | 7 ビット シフト OVSS = 0111 | 8 ビット シフト OVSS = 1000 |
|---------------------|-------------|-----------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|
| 2x | 0x1FFE | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 | 0x0200 | 0x0100 | 0x0080 | 0x0040 | 0x0020 |
| 4x | 0x3FFC | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 | 0x0200 | 0x0100 | 0x0080 | 0x0040 |
| 8x | 0x7FF8 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 | 0x0200 | 0x0100 | 0x0080 |
| 16x | 0xFFF0 | 0xFFF0 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 | 0x0200 | 0x0100 |
| 32x | 0x1FFE0 | 0xFFE0 | 0xFFF0 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 | 0x0200 |
| 64x | 0x3FFC0 | 0xFFC0 | 0xFFE0 | 0xFFF0 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 | 0x0400 |
| 128x | 0x7FF80 | 0xFF80 | 0xFFC0 | 0xFFE0 | 0xFFF0 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF | 0x0800 |
| 256x | 0xFFF00 | 0xFF00 | 0xFF80 | 0xFFC0 | 0xFFE0 | 0xFFF0 | 0x7FF8 | 0x3FFC | 0x1FFE | 0x0FFF |

オーバーサンプリングモードでの変換タイミングは、標準変換モードと同じです。サンプル時間はオーバーサンプリングシーケンス全体を通じて一定に保たれます。新しいデータは N 回の変換ごとに提供され、同等遅延は $N \times t_{ADC} = N \times (t_{SMPL} + t_{SAR})$ に等しくなります。フラグ機能は、次のように実行されます。

- サンプルングフェーズの終了 (EOSMP) は、各サンプリングフェーズ後にセットされます。
- 変換の終了 (EOC) は、N 回の変換ごとに発生し、オーバーサンプリングされた結果が使用可能になります。
- シーケンスの終了 (EOCSEQ) は、オーバーサンプリングされたデータのシーケンスが完了すると発生します (すなわち、N x シーケンス長の変換合計後)。

14.9.1 オーバーサンプリング時の ADC 動作モードのサポート

オーバーサンプリングモードでは、ほとんどの ADC 動作モードが使用可能です。

- シングルまたは連続モード変換、前方または後方スキャンシーケンス
- ソフトウェアまたはトリガによる ADC 変換の開始
- 変換中の ADC の停止 (中止)
- オーバーラン検出時の CPU または DMA 経由でのデータの読み出し
- 低電力モード (WAIT、AUTOFF)
- プログラム可能な分解能: この場合、削減された変換値 (ADC_CFGR1 レジスタの RES[1:0] ビットに従って) の累積、切り詰め、丸め、およびシフトは、12 ビット変換と同様に行われます。

注: *オーバーサンプリングされたデータを操作するときには、配置モードは使用できません。ADC_CFGR1 の ALIGN ビットは無視され、データは常に右詰めで提供されます。*

14.9.2 アナログウォッチドッグ

アナログウォッチドッグ機能を使用できますが (AWDSGL および AWDEN ビット)、次のような違いがあります。

- RES[1:0] ビットは無視され、比較は常に 12 ビット値 HT[11:0] および LT[11:0] を使用して行われます。
- 比較は、16 ビットのオーバーサンプリングされた結果の上位 12 ビット ADC_DR[15:4] に対して行われます。

注: *高いシフト値を使用するときには注意が必要です。これによって比較範囲が小さくなります。たとえば、オーバーサンプリングされた結果が 4 ビットシフトされた場合、12 ビットの右詰めデータになり、有効なアナログウォッチドッグ比較が行われるのは、8 ビットに対してだけです。比較は ADC_DR[11:4] と HT[0:7] / LT[0:7] の間で行われ、HT[11:8] / LT[11:8] はリセットする必要があります。*

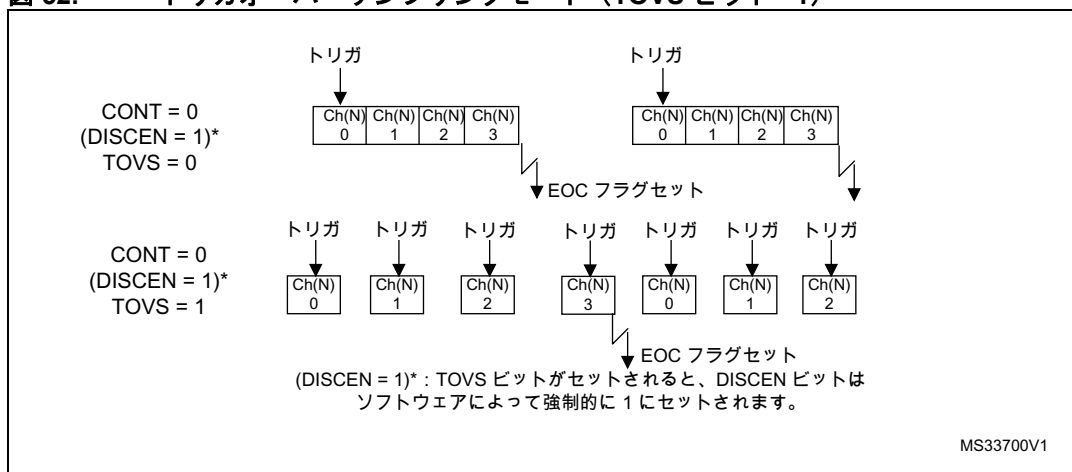
14.9.3 トリガモード

平均化回路は、基本的なフィルタリング目的で使用することもできます。あまり効率的なフィルタではありませんが (ロールオフが低速であり、阻止帯域減衰に限られる)、ノッチフィルタとして使用して、周期的なノイズ周波数を低減できます (一般に主電源またはスイッチモードの電源が原因)。この目的のために、ADC_CFGR2 の TOVS ビットで特定の不連続モードを有効にして、変換時間に依存せずに、オーバーサンプリング周波数をユーザが定義できるようにできます。

[図 52](#) に、不連続モードでトリガに反応して変換を開始する方法を示します。

TOVS ビットがセットされている場合、DISCEN ビットの内容は無視され、1 とみなされます。

図 52. トリガオーバーサンプリングモード (TOVS ビット = 1)



14.10 温度センサと内部基準電圧

温度センサを使用して、デバイスの接合温度 (T_J) を測定できます。温度センサは、センサの出力電圧をデジタル値に変換する ADC_IN18 入力チャネルに内部接続されます。温度センサのアナログピンのサンプリング時間は、2.2 μ s 以上である必要があります。使用しないときには、センサをパワーダウンモードにできます。

内部電圧基準 (VREFINT) は、ADC とコンパレータに安定した (バンドギャップ) 電圧出力を提供します。VREFINT は、ADC_IN17 入力チャネルに内部接続されます。VREFINT の正確な電圧は、生産試験時に ST によって部品ごとに個別に測定され、システムメモリ領域に格納されます。読み出し専用モードでアクセスできます。

図 53 に、温度センサ、内部電圧基準、および ADC 間の接続のブロック図を示します。

ADC_IN18 (温度センサ) の変換を有効にするには、TSEN ビットをセットする必要があります。ADC_IN17 (VREFINT) の変換を有効にするには、VREFEN ビットをセットする必要があります。

温度センサの出力電圧は、温度に比例して変化します。このラインのオフセットは、プロセスのばらつきにより、チップごとに異なります (チップ間で最大 45 °C)。

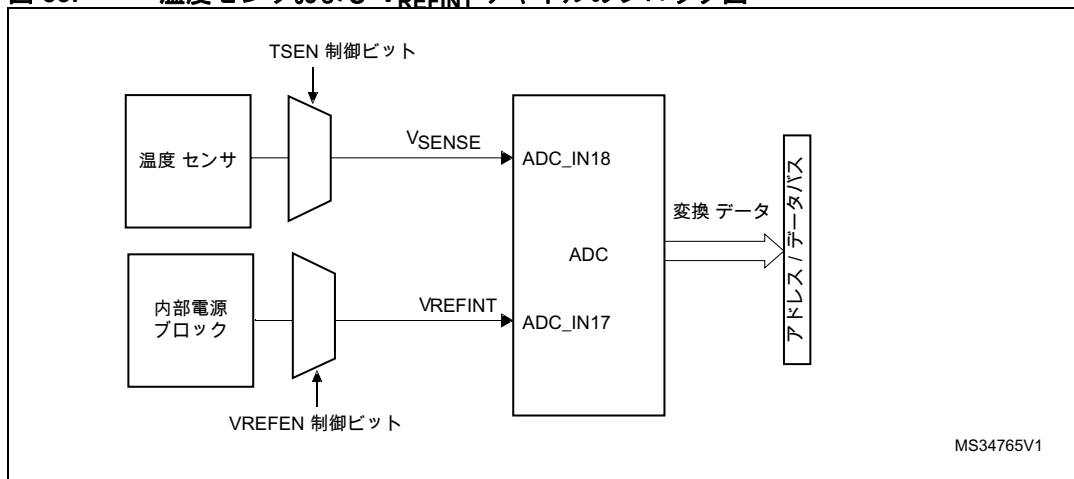
較正されていない内部温度センサは、温度の絶対値の代わりに温度変化を検出するアプリケーションに適しています。温度センサの測定精度を高めるために、生産時に ST によって各デバイスの較正值がシステムメモリに格納されています。

製造プロセス中に、温度センサと内部電圧基準の較正データがシステムメモリ領域に格納されます。ユーザアプリケーションはこれらを読み出して、温度センサまたは内部基準の精度の向上に使用できます。詳細については、データシートを参照してください。

主な特長

- サポートしている温度範囲: -40 ~ 125 °C
- 直線性: 最大 $\pm 2^\circ$ C、精度は較正に依存

図 53. 温度センサおよび V_{REFINT} チャンネルのブロック図



温度の読み出し

1. ADC_IN18 入力チャンネルを選択します。
2. デバイスのデータシートで指定されている適切なサンプリング時間 (T_{S_temp}) を選択します。
3. ADC_CCR レジスタの TSEN ビットをセットして、温度センサをパワーダウンモードからウェイクアップし、安定時間 (t_{START}) だけ待ちます。
4. ADC_CR レジスタの ADSTART ビットをセットすることによって (または外部トリガによって) ADC 変換を開始します。
5. ADC_DR レジスタから結果の V_{SENSE} データを読み出します。
6. 次の式を使用して、温度を計算します。

$$\text{温度 (}^{\circ}\text{C)} = \frac{130^{\circ}\text{C} - 30^{\circ}\text{C}}{\text{TS_CAL2} - \text{TS_CAL1}} \times (\text{TS_DATA} - \text{TS_CAL1}) + 30^{\circ}\text{C}$$

ここで、

- TS_CAL2 は、130°C で得られた温度センサの較正值です。
- TS_CAL1 は、30°C で得られた温度センサの較正值です。
- TS_DATA は、ADC によって変換された実際の温度センサの出力値です。

TS_CAL1 および TS_CAL2 較正ポイントの詳細については、特定のデバイスのデータシートを参照してください。

注： センサがパワーダウンモードからウェイクアップして、正しいレベルで V_{SENSE} を出力できるようになるまでには時間がかかります (スタートアップ時間)。ADC にも起動後のスタートアップ時間があるので、遅延を最小にするには、ADEN ビットと TSEN ビットを同時にセットしてください。

内部基準電圧を使用した実際の V_{DDA} 電圧の計算

マイクロコントローラに印加される V_{DDA} 電源電圧は、ばらつきがあり、または正確にはわかりません。埋め込みの内部電圧基準 (VREFINT) と、製造プロセス時に V_{DDA} = 3.3 V で ADC によって得られた較正データを使用して、実際の V_{DDA} 電圧レベルを評価することができます。

デバイスに印加される実際の V_{DDA} 電圧は、次の式で求められます。

$$V_{DDA} = 3.3 \text{ V} \times \text{VREFINT_CAL} / \text{VREFINT_DATA}$$

ここで、

- VREFINT_CAL は、VREFINT の較正值です。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。

供給に相対的な ADC 測定値から絶対電圧値への変換

ADC は、アナログ電源と変換されるチャンネルに印加される電圧との比に対応するデジタル値を提供するように設計されています。ほとんどのアプリケーションの使用事例では、この比を V_{DDA} に依存しない電圧に変換する必要があります。 V_{DDA} が既知であり、ADC によって変換された値が右詰めされるアプリケーションでは、次の式を使用して、この絶対値を求めることができます。

$$V_{\text{CHANNEL}x} = \frac{V_{DDA}}{\text{FULL_SCALE}} \times \text{ADC_DATA}_x$$

V_{DDA} 値が既知ではないアプリケーションの場合、内部電圧基準を使用する必要があり、 V_{DDA} をセクション [内部基準電圧を使用した実際の \$V_{DDA}\$ 電圧の計算](#) に記載されている式で置き換えることができます、次の式を使用できます。

$$V_{\text{CHANNEL}x} = \frac{3.3 \text{ V} \times \text{VREFINT_CAL} \times \text{ADC_DATA}_x}{\text{VREFINT_DATA} \times \text{FULL_SCALE}}$$

ここで、

- VREFINT_CAL は、VREFINT の較正值です。
- ADC_DATA_x は、チャンネル x で ADC によって測定された値です（右詰め）。
- VREFINT_DATA は、ADC によって変換された実際の VREFINT の出力値です。
- FULL_SCALE は、ADC 出力の最大デジタル値です。たとえば、12 ビット分解能では $2^{12} - 1 = 4095$ になり、8 ビット分解能では $2^8 - 1 = 255$ になります。

注： ADC 測定が 12 ビット右詰め以外の出力形式を使用して行われる場合、計算を行う前に、すべてのパラメータを互換性のある形式に変換しておく必要があります。

14.11 V_{LCD} 電圧監視

ADC_CCR レジスタの VLCDEN ビットによって、 V_{LCD} ピンの LCD 電源を測定できます。 V_{LCD} 電圧は V_{DDA} より高くなることがあるので、ADC の正しい動作を確保するために、 V_{LCD} ピンはブリッジ 4 分圧回路に内部接続されています。このブリッジは、VLCDEN ビットがセットされると自動的に有効になり、 $V_{\text{LCD}}/2$ を ADC1_IN18 入力チャンネルに接続します。結果として、変換されたデジタル値は V_{LCD} 電圧の 1/4 です。望ましくないバッテリー消費を避けるには、必要なときだけ、すなわち ADC 変換を行うときだけ、ブリッジ分圧回路を有効にすることが推奨されます。

14.12 ADC 割り込み

割り込みは、次のイベントによって生成できます。

- 較正の終了 (EOCAL フラグ)
- ADC パワーアップ、ADC の準備ができたとき (ADRDY フラグ)
- 変換の終了 (EOC フラグ)
- 変換シーケンスの終了 (EOSEQ フラグ)
- アナログウォッチドッグ検出の発生時 (AWD フラグ)
- サンプリングフェーズの終了が発生したとき (EOSMP フラグ)
- データオーバーランの発生時 (OVR フラグ)

高い柔軟性を実現するため、個別の割り込みイネーブルビットを使用できます。

表 57. ADC 割り込み

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|-------------------------|---------|------------|
| 較正終了 | EOCAL | EOCALIE |
| ADC レディ | ADRDY | ADRDYIE |
| 変換の終了 | EOC | EOCIE |
| 変換シーケンスの終了 | EOSEQ | EOSEQIE |
| アナログウォッチドッグステータスビットのセット | AWD | AWDIE |
| サンプリングフェーズの終了 | EOSMP | EOSMPIE |
| オーバーラン | OVR | OVRIE |

14.13 ADC レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

14.13.1 ADC 割り込みおよびステータスレジスタ (ADC_ISR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|-------|------|------|------|------|------|------|------|-------|-------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | EOCAL | Res. | Res. | Res. | AWD | Res. | Res. | OVR | EOSEQ | EOC | EOSMP | ADRDY |
| | | | | r_w1 | | | | r_w1 | | | r_w1 | r_w1 | rc_w1 | r_w1 | r_w1 |

ビット 31:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11 **EOCAL** : 校正終了フラグ

このビットは、校正が完了したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0 : 校正は完了していません。

1 : 校正は完了しています。

ビット 10:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **AWD**: アナログウォッチドッグフラグ

このビットは、変換された電圧が、ADC_LTR および ADC_HTR レジスタでプログラミングされた値を逸脱したときに、ハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0 : アナログウォッチドッグイベントは発生しませんでした (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。

1 : アナログウォッチドッグイベントが発生しました。

ビット 6:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **OVR** : ADC オーバーラン

このビットは、オーバーランが発生したときにハードウェアによってセットされ、EOC フラグがすでにセットされているときに新しい変換が完了したことを意味します。ソフトウェアで 1 を書き込むことによってクリアされます。

0 : オーバーランは発生しませんでした (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。

1 : オーバーランが発生しました。

ビット 3 **EOSEQ** : シーケンス終了フラグ

このビットは、CHSEL ビットによって選択されたチャネルのシーケンスの変換終了時にハードウェアによってセットされます。ソフトウェアで 1 を書き込むことによってクリアされます。

0 : 変換シーケンスは完了していません (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。

1 : 変換シーケンスが完了しました。

ビット 2 **EOC** : 変換終了フラグ

このビットは、チャンネルの各変換の終了時に、新しいデータ結果が ADC_DR レジスタで使用可能になったときに、ハードウェアによってセットされます。ソフトウェアによって 1 を書き込むことによって、または ADC_DR レジスタを読み出すことによってクリアされます。

- 0 : チャンネル変換は完了していません (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。
- 1 : チャンネル変換が完了しました。

ビット 1 **EOSMP** : サンプリング終了フラグ

このビットは、変換中、サンプリングフェーズの終了時にハードウェアによってセットされます。

- 0 : サンプリングフェーズの終了時ではありません (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。
- 1 : サンプリングフェーズの終了に達しました。

ビット 0 **ADRDY** : ADC レディ

このビットは、ADC が有効にされた後 (ビット ADEN=1)、ADC が変換リクエストを受け入れる準備ができた状態に達したときに、ハードウェアによってセットされます。

ソフトウェアで 1 を書き込むことによってクリアされます。

- 0 : ADC はまだ変換を開始する準備ができていません (またはフラグイベントはソフトウェアによってすでに確認され、クリアされました)。
- 1 : ADC は変換を開始する準備ができました。

14.13.2 ADC 割り込み有効レジスタ (ADC_IER)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|---------|------|------|------|-------|------|------|-------|---------|-------|---------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | EOCALIE | Res. | Res. | Res. | AWDIE | Res. | Res. | OVRIE | EOSEQIE | EOCIE | EOSMPIE | ADRDYIE |
| | | | | rw | | | | rw | | | rw | rw | rw | rw | rw |

ビット 31:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11 **EOCALIE** : 較正終了割り込みイネーブル

このビットは、較正終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

- 0 : 較正終了割り込みは無効です。
- 1 : 較正終了割り込みは有効です。

注 : ソフトウェアは、ADSTART=0 (変換が進行中でない) ときのみ、このビットに書き込むことができます。

ビット 10:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **AWDIE** : アナログウォッチドッグ割り込みイネーブル

このビットは、アナログウォッチドッグ割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

- 0 : アナログウォッチドッグ割り込みは無効です。
- 1 : アナログウォッチドッグ割り込みは有効です。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 6:5 予約済みであり、リセット値のままにしておかなければなりません。



ビット 4 OVRIE : オーバーラン割り込みイネーブル

このビットは、オーバーラン割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : オーバーラン割り込みは無効です。

1 : オーバーラン割り込みは有効です。OVR ビットがセットされると、割り込みが生成されます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 3 EOSEQIE : 変換シーケンス終了割り込み有効

このビットは、変換シーケンス終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : EOSEQ 割り込みは無効です。

1 : EOSEQ 割り込みは有効です。EOSEQ ビットがセットされると、割り込みが生成されます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 2 EOCIE : 変換終了割り込み有効

このビットは、変換終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : EOC 割り込みは無効です。

1 : EOC 割り込みは有効です。EOC ビットがセットされると、割り込みが生成されます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 1 EOSMPIE : サンプリング終了フラグ割り込み有効

このビットは、サンプリングフェーズ終了時の割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : EOSMP 割り込みは無効です。

1 : EOSMP 割り込みは有効です。EOSMP ビットがセットされると、割り込みが生成されます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 0 ADRDYIE : ADC レディ割り込みイネーブル

このビットは、ADC レディ割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : ADRDY 割り込みは無効です。

1 : ADRDY 割り込みは有効です。ADRDY ビットがセットされると、割り込みが生成されます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

14.13.3 ADC 制御レジスタ (ADC_CR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|-------|------|------|--------------|------|------|------|------|------|------|------|------------|------|-------------|-------|------|
| ADCAL | Res. | Res. | ADVRE GEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| rs | | | rw | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADSTP T | Res. | ADSTAR T | ADDIS | ADEN |
| | | | | | | | | | | | rs | | rs | rs | rs |

ビット 31 ADCAL : ADC 較正

このビットは、ADC の較正を開始するためにソフトウェアによってセットされます。

較正の完了後、ハードウェアによってクリアされます。

0 : 較正が完了しました。

1 : ADC を較正するには、1 を書き込みます。1 として読み出されたときには、較正が実行中であることを意味します。

注 : ソフトウェアは、ADC が無効のときだけ (ADCAL=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0)、ADCAL をセットできます。

注 : ソフトウェアは、ADEN=1 かつ ADSTART=0 (ADC が有効であり、変換中でない) のときだけ、ADC_CALFACT を書き込むことによって較正係数を更新できます。

ビット 30:29 予約済みであり、リセット値のままにしておかなければなりません。

ビット 28 ADVREGEN : ADC 電圧レギュレータイネーブル

このビットは、以下のようにセットされます。

- ADC 内部電圧レギュレータを有効にするために、ソフトウェアによってセットされます。
- 較正を始めるとき (ADCAL=1 を設定) や、ADC を有効にするとき (ADEN=1 を設定) に、ハードウェアによってセットされます。

電圧レギュレータを無効にするために、ソフトウェアによってクリアされます (ADEN=0 の場合のみクリアされる)。

0 : ADC 電圧レギュレータは無効です。

1 : ADC 電圧レギュレータは有効です。

注 : ソフトウェアは、ADC が無効のときのみ (ADCAL=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0)、このビットフィールドをプログラムできます。

ビット 27:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 ADSTP : ADC 変換停止コマンド

このビットは、実行中の変換を停止および破棄するためにソフトウェアによってセットされます (ADSTP コマンド)。

変換が効果的に破棄され、ADC が新しい変換開始コマンドを受け入れる準備ができたときに、ハードウェアによってクリアされます。

0 : ADC 変換停止コマンドは実行中ではありません。

1 : ADC を停止するには、1 を書き込みます。1 として読み出されたときには、ADSTP コマンドが実行中であることを意味します。

注 : ソフトウェアは、ADSTART=1 かつ ADDIS=0 (ADC が有効であり、変換中である可能性があり、ADC を無効にする保留中のリクエストがない) のときだけ、ADSTP をセットできます。

ビット 3 予約済みであり、リセット値のままにしておかなければなりません。



ビット 2 ADSTART : ADC 変換開始コマンド

このビットは、ADC 変換を開始するためにソフトウェアによってセットされます。EXTEN [1:0] 設定ビットに応じて、変換はただちに開始されるか（ソフトウェアトリガ設定）またはハードウェアトリガイベントが発生したときに（ハードウェアトリガ設定）開始されます。

次のときに、ハードウェアによってクリアされます。

- シングル変換モードでは、ソフトウェアトリガが選択されたとき (EXTSEL=0x0): 変換シーケンス終了 (EOSEQ) フラグのアサート時。
- すべての場合 : ADSTP コマンドの実行後、ADSTP ビットがハードウェアによってクリアされると同時に。

0 : ADC 変換は実行中ではありません。

1 : ADC を開始するには、1 を書き込みます。1 として読み出されたときには、ADC が動作中であり、変換中である可能性があることを意味します。

注 : ソフトウェアは、ADEN=1 かつ ADDIS=0 (ADC が有効であり、ADC を無効にする保留中のリクエストがない) のときだけ、ADSTART をセットできます。

ビット 1 ADDIS:ADC 無効化コマンド

このビットは、ADC を無効にして (ADDIS コマンド)、パワーダウン状態 (OFF 状態) にするためにソフトウェアによってセットされます。

ADC が効果的に無効化されると、ハードウェアによってクリアされます (ADEN もこの時点でハードウェアによってクリアされます)。

0 : ADDIS コマンドは実行中ではありません。

1 : ADC を無効にするには、1 を書き込みます。1 として読み出されたときには、ADDIS コマンドが実行中であることを意味します。

注 : ソフトウェアは、ADEN=1 かつ ADSTART=0 (変換が実行中でない) のときのみ、ADDIS をセットできます。

ビット 0 ADEN : ADC 有効化コマンド

このビットは、ADC を有効にするために、ソフトウェアによってセットされます。ADRDY フラグがセットされると、ADC は動作する準備ができています。

ADDIS コマンドの実行後、ADC が無効になるとハードウェアによってクリアされます。

0 : ADC は無効です (OFF 状態)。

1 : ADC を有効にするには、1 を書き込みます。

注 : ソフトウェアは、ADC_CR レジスタのすべてのビットが 0 のとき (ADCAL=0、ADSTP=0、ADSTART=0、ADDIS=0、および ADEN=0) だけ、ADEN をセットできます。

14.13.4 ADC 設定レジスタ 1 (ADC_CFGR1)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------|------------|------|--------|------------|----|------|-------------|-------|--------|-------|----------|------|-------------|------------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | AWDCH[4:0] | | | | | Res. | Res. | AWDEN | AWDSGL | Res. | Res. | Res. | Res. | Res. | DISCEN |
| | rW | rW | rW | rW | rW | | | rW | rW | | | | | | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| AUTOFF | WAIT | CONT | OVRMOD | EXTEN[1:0] | | Res. | EXTSEL[2:0] | | | ALIGN | RES[1:0] | | SCANDI R | DMACF G | DMAEN |
| rW | rW | rW | rW | rW | | | rW | | | rW | rW | | rW | rW | rW |

- ビット 31 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 30:26 **AWDCH[4:0]** : アナログウォッチドッグチャンネル選択
これらのビットは、ソフトウェアによってセット／クリアされます。アナログウォッチドッグによって保護される入力チャンネルを選択します。
00000 : ADC アナログ入力チャンネル 0 が AWD によって監視されます。
00001 : ADC アナログ入力チャンネル 1 が AWD によって監視されます。
.....
10011 : ADC アナログ入力チャンネル 18 が AWD によって監視されます。
その他の値 : 予約済み。使用できません。
注 : **AWDCH [4:0] ビットによって選択されたチャンネルは、CHSELR レジスタにもセットされる必要があります。**
注 : **ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。**
- ビット 25:24 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 23 **AWDEN** : アナログウォッチドッグ有効
このビットは、ソフトウェアによってセット／クリアされます。
0 : アナログウォッチドッグは無効です。
1 : アナログウォッチドッグは有効です。
注 : **ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。**
- ビット 22 **AWDSGL** : ウォッチドッグを単一チャンネルまたはすべてのチャンネルで有効にします。
このビットは、AWDCH[4:0] ビットによって指定されたチャンネルまたはすべてのチャンネルに対するアナログウォッチドッグを有効にするために、ソフトウェアによってセット／クリアされます。
0 : すべてのチャンネルでアナログウォッチドッグは有効です。
1 : 単一チャンネルでアナログウォッチドッグは有効です。
注 : **ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。**
- ビット 21:17 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 16 **DISCEN** : 不連続モード
このビットは、不連続モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。
0 : 不連続モードは無効です。
1 : 不連続モードは有効です。
注 : **不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方のビットをセットすることは禁じられています。**
注 : **ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。**

ビット 15 AUTOFF : オートオフモード

このビットは、オートオフモードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : オートオフモードは無効です。

1 : オートオフモードは有効です。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 14 WAIT : ウェイト変換モード

このビットは、ウェイト変換モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : ウェイト変換モードはオフです。

1 : ウェイト変換モードはオンです。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 13 CONT : シングル／連続変換モード

このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされた場合、それがクリアされるまで連続的に変換が行われます。

0 : シングル変換モード

1 : 連続変換モード

注 : 不連続モードと連続モードの両方を有効にすることはできません。DISCEN=1 と CONT=1 の両方のビットをセットすることは禁じられています。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 12 OVRMOD : オーバーラン管理モード

このビットは、ソフトウェアによってセット／クリアされ、データオーバーランを管理する方法を設定します。

0 : オーバーランが検出されたとき、ADC_DR レジスタの古いデータが保存されます。

1 : オーバーランが検出されたとき、ADC_DR レジスタは最後の変換結果で上書きされます。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 11:10 EXTEN[1:0] : 外部トリガ有効および極性選択

これらのビットは、外部トリガ極性を選択し、トリガを有効にするために、ソフトウェアによってセット／クリアされます。

00 : ハードウェアトリガ検出は無効です (変換はソフトウェアによって開始できます)。

01 : 立ち上がりエッジでハードウェアトリガを検出します。

10 : 立ち下がりエッジでハードウェアトリガを検出します。

11 : 立ち上がりエッジと立ち下がりエッジの両方でハードウェアトリガを検出します。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。

ビット 9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8:6 EXTSEL[2:0] : 外部トリガ選択

これらのビットは、変換の開始をトリガするために使用される外部イベントを選択します (詳細については、表 52: 外部トリガを参照)。

000 : TRG0

001 : TRG1

010 : TRG2

011 : TRG3

100 : TRG4

101 : TRG5

110 : TRG6

111 : TRG7

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。

ビット 5 ALIGN : データの配置

このビットは、右詰めまたは左詰めを選択するために、ソフトウェアによってセット/クリアされます。[図 44 : データの配置と分解能 \(オーバーサンプリング無効 : OVSE = 0\) \(281 ページ\)](#) を参照してください。

- 0 : 右詰め
- 1 : 左詰め

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 4:3 RES[1:0] : データ分解能

これらのビットは、変換の分解能を選択するために、ソフトウェアによって書き込まれます。

- 00 : 12 ビット
- 01 : 10 ビット
- 10 : 8 ビット
- 11 : 6 ビット

注 : ソフトウェアは、ADEN=0 のときだけ、これらのビットを書き込むことができます。

ビット 2 SCANDIR : スキャンシーケンス方向

このビットは、シーケンス内のチャンネルをスキャンする方向を選択するために、ソフトウェアによってセット/クリアされます。

- 0 : 前方スキャン (CHSEL0 から CHSEL18)
- 1 : 後方スキャン (CHSEL18 から CHSEL0)

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 1 DMACFG : ダイレクトメモリアクセス設定

このビットは、2 つの DMA 動作モードを選択するためにソフトウェアによってセット/クリアされ、DMAEN=1 のときのみ有効です。

- 0 : DMA ワンショットモードが選択されています。
- 1 : DMA サーキュラモードが選択されています。

詳細については、[セクション 14.6.5 : DMA を使用した変換データの管理 \(282 ページ\)](#) を参照してください。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 0 DMAEN : ダイレクトメモリアクセス有効

このビットは、DMA リクエストの生成を有効にするために、ソフトウェアによってセット/クリアされます。これにより、DMA コントローラを使用して変換データを自動的に管理できます。詳細については、[セクション 14.6.5 : DMA を使用した変換データの管理 \(282 ページ\)](#) を参照してください。

- 0 : DMA は無効です。
- 1 : DMA は有効です。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

14.13.5 ADC 設定レジスタ 2（ADC_CFGR2）

アドレスオフセット：0x10

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|------|------|------|------|------|------|-----------|------|------|------|-----------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| CKMODE[1:0] | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| r/w | r/w | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | TOVS | OVSS[3:0] | | | | OVSR[2:0] | | | Res. | OVSE |
| | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | r/w |

ビット 31:30 **CKMODE[1:0]**：ADC クロックモード

これらのビットは、アナログ ADC クロックの動作方法を定義するために、ソフトウェアによってセット／クリアされます。

00：ADCLK（非同期クロックモード）。製品レベルで生成されます（RCC セクションを参照）。

01：PCLK/2（同期クロックモード）

10：PCLK/4（同期クロックモード）

11：PCLK（同期クロックモード）。この設定は、PCLK が 50% デューティクロックサイクルを持つ場合のみ有効にする必要があります（RCC 内で設定された APB ブリスケーラを迂回する必要があり、システムクロックは 50% デューティサイクルで動作する必要があります）。

すべての同期クロックモードにおいて、タイマトリガから変換開始までの遅延にジッタはありません。

注： ソフトウェアは、ADC が無効のときだけ (ADCAL=0、ADSTART=0、ADSTP=0、ADDIS=0、および ADEN=0)、これらのビットを書き込みことができます。

ビット 29:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **TOVS**：トリガオーバーサンプリング

このビットは、ソフトウェアによってセット／クリアされます。

0：チャンネルのオーバーサンプリング変換はすべて、トリガ後に連続的に行われます。

1：チャンネルのオーバーサンプリング変換ごとにトリガが必要です。

注： ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 8:5 **OVSS[3:0]**：オーバーサンプリングシフト

このビットは、ソフトウェアによってセット／クリアされます。

0000：シフトなし

0001：1 ビットシフト

0010：2 ビットシフト

0011：3 ビットシフト

0100：4 ビットシフト

0101：5 ビットシフト

0110：6 ビットシフト

0111：7 ビットシフト

1000：8 ビットシフト

その他のコードは予約済み

注： ソフトウェアは、ADSTART=0（変換が実行中でない）ときのみ、このビットに書き込むことができます。

ビット 4:2 **OVS[2:0]** : オーバーサンプリング比

このビットは、オーバーサンプリング比の数を定義します。

000 : 2x
001 : 4x
010 : 8x
011 : 16x
100 : 32x
101 : 64x
110 : 128x
111 : 256x

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **OVSE** : オーバーサンプリング回路有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : オーバーサンプリング回路は無効です。
1 : オーバーサンプリング回路は有効です。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、このビットに書き込むことができます。

14.13.6 ADC サンプリング時間レジスタ (ADC_SMPR)

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

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

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

ビット 2:0 **SMP[2:0]** : サンプリング時間選択

これらのビットは、すべてのチャンネルに適用されるサンプリング時間を選択するために、ソフトウェアによって書き込まれます。

000 : 1.5 ADC クロックサイクル
001 : 7.5 ADC クロックサイクル
010 : 13.5 ADC クロックサイクル
011 : 28.5 ADC クロックサイクル
100 : 41.5 ADC クロックサイクル
101 : 55.5 ADC クロックサイクル
110 : 71.5 ADC クロックサイクル
111 : 239.5 ADC クロックサイクル

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。



14.13.7 ADC ウォッチドッグ閾値レジスタ (ADC_TR)

アドレスオフセット : 0x20

リセット値 : 0x0FFF 0000

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

ビット 31:28 予約済みであり、リセット値のままにしておかなければなりません。

ビット 27:16 HT[11:0] : アナログウォッチドッグ高閾値

これらのビットは、アナログウォッチドッグの高閾値を定義するために、ソフトウェアによって書き込まれます。[セクション 14.8 : アナログウィンドウウォッチドッグ \(AWDEN, AWDSGL, AWDCH, AWD_HTR/LTR, AWD\) \(286 ページ\)](#) を参照してください。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。

ビット 15:12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11:0 LT[11:0] : アナログウォッチドッグ低閾値

これらのビットは、アナログウォッチドッグの低閾値を定義するために、ソフトウェアによって書き込まれます。[セクション 14.8 : アナログウィンドウウォッチドッグ \(AWDEN, AWDSGL, AWDCH, AWD_HTR/LTR, AWD\) \(286 ページ\)](#) を参照してください。

注 : ソフトウェアは、ADSTART=0 (変換が実行中でない) ときのみ、これらのビットを書き込むことができます。

14.13.8 ADC チャンネル選択レジスタ (ADC_CHSELR)

アドレスオフセット : 0x28

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|-------------|-------------|-------------|-------------|-------------|--------|--------|--------|--------|--------|--------|--------|-------------|-------------|-------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CHSEL1 8 | CHSEL1 7 | CHSEL1 6 |
| | | | | | | | | | | | | | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CHSEL1 5 | CHSEL1 4 | CHSEL1 3 | CHSEL1 2 | CHSEL1 1 | CHSEL1 0 | CHSEL9 | CHSEL8 | CHSEL7 | CHSEL6 | CHSEL5 | CHSEL4 | CHSEL3 | CHSEL2 | CHSEL1 | CHSEL0 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:19 予約済みであり、リセット値のままにしておかなければなりません。

ビット 18:0 **CHSELx** : チャンネル x 選択

これらのビットはソフトウェアによって書き込まれ、変換されるチャンネルのシーケンスの一部になるチャンネルを定義します。

0 : 入力チャンネル x は変換対象として選択されません。

1 : 入力チャンネル x は変換対象として選択されます。

注 : ソフトウェアは、**ADSTART=0 (変換が実行中でない)** のときのみ、これらのビットを書き込むことができます。

14.13.9 ADC データレジスタ (ADC_DR)

アドレスオフセット : 0x40

リセット値 : 0x0000 0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **DATA[15:0]** : 変換データ

これらのビットは読み出し専用です。これらは、最後の変換チャンネルの変換結果を含んでいます。[図 44 : データの配置と分解能 \(オーバーサンプリング無効 : OVSE = 0\) \(281 ページ\)](#) に示すように、データは左詰めまたは右詰めされています。

較正の完了直後、DATA[6:0] は較正係数を含んでいます。

14.13.10 ADC 較正係数（ADC_CALFACT）

アドレスオフセット：0xB4

リセット値：0x0000 0000

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

ビット 31:7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6:0 **CALFACT[6:0]**：較正係数

- これらのビットは、ハードウェアまたはソフトウェアによって書き込まれます。
- シングルエンド入力較正が完了すると、ハードウェアによって較正係数で更新されます。
 - ソフトウェアは、これらのビットに新しい較正係数を書き込むことができます。新しい較正係数がアナログ ADC に格納されている現在のものと異なる場合は、新しいシングルエンド構成の起動時に適用されます。
 - 較正の完了直後、DATA[6:0] は較正係数を含んでいます。

注： ソフトウェアは、**ADEN=1** かつ **ADSTART=0** のときのみ（ADC が有効であり、較正中でなく、変換中ではない）、これらのビットを書き込むことができます。

14.13.11 ADC 共通設定レジスタ (ADC_CCR)

アドレスオフセット : 0x308

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-------|---------|-------|---------|------------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | LFMEN | VLCD EN | TS EN | VREF EN | PRESC[3:0] | | | | Res. | Res. |
| | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |

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

ビット 25 **LFMEN** : 低周波数モードイネーブル

このビットは、低周波数モードを有効／無効にするために、ソフトウェアによってセット／クリアされます。

2.8 MHz よりも低い ADC クロック周波数を選択する場合は、このモードを有効にする必要があります。

0 : 低周波数モードは無効です。

1 : 低周波数モードは有効です。

注 : ソフトウェアは、**ADSTART=0** (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 24 **VLCDEN** : V_{LCD} 有効

このビットは、 V_{LCD} 読み出し回路を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : V_{LCD} 読み出し回路は無効です。

1 : V_{LCD} 読み出し回路は有効です。

注 : ソフトウェアは、**ADSTART=0** (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 23 **TSEN** : 温度センサ有効

このビットは、温度センサを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : 温度センサは無効です。

1 : 温度センサは有効です。

注 : ソフトウェアは、**ADSTART=0** (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 22 **VREFEN** : V_{REFINT} 有効

このビットは、 V_{REFINT} を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : V_{REFINT} は無効です。

1 : V_{REFINT} は有効です。

注 : ソフトウェアは、**ADSTART=0** (変換が実行中でない) ときのみ、このビットに書き込むことができます。

ビット 21:18 **PRESC[3:0]** : ADC プリスケーラ

ADC へのクロックの周波数を選択するために、ソフトウェアによってセット／クリアされます。クロックはすべての ADC に共通です。

0000 : ADC 入力クロックは分周されません。

0001 : ADC 入力クロックが 2 分周されます。

0010 : ADC 入力クロックが 4 分周されます。

0011 : ADC 入力クロックが 6 分周されます。

0100 : ADC 入力クロックが 8 分周されます。

0101 : ADC 入力クロックが 10 分周されます。

0110 : ADC 入力クロックが 12 分周されます。

0111 : ADC 入力クロックが 16 分周されます。

1000 : ADC 入力クロックが 32 分周されます。

1001 : ADC 入力クロックが 64 分周されます。

1010 : ADC 入力クロックが 128 分周されます。

1011 : ADC 入力クロックが 256 分周されます。

その他 : 予約済み

注 : ソフトウェアは、ADC が無効のときだけ (**ADCAL=0**、**ADSTART=0**、**ADSTP=0**、**ADDIS=0**、および **ADEN=0**)、これらのビットを書き込みことができます。

ビット 17:0 予約済みであり、リセット値のままにしておかなければなりません。

14.13.12 ADC レジスタマップ

次の表は ADC レジスタの一覧です。

表 58. ADC レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|--------------------------------------|-------------|-------------|------------|------|----------|----------|------|-------|--------|-------|--------|------------|------|------|---------|---------|---------|------------|---------|---------|----------|---------|------------|--------|------------|--------------|--------------|------------|-----------|-----------|---------|--------|---------|---------|
| 0x00 | ADC_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EOCAL | Res. | Res. | Res. | Res. | AWD | Res. | Res. | OVR | EOSEQ | EOC | EOSMP | ADRDY |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | | | 0 | | 0 | 0 | 0 | |
| 0x04 | ADC_IER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EOCALIE | Res. | Res. | Res. | Res. | AWDIE | Res. | Res. | OVRIE | EOSEQIE | EOCIE | EOSMPIE | ADRDYIE |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | | | 0 | | 0 | 0 | 0 | |
| 0x08 | ADC_CR | ADCAL | Res. | | ADVREGEN | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADSTP | Res. | ADSTART | ADDIS | ADEN | |
| | リセット値 | 0 | | | 0 | | | | | | | | | | | | | | | | | 0 | | | | | | | 0 | | 0 | 0 | 0 | |
| 0x0C | ADC_CFGR1 | Res. | AWDCH[4:0] | | | | | | | AWDEN | AWDSGL | Res. | Res. | Res. | Res. | Res. | DISCEN | AUTOFF | WAIT | CONT | OVRMOD | Res. | EXTEN[1:0] | Res. | Res. | EXTSEL [2:0] | | ALIGN | RES [1:0] | SCANDIR | DMACFG | DMAEN | | |
| | リセット値 | | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x10 | ADC_CFGR2 | CKMODE[1:0] | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TOVS | OVSS [3:0] | | | OVSR [2:0] | | | Res. | OVSE | | |
| | リセット値 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | |
| 0x14 | ADC_SMPR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SMP [2:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | |
| 0x18 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x1C | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x20 | ADC_TR | Res. | Res. | Res. | Res. | HT[11:0] | | | | | | | | | | | Res. | Res. | Res. | Res. | LT[11:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x24 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x28 | ADC_CHSELR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CHSEL18 | CHSEL17 | CHSEL16 | CHSEL15 | CHSEL14 | CHSEL13 | CHSEL12 | CHSEL11 | CHSEL10 | CHSEL9 | CHSEL8 | CHSEL7 | CHSEL6 | CHSEL5 | CHSEL4 | CHSEL3 | CHSEL2 | CHSEL1 | CHSEL0 | |
| | リセット値 | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x2C 0x30 0x34 0x38 0x3C | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x40 | ADC_DR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DATA[15:0] | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x44 ... 0xB0 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0xB4 | ADC_CALFACT | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CALFACT[6:0] | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0xB8 ... 0x304 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x308 | ADC_CCR | Res. | Res. | Res. | Res. | Res. | | LFMEN | VLODEN | TSEN | VREFEN | PRESC[3:0] | | | | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | | | | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

15 D/A コンバータ (DAC)

15.1 概要

DAC モジュールは、12 ビットの電圧出力デジタルアナログコンバータです。DAC は、8 または 12 ビットモードで設定でき、DMA コントローラと組み合わせて使用することもできます。12 ビットモードでは、データを左詰め右詰めのものにも配置できます。入力基準電圧、 V_{REF+} (ADC と共用) が使用できます。より高い電流駆動力を得るために、任意で出力をバッファすることができます。

15.2 DAC1 の主な機能

デバイスは、12 ビット DAC チャンネルの DAC_OUT1 を 1 個実装しています。

DAC1 の主な機能を以下に示します。

- 1 個のデータ保持レジスタを内蔵
- 12 ビットモードでのデータの左詰めまたは右詰め
- 同期更新機能
- ノイズ波生成
- 三角波生成
- DMA 機能 (アンダーラン検出を含む)
- 変換外部トリガ
- 入力基準電圧、 V_{DDA}

[図 54](#) は DAC チャンネルのブロック図を、[表 59](#) はピンの概要を示します。

図 54. DAC ブロック図

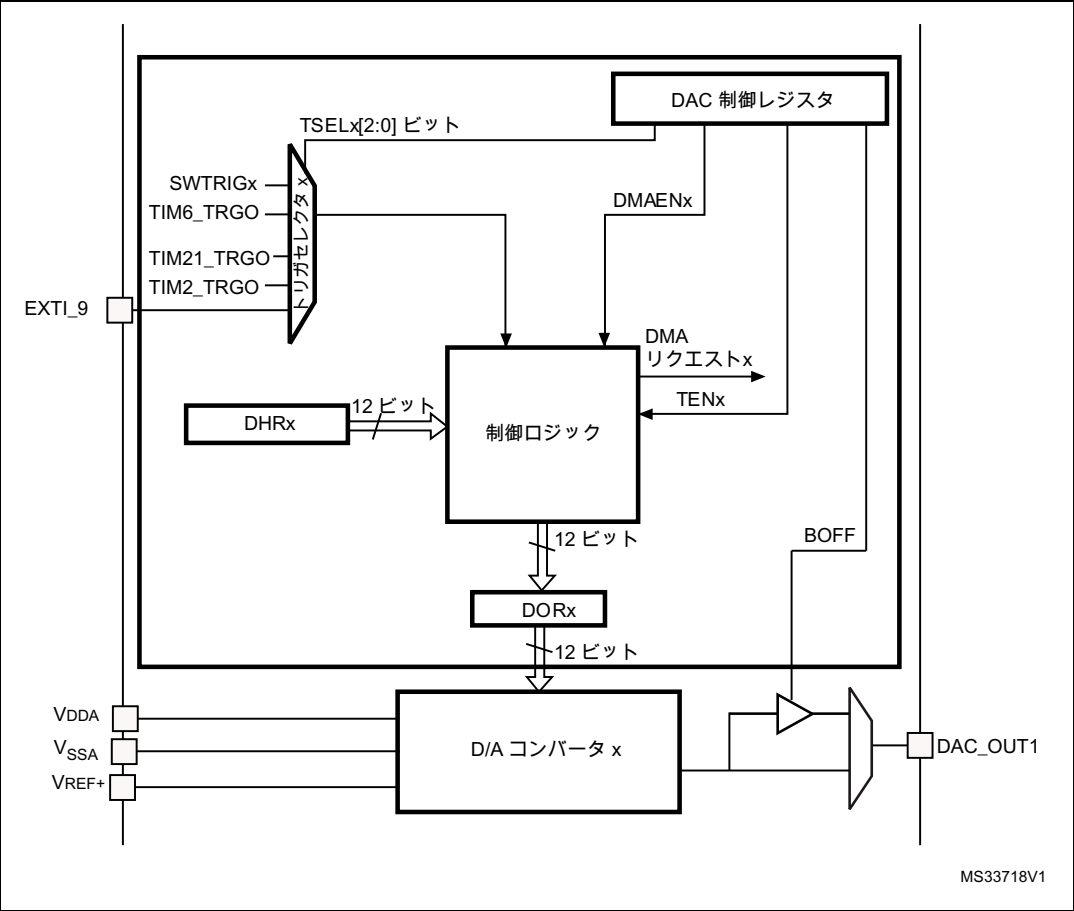


表 59. DAC ピン

| 名称 | 信号タイプ | 摘要 |
|-------------------|---------------|--------------------|
| V _{DDA} | 入力、アナログ電源供給 | アナログ電源供給 |
| V _{SSA} | 入力、アナログ供給グランド | アナログ電源供給のグランド |
| V _{REF+} | 入力、アナログ基準電圧正 | DAC1 のより高い／正の基準電圧 |
| DAC_OUT1 | アナログ出力信号 | DAC チャンネル x アナログ出力 |

注: DAC チャンネル x が有効になると、対応する GPIO ピン (PA4) が自動的にアナログコンバータ出力 (DAC_OUT1) に接続されます。寄生消費を防ぐために、PA4 ピンはまずアナログ (AIN) として設定してください。

15.3 シングルモードの機能説明

15.3.1 DAC チャネルイネーブル

DAC チャネルは、DAC_CR レジスタの EN1 ビットをセットすることによって起動できます。DAC チャネルは、スタートアップ時間 t_{WAKEUP} 後に有効になります。

注： **ENx ビットは、アナログ DAC チャネル x のマクロセルのみを有効にします。DAC チャネル x デジタルインタフェースは、ENx ビットがリセットされた場合でも有効になります。**

15.3.2 DAC 出力バッファイネーブル

DAC は、出力インピーダンスを減らし、外部動作アンプがなくても外部負荷を直接駆動できる 1 つの出力バッファを内蔵しています。DAC チャネル 出力バッファは、DAC_CR レジスタの BOFF1 ビットを使用して有効／無効にできます。

15.3.3 DAC データフォーマット

以下に示すように、選択された設定モードに応じて、指定されたレジスタにデータを書き込む必要があります。

- この場合、次の 3 つの可能性があります。
 - － 8 ビット右詰め：データは、DAC_DHR8Rx [7:0] ビット (DHRx [11:4] ビットに格納) にソフトウェアによってロードされること。
 - － 12 ビット左詰め：データは、DAC_DHR12Lx [15:4] ビット (DHRx [11:0] ビットに格納) にソフトウェアによってロードされること。
 - － 12 ビット右詰め：データは、DAC_DHR12Rx [11:0] ビット (DHRx [11:0] ビットに格納) にソフトウェアによってロードされること。

ユーザによって書き込まれたデータは、ロードされた DAC_DHRyyyx レジスタに応じて、シフトされてから、DHRx (メモリマップされない内部レジスタであるデータ保持レジスタ x) に格納されます。その後、DHRx レジスタは自動的に、ソフトウェアトリガによって、または外部イベントトリガによって、DORx レジスタにロードされます。

図 55. シングル DAC チャネルモードのデータレジスタ



15.3.4 DAC チャネル変換

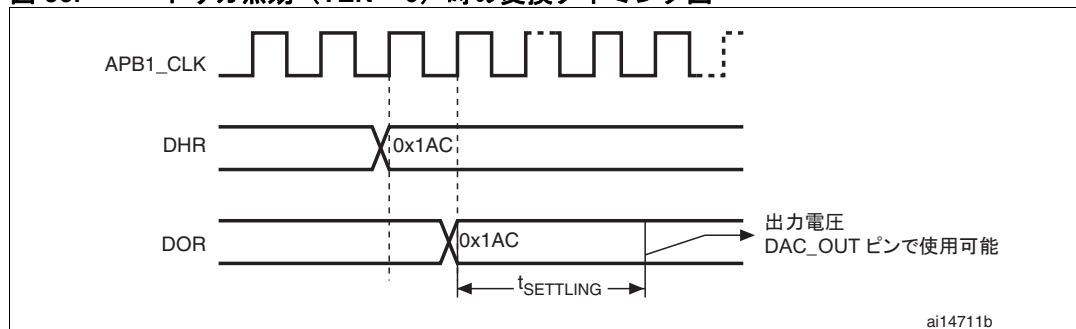
DAC_DORx に直接書き込むことはできませんので、DAC_DHRx レジスタをロードする (DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx への書き込み) ことによって、DAC チャネル x へのデータ転送を行う必要があります。

DAC_DHRx レジスタに格納されたデータは、ハードウェアトリガが選択されていない (DAC_CR レジスタの TENx ビットがリセットされている) 場合に、1 APB1 クロックサイクル後に DAC_DORx レジスタに自動的に転送されます。ただし、ハードウェアトリガが選択されている (DAC_CR レジ

スタの TENCx ビットがセットされている) ときには、トリガが発生すると、転送は 3 PCLK1 クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx の内容がロードされると、電源電圧とアナログ出力負荷に応じて決定される t_{SETTLING} 時間後にアナログ出力電圧が使用可能になります。

図 56. トリガ無効 (TEN = 0) 時の変換タイミング図



1 つの LFSR 生成による独立トリガ

DAC をこの変換モード ([セクション 15.4: ノイズ生成](#)を参照) に設定するには、次の手順が必要です。

1. DAC チャンネルトリガイネーブルビット TENCx をセットします。
2. TSELx[2:0] ビットをセットすることによって、トリガソースを設定します。
3. DAC チャンネル WAVEx[1:0] ビットを“01”に設定し、MAMPx[3:0] ビットで同じ LFSR マスク値を設定します。
4. 目的の DAC_DHRx レジスタ (DHR12RD、DHR12LD、または DHR8RD) に、DAC チャンネルデータをロードします。

DAC チャンネル x トリガが発生すると、同じマスクを持つ LFSRx カウンタが DHRx レジスタに加算され、合計が DAC_DORx に転送されます (3 APB クロックサイクル後)。その後、LFSR x カウンタが更新されます。

1 つの三角波生成による独立トリガ

DAC をこの変換モード ([セクション 15.5: 三角波生成](#)を参照) に設定するには、次の手順が必要です。

1. DAC チャンネル x トリガイネーブルビット TENCx をセットします。
2. TSELx[2:0] ビットをセットすることによって、トリガソースを設定します。
3. DAC チャンネル x の WAVEx[1:0] ビットを“1x”に設定し、MAMPx[3:0] ビットで同じ最大振幅値を設定します。
4. DAC チャンネル x データを、目的の DAC_DHRx レジスタ (DHR12RD、DHR12LD、または DHR8RD) に、ロードします。

DAC チャンネル x トリガが発生すると、同じ三角波振幅を持つ DAC チャンネル x の三角波カウンタが DHRx レジスタに加算され、合計が DAC_DORx に転送されます (3 APB クロックサイクル後)。その後、DAC チャンネル x の三角波カウンタが更新されます。

15.3.5 DAC 出力電圧

デジタル入力は、0 から $V_{\text{REF+}}$ までのリニア変換で出力電圧に変換されます。

各 DAC チャンネルピンのアナログ出力電圧は、次の式によって求められます。

$$\text{DACoutput} = V_{\text{REF+}} \times \frac{\text{DOR}}{4095}$$



15.3.6 DAC トリガ選択

TENx 制御ビットがセットされている場合、外部イベント（タイマカウンタ、外部割り込みラインなど）によって変換をトリガできます。[表 60](#) に示すように、8 つの考えられるイベントのうち、どのイベントが変換をトリガするかは TSELx[2:0] 制御ビットによって決まります。

表 60. 外部トリガ

| 転送元 | タイプ | TSEL[2:0] |
|-----------------|-----------------|-----------|
| TIM6 TRGO イベント | オンチップタイマからの内部信号 | 000 |
| 予約済み | | 001 |
| 予約済み | | 010 |
| TIM21 TRGO イベント | | 011 |
| TIM2 TRGO イベント | | 100 |
| 予約済み | | 101 |
| EXTI ライン 9 | 外部ピン | 110 |
| SWTRIG | ソフトウェア制御ビット | 111 |

DAC インタフェースが選択されたタイマ TRGO 出力または選択された外部割り込みライン 9 の立ち上がりエッジを検出するたびに、DAC_DHRx レジスタに最後に格納されたデータが DAC_DORx レジスタに転送されます。DAC_DORx レジスタは、トリガが発生してから 3 APB1 サイクル後に更新されます。

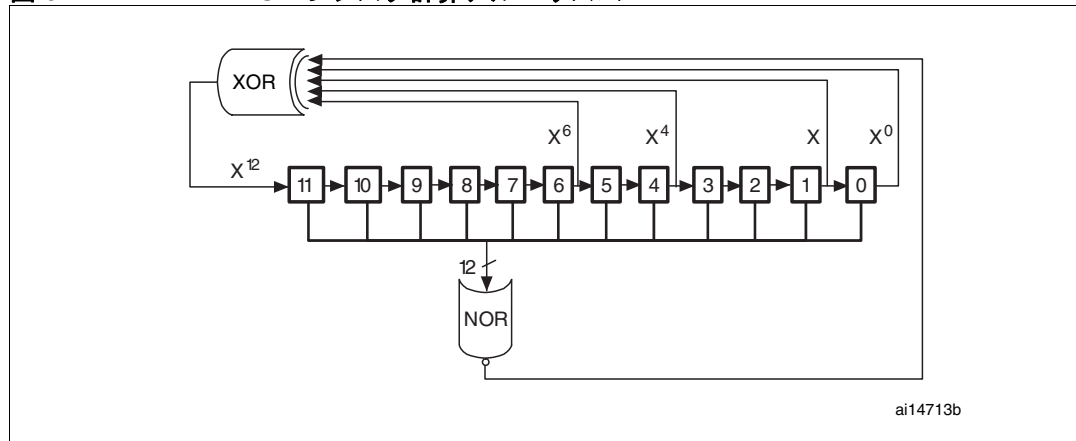
ソフトウェアトリガが選択されている場合、変換は、SWTRIG ビットがセットされると開始されます。SWTRIG ビットは、DAC_DHRx レジスタの内容が DAC_DORx レジスタにロードされると、ハードウェアによってリセットされます。

注： *ENx ビットがセットされているときには、TSELx[2:0] ビットを変更することはできません。ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DORx レジスタへの転送は、わずか 1 APB1 クロックサイクルで行われます。*

15.4 ノイズ生成

リニアフィードバックシフトレジスタ (LFSR) を使用して、可変振幅の擬似ノイズを生成することができます。DAC ノイズ生成を選択するには、WAVEx[1:0] に“01”をセットします。LFSRにプリロードされる値は 0xAAA です。このレジスタは、各トリガイイベントの 3 APB クロックサイクル後に、特定の計算アルゴリズムに従って更新されます。

図 57. DAC LFSR レジスタ計算アルゴリズム

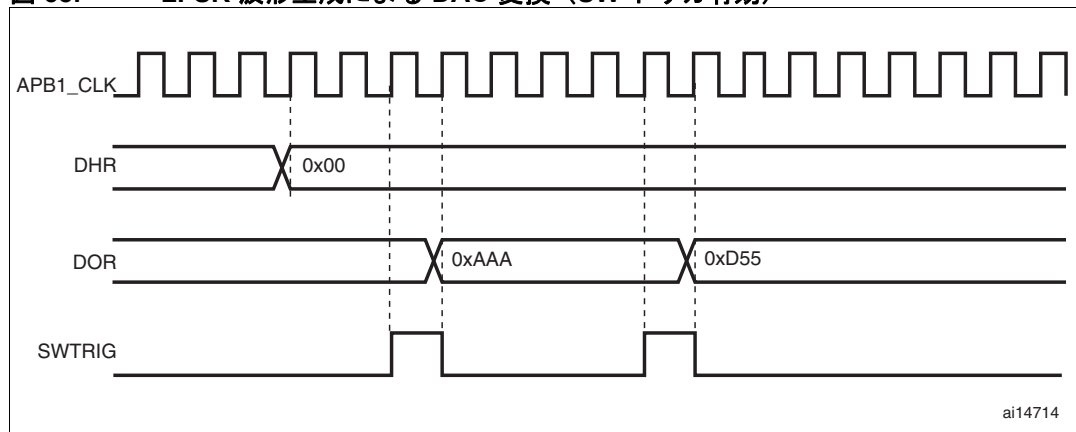


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算され、DAC_DORx レジスタに格納されます。

LFSR が 0x0000 の場合、“1”がインジェクトされます (アンチロックアップメカニズム)。

WAVEx[1:0] ビットをリセットすることによって、LFSR 波形生成をリセットできます。

図 58. LFSR 波形生成による DAC 変換 (SW トリガ有効)



注： ノイズ生成のためには、DAC_CR レジスタの TENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。

15.5 三角波生成

DC または低周波数信号上に、小さな振幅の三角波を追加することが可能です。DAC 三角波生成を選択するには、WAVEx[1:0] を“10”にセットします。振幅は、DAC_CR レジスタのMAMPx[3:0] ビットを介して設定されます。内部三角波カウンタは、各トリガイイベントの 3 APB クロックサイクル後にインクリメントされます。このカウンタの値は、オーバーフローなしに DAC_DHRx レジスタに加えられ、合計は DAC_DORx レジスタに格納されます。三角波カウンタは、MAMPx[3:0] ビットによって定義された最大振幅以上になるまでインクリメントされます。設定された振幅に達すると、カウンタは 0 にデクリメントされ、再びインクリメントが開始されます。

WAVEx[1:0] ビットをリセットすることによって、三角波生成をリセットすることができます。

図 59. DAC 三角波生成

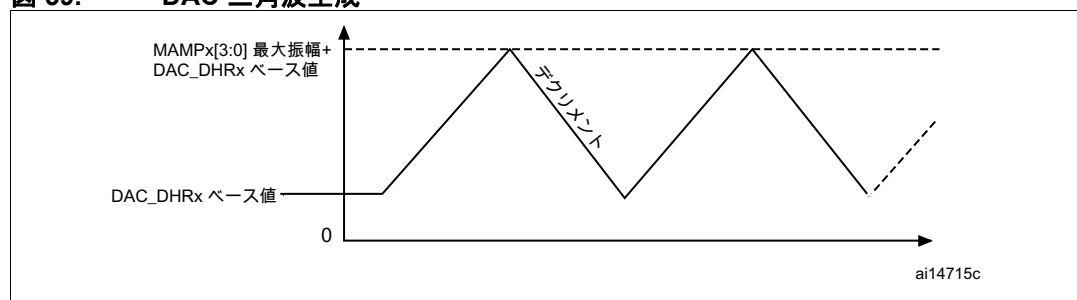
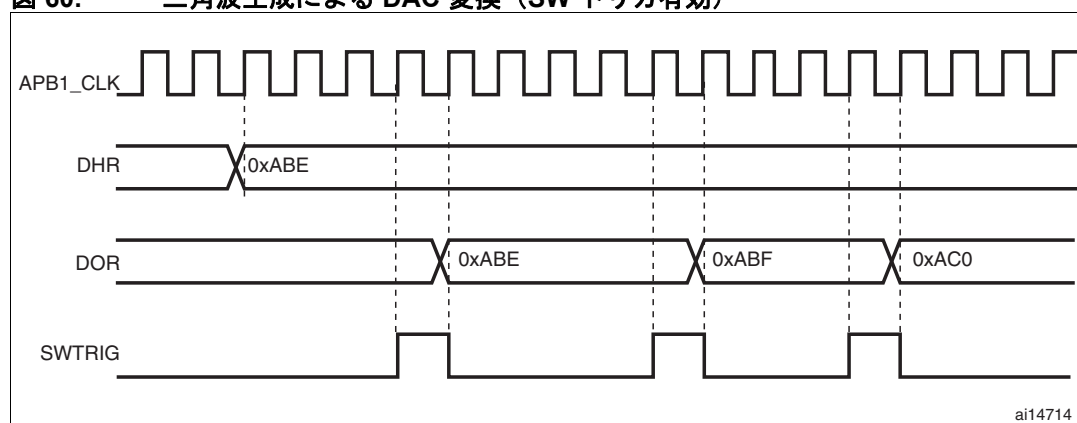


図 60. 三角波生成による DAC 変換 (SW トリガ有効)



注： 三角波生成のためには、DAC_CR レジスタのTENx ビットをセットすることによって、DAC トリガを有効にしなければなりません。

DAC を有効にするには、その前に MAMPx[3:0] ビットを設定する必要があります。そうしないと、これらのビットは変更できません。

15.6 DMA リクエスト

各 DAC チャンネルは、DMA 機能を備えています。DAC チャンネルの DMA リクエストは、2 つの DMA チャンネルを使用して処理されます。

DAC DMA リクエストは、DMAENx ビットがセットされているときに、外部トリガ（ソフトウェアトリガでなく）が発生したときに生成されます。その場合、DAC_DHRx レジスタの値が DAC_DORx レジスタに転送されます。

DMA アンダーラン

DAC DMA リクエストはキューされないで、最初の外部トリガに対する確認応答が受信される（最初のリクエスト）前に 2 番目の外部トリガが発生すると、新しいリクエストは発行されず、DAC_SR レジスタの DMA チャンネル x アンダーランフラグ DMAUDRx がセットされてエラー状態を報告します。続いて DMA データ転送が無効になり、その後の DMA リクエストは処理されません。DAC チャンネル x は、古いデータを変換し続けます。

ソフトウェアでは、“1”を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし、DMA と DAC のチャンネル x を再初期化して転送を正しくリスタートさせてください。また、DAC トリガ変換周波数を変更するか、またはDMA の負荷を軽減して、新しいDMA アンダーランを回避してください。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャンネル では、DAC_CR レジスタの対応する DMAUDRIEx ビットが有効にされた場合、割り込みも生成されます。

15.7 DAC レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスする必要があります。

15.7.1 DAC 制御レジスタ (DAC_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|-----------|--------|------------|----|----|----|------------|----|------------|----|----|------|-------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | | | | Res. | | Res. | | | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | DMAUDRIE1 | DMAEN1 | MAMP1[3:0] | | | | WAVE1[1:0] | | TSEL1[2:0] | | | TEN1 | BOFF1 | EN1 |
| | | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

- ビット 31:14 予約済みであり、リセット値に保持する必要があります。
- ビット 13 **DMAUDRIE1** : DAC チャンネル 1 DMA アンダーラン割り込みイネーブル
このビットはソフトウェアによってセット／クリアされます。
0 : DAC チャンネル 1 DMA アンダーラン割り込みは無効です。
1 : DAC チャンネル 1 DMA アンダーラン割り込みは有効です。
- ビット 12 **DMAEN1** : DAC チャンネル 1 DMA イネーブル
このビットはソフトウェアによってセット／クリアされます。
0 : DAC チャンネル 1 DMA モードは無効です。
1 : DAC チャンネル 1 DMA モードは有効です。

ビット 11:8 MAMP1[3:0] : DAC チャンネル 1 マスク／振幅セクタ

これらのビットは、波形生成モードのマスクまたは三角波生成モードの振幅を選択するために、ソフトウェアによって書き込まれます。

0000 : LFSR／三角波振幅のアンマスクビット 0 は 1 に等しい。

0001 : LFSR／三角波振幅のアンマスクビット [1:0] は 3 に等しい。

0010 : LFSR／三角波振幅のアンマスクビット [2:0] は 7 に等しい。

0011 : LFSR／三角波振幅のアンマスクビット [3:0] は 15 に等しい。

0100 : LFSR／三角波振幅のアンマスクビット [4:0] は 31 に等しい。

0101 : LFSR／三角波振幅のアンマスクビット [5:0] は 63 に等しい。

0110 : LFSR／三角波振幅のアンマスクビット [6:0] は 127 に等しい。

0111 : LFSR／三角波振幅のアンマスクビット [7:0] は 255 に等しい。

1000 : LFSR／三角波振幅のアンマスクビット [8:0] は 511 に等しい。

1001 : LFSR／三角波振幅のアンマスクビット [9:0] は 1023 に等しい。

1010 : LFSR／三角波振幅のアンマスクビット [10:0] は 2047 に等しい。

≥ 1011 : LFSR／三角波振幅のアンマスクビット [11:0] は 4095 に等しい。

ビット 7:6 WAVE1[1:0] : DAC チャンネル 1 ノイズ／三角波生成イネーブル

これらのビットは、ソフトウェアによってセット／クリアされます。

00 : 波形生成は無効です。

01 : ノイズ波生成は有効です。

1x : 三角波生成は有効です。

注： ビット *TEN1* = 1 (DAC チャンネル 1 トリガ有効) の場合のみ使用されます。

ビット 5:3 TSEL1[2:0] : DAC チャンネル 1 トリガ選択

これらのビットは、DAC チャンネル 1 をトリガするために使用される外部イベントを選択します。

000 : タイマ 6 TRGO イベント

001 : 予約済み

010 : 予約済み

011 : タイマ 21 TRGO イベント

100 : タイマ 2 TRGO イベント

101 : 予約済み

110 : EXTI ライン 9

111 : ソフトウェアトリガ

注： ビット *TEN1* = 1 (DAC チャンネル 1 トリガ有効) の場合のみ使用されます。

ビット 2 TEN1 : DAC チャンネル 1 トリガイネーブル

このビットは、DAC チャンネル 1 トリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : DAC チャンネル 1 トリガは無効であり、DAC_DHRx レジスタに書き込まれたデータは、1 APB1 クロックサイクル後に DAC_DOR1 レジスタに転送されます。

1 : DAC チャンネル 1 トリガは有効であり、DAC_DHRx レジスタから転送されたデータは、3 APB1 クロックサイクル後に DAC_DOR1 レジスタに転送されます。

注： ソフトウェアトリガが選択されているときには、DAC_DHRx レジスタから DAC_DOR1 レジスタへの転送は、わずか 1 APB1 クロックサイクルで行われます。

ビット 1 BOFF1 : DAC チャンネル 1 出力バッファディセーブル

このビットは、DAC チャンネル 1 出力バッファを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : DAC チャンネル 1 出力バッファは有効です。

1 : DAC チャンネル 1 出力バッファは無効です。

ビット 0 EN1 : DAC チャンネル 1 イネーブル

このビットは、DAC チャンネル 1 を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : DAC チャンネル 1 は無効です。

1 : DAC チャンネル 1 は有効です。

15.7.2 DAC ソフトウェアトリガレジスタ (DAC_SWTRIGR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SWTRIG1 |
| | | | | | | | | | | | | | | | w |

ビット 31:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **SWTRIG1** : DAC チャンネル 1 ソフトウェアトリガ

このビットは、ソフトウェアトリガを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : ソフトウェアトリガは無効です。

1 : ソフトウェアトリガは有効です。

注 : このビットは、DAC_DHR1 レジスタの値が DAC_DOR1 レジスタにロードされると、ハードウェアによってクリアされます (1 APB1 クロックサイクル後)。

15.7.3 DAC チャンネル 1 の 12 ビット右詰めデータ保持レジスタ (DAC_DHR12R1)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|----------------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | DACC1DHR[11:0] | | | | | | | | | | | |
| | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11: **DACC1DHR[11:0]** : DAC チャンネル 1 の 12 ビット右詰めデータ

これらのビットは、DAC チャンネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

15.7.4 DAC チャンネル 1 の 12 ビット左詰めデータ保持レジスタ (DAC_DHR12L1)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|----------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DACC1DHR[11:0] | | | | | | | | | | | | Res. | Res. | Res. | Res. |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | | | | |

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:4 **DACC1DHR[11:0]** : DAC チャンネル 1 の 12 ビット左詰めデータ

これらのビットは、DAC チャンネル 1 の 12 ビットデータを指定するために、ソフトウェアによって書き込まれます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

15.7.5 DAC チャンネル 1 の 8 ビット右詰めデータ保持レジスタ (DAC_DHR8R1)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|---------------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DACC1DHR[7:0] | | | | | | | |
| | | | | | | | | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **DACC1DHR[7:0]** : DAC チャンネル 1 の 8 ビット右詰めデータ

これらのビットは、DAC チャンネル 1 の 8 ビットデータを指定するために、ソフトウェアによって書き込まれます。

15.7.6 DAC チャンネル 1 データ出力レジスタ (DAC_DOR1)

アドレスオフセット : 0x2C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|----------------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | DACC1DOR[11:0] | | | | | | | | | | | |
| | | | | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **DACC1DOR[11:0]** : DAC チャンネル 1 データ出力
これらのビットは読み出し専用であり、DAC チャンネル 1 のデータ出力を含みます。

15.7.7 **DAC ステータスレジスタ (DAC_SR)**

アドレスオフセット : 0x34

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|---------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | DMAUDR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | rc_w1 | | | | | | | | | | | | | |

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13 **DMAUDR1** : DAC チャンネル 1 の DMA アンダーランフラグ
このビットは、ハードウェアによってセットされ、(1 を書き込むことによって) ソフトウェアによってクリアされます。
0 : DAC チャンネル 1 に DMA アンダーランエラー条件は発生しませんでした。
1 : DAC チャンネル 1 に DMA アンダーランエラー条件が発生しました (現在選択されているトリガは、DMA サービス機能のレートを上回る周波数で DAC チャンネル 1 変換を駆動しています)。

ビット 12:0 予約済みであり、リセット値に保持する必要があります。

15.7.8 DAC レジスタマップ

表 61 に DAC レジスタの要約を示します。

表 61. DAC レジスタマップ とリセット値

| オフ セット | レジスタ名 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
|-----------|-------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---------|--------|----------------|------|------|------|---------------|------|-----------|------|------|------|-------|------|---------|------|--|
| 0x00 | DAC_CR | Res. | Res. | Res. | Res. | Res. | | | | Res. | Res. | | | | Res. | Res. | Res. | Res. | Res. | DMAUDR1 | DMAEN1 | MAMP[3:0] | | | | WAVE[1:0] | | TSEL[2:0] | | TEN1 | | BOFF1 | EN1 | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x04 | DAC_SWTRIGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SWTRIG1 | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | Res. | Res. | | | | | | | | | | | | 0 | | | |
| 0x08 | DAC_DHR12R1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DACC1DHR[11:0] | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x0C | DAC_DHR12L1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DACC1DHR[11:0] | | | | | | | | | | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | |
| 0x10 | DAC_DHR8R1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DACC1DHR[7:0] | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x2C | DAC_DOR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DACC1DOR[11:0] | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x34 | DAC_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DMAUDR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | | | | | | | | | | | | | | | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#)を参照してください。

16 コンパレータ (COMP)

16.1 概要

STM32L0x3 デバイスは、2 個の 超低電力 コンパレータ COMP1、および COMP2 を内蔵しており、それらはスタンバイデバイス（すべての端子を I/O で使用可能）として、またはタイマと組み合わせ使用することができます。コンパレータは以下のようなさまざまな機能に使用できます。

- アナログ信号によってトリガされる低電力モードからのウェイクアップ
- アナログ信号調節
- DAC と タイマからの PWM 出力を組み合わせた場合のサイクルごとの電流制御ループ

16.2 COMP の主な機能

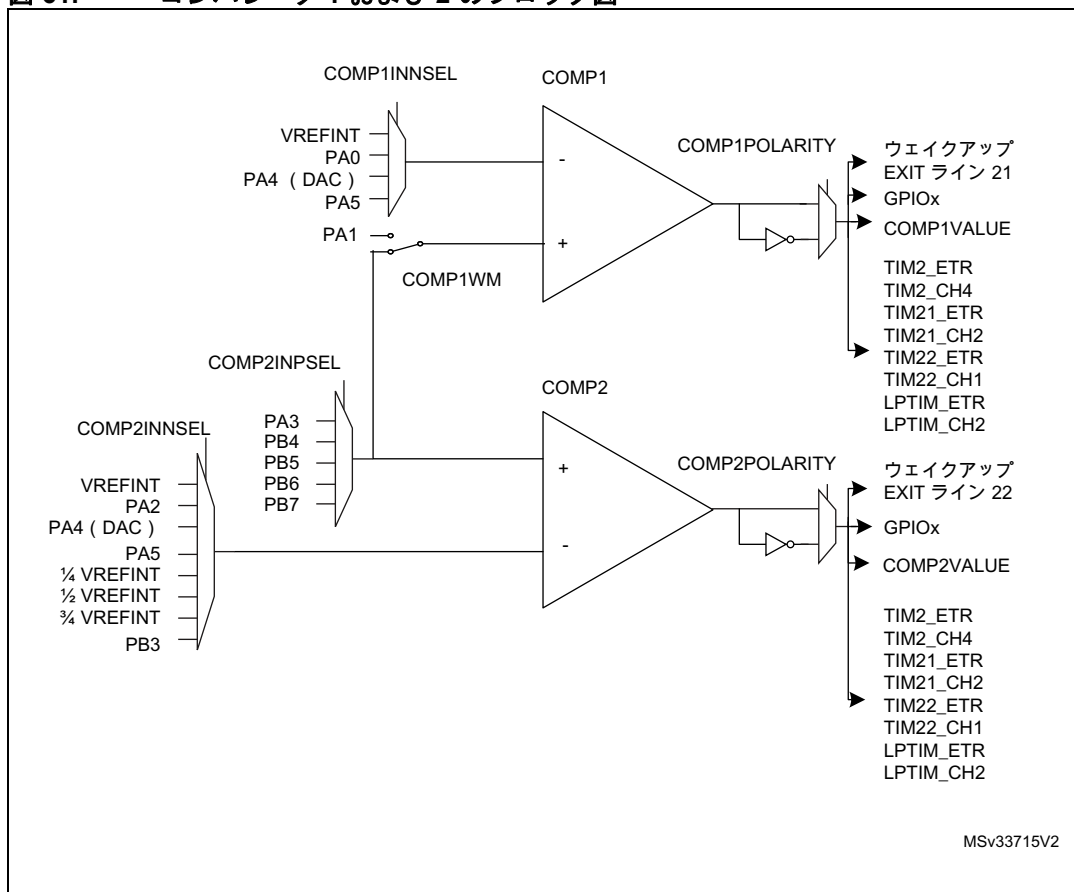
- 超低電力消費型の COMP1 コンパレータ
- レールツーレール入力の高速または低速モードを備えた COMP2 コンパレータ
- 各コンパレータは、電圧を柔軟に選択できるように、次のような正入力および設定可能な負入力を備えています。
 - I/O ピン
 - DAC
 - スケーラ（バッファ付き分圧器）が提供する内部基準電圧および 3 つの約数（1/4、1/2、3/4）
- プログラム可能なスピード／消費（COMP2 のみ）
- 出力先を I/O または トリガに使用する複数のタイマ入力に変更することができます。
 - キャプチャイベント
- COMP1、および COMP2 をウィンドウコンパレータで結合することができます。各コンパレータは、SLEEP および STOP モードからのウェイクアップ（EXTI コントローラ経由）を備えた割り込み生成機能を持っています。

16.3 COMP の機能説明

16.3.1 COMP ブロック図

コンパレータのブロック図を [図 61 : コンパレータ 1 および 2 のブロック図](#) に示します。

図 61. コンパレータ 1 および 2 のブロック図



16.3.2 COMP ピンおよび内部信号

コンパレータ入力として使用される I/O は、GPIO レジスタのアナログモードで設定する必要があります。

コンパレータの出力は、データシートの「オルタネート機能配置」表に示されているオルタネート機能チャンネルを使用して I/O に接続することができます。

出力先を、以下の目的に使用される各種タイマ入力に内部で変更することも可能です。

- タイミング測定のための入力キャプチャ

コンパレータの出力先を内部および外部から一斉に変更することが可能です。

16.3.3 COMP のリセットおよびクロック

クロックコントローラによって提供される COMP クロックは、PCLK (APB クロック) と同期しています。

RCC コントローラには、クロックイネーブル制御ビットは提供されていません。リセットおよびクロックイネーブルビットは COMP と SYSCFG に共通です。

注： **重要：極性選択ロジックおよびポートへの出力先変更は、PCLK クロックから個別に機能します。これにより、コンパレータは STOP モードでも機能することができます。**

16.3.4 コンパレータのロック機構

コンパレータは、過電流保護や熱保護などの安全上の目的で使用されます。特定の機能安全要件があるようなアプリケーションの場合、万が一誤ったレジスタへのアクセスやプログラムカウンタの破壊が起こった場合に、コンパレータのプログラミングが絶対に変更されないようにすることが必要です。

このような目的で、コンパレータの制御レジスタおよびステータスレジスタを書き込み保護することができます (読み出し専用)。

プログラミングが完了したら、COMPxLOCK ビットを"1"にセットすることができます。これにより、COMPxLOCK ビットを含む COMPx_CSR レジスタ全体が読み出し専用になります。

書き込み保護は MCU リセットによってのみリセット可能です。

16.4 電力モード

COMP2 の電力消費と伝搬遅延を調節して、特定のアプリケーションに最適なトレードオフを実現させることができます。

COMP2_CSR レジスタの COMP2_SPEED ビットは、スピード／消費を増やすか、または減らすようにプログラムすることができます。

16.5 割り込み

コンパレータの出力は拡張割り込み／イベントコントローラに内部的に接続されます。各コンパレータには専用の EXTI ラインがあり、割り込みまたはイベントを生成することができます。低電力モードを終了するときにも同じ方法が使用されます。

詳細については、割り込みおよびイベントのセクションを参照してください。

16.6 COMP レジスタ

16.6.1 コンパレータ 1 制御/ステータスレジスタ (COMP1_CSR)

COMP1_CSR は、コンパレータ 1 の制御/ステータスレジスタです。このレジスタは、コンパレータ 1 に関するすべてのビット/フラグを格納しています。

アドレスオフセット : 0x18

システムリセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|----------------|-------------|------|------|------|------|------|----------|------|------|--------------|------|------|------|----------|------|
| COMP1 LOCK | COMP1 VALUE | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| rs | r | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| COMP1 POLARITY | Res. | Res. | Res. | Res. | Res. | Res. | COMP1 WM | Res. | Res. | COMP1INN SEL | Res. | Res. | Res. | COMP1 EN | |
| rw | | | | | | | rw | | | rw | rw | | | | rw |

ビット 31 **COMP1LOCK** : COMP1_CSR レジスタロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、コンパレータ 1 制御レジスタ COMP1_CSR[31:0] の内容全体をロックします。

0 : コンパレータ 1 の COMP1_CSR[31:0] ビットは読み出し/書き込み用です。

1 : コンパレータ 1 の COMP1_CSR[31:0] ビットは読み出し専用です。

ビット 30 **COMP1VALUE** : コンパレータ 1 出力ステータスビット

このビットは読み出し専用です。このビットは、COMP1POLARITY ビットの影響を考慮して現在のコンパレータ 1 出力を反映します。

ビット 29:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **COMP1POLARITY** : コンパレータ 1 極性選択ビット

このビットは、ソフトウェアによってセット/クリアされます (COMP1LOCK がセットされていない場合のみ)。このビットはコンパレータ 1 の極性を反転させます。

0 : コンパレータ 1 の出力値は反転されません。

1 : コンパレータ 1 の出力値は反転されます。

ビット 14:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **COMP1WM** : コンパレータ 1 ウィンドウモード選択ビット

このビットは、ソフトウェアによってセット/クリアされます (COMP1LOCK がセットされていない場合のみ)。このビットは、2 つのコンパレータの正入力相互接続されている場合に、コンパレータ 1 ウィンドウモードを選択します。

0 : コンパレータ 1 の正入力は PA1 に接続されています。

1 : コンパレータ 1 の正入力はコンパレータ 2 の正入力により短絡しています (COMP1_CSR を参照)。

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **COMP1INSEL** : コンパレータ 1 負入力接続設定ビット

これらのビットは、ソフトウェアによってセット/クリアされます (COMP1LOCK がセットされていない場合のみ)。これらのビットは、コンパレータ 1 の負入力に接続される入力を選択します。

00 : VREFINT
01 : PA0
10 : DAC 1/PA4
11 : PA5

ビット 3:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **COMP1EN** : コンパレータ 1 イネーブルビット

このビットは、ソフトウェアによってセット/クリアされます (COMP1LOCK がセットされていない場合のみ)。このビットはコンパレータ 1 をスイッチオンします。

0 : コンパレータ 1 はスイッチオフされています。
1 : コンパレータ 1 はスイッチオンされています。

16.6.2 コンパレータ 2 制御/ステータスレジスタ (COMP2_CSR)

COMP2_CSR は、コンパレータ 2 の制御/ステータスレジスタです。このレジスタは、コンパレータ 2 に関するすべてのビット/フラグを格納しています。

アドレスオフセット : 0x1C

システムリセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|----------------|-------------|------|------|------|-------------|------|------|------|-------------|------|------|-------------|------|------|----------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| COMP2 LOCK | COMP2 VALUE | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| rs | r | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| COMP2 POLARITY | Res. | Res. | Res. | Res. | COMP2INPSEL | | | Res. | COMP2INNSEL | | | COMP2 SPEED | Res. | Res. | COMP2 EN |
| rw | | | | | rw | rw | rw | | rw | rw | rw | rw | | | rw |

ビット 31 **COMP2LOCK** : COMP2_CSR レジスタロックビット

このビットは、ソフトウェアでセットされ、ハードウェアシステムリセットでクリアされます。このビットは、コンパレータ 2 制御レジスタ COMP2_CSR[31:0] の内容全体をロックします。

0 : コンパレータ 2 の COMP2_CSR[31:0] ビットは読み出し/書き込み用です。
1 : コンパレータ 2 の COMP2_CSR[31:0] ビットは読み出し専用です。

ビット 30 **COMP2VALUE** : コンパレータ 2 出力ステータスビット

このビットは読み出し専用です。このビットは、COMP2POLARITY ビットの影響を考慮して現在のコンパレータ 2 出力を反映します。

ビット 29:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **COMP2POLARITY** : コンパレータ 2 極性選択ビット

このビットは、ソフトウェアによってセット/クリアされます (COMP2LOCK がセットされていない場合のみ)。このビットはコンパレータ 2 の極性を反転させます。

0 : コンパレータ 2 の出力値は反転されません。
1 : コンパレータ 2 の出力値は反転されます。

ビット 14:11 予約済みであり、リセット値に保持する必要があります。

ビット 10:8 COMP2INSEL : コンパレータ 2 正入力接続設定ビット

これらのビットは、ソフトウェアによってセット／クリアされます (COMP2LOCK がセットされていない場合のみ)。これらのビットは、コンパレータ 2 の正入力に接続される入力を選択します。

000 : PA3
001 : PB4
010 : PB5
011 : PB6
100 : PB7
101 : PB7
110 : PB7
111 : PB7

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 COMP2INNSEL : コンパレータ 2 負入力接続設定ビット

これらのビットは、ソフトウェアによってセット／クリアされます (COMP2LOCK がセットされていない場合のみ)。これらのビットは、コンパレータ 2 の負入力に接続される入力を選択します。

000 : VREFINT
001 : PA2
010 : DAC 1/PA4
011 : PA5
100 : 1/4 VREFINT
101 : 1/2 VREFINT
110 : 3/4 VREFINT
111 : PB3

ビット 3 COMP2SPEED : コンパレータ 2 電力モード選択ビット

このビットは、ソフトウェアによってセット／クリアされます (COMP2LOCK がセットされていない場合のみ)。このビットは、コンパレータ 2 の電力モードを選択します。

0 : 低速
1 : 高速

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 0 COMP2EN : コンパレータ 2 イネーブルビット

このビットは、ソフトウェアによってセット／クリアされます (COMP2LOCK がセットされていない場合のみ)。このビットはコンパレータ 2 をスイッチオンします。

0 : コンパレータ 2 はスイッチオフされています。
1 : コンパレータ 2 はスイッチオンされています。

16.6.3 COMP レジスタマップ

次の表に コンパレータレジスタの一覧を示します。

コンパレータレジスタは、SYS_CFG ペリフェラルレジスタのベースアドレスを共用しています。

表 62. COMP レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-----------|------------|-------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|----------------|------|------|------|------|-------------|------|----------|--------------|------|--------------|---|------|------|---------|---------|
| 0x18 | COMP1_CSR | COMP1LOCK. | COMP1VALUE. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | COMP1POLARITY. | Res. | Res. | Res. | Res. | Res. | Res. | COMP1WM. | Res. | Res. | COMP1INNSSEL | | Res. | Res. | Res. | COMP1EN |
| | リセット値 | 0 | 0 | | | | | | | | | | | | | | | 0 | | | | | | | 0 | | | 0 | 0 | | | | 0 |
| 0x1C | COMP2_CSR | COMP2LOCK. | COMP2VALUE. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | COMP2POLARITY. | Res. | Res. | Res. | Res. | COMP2INPSEL | | Res. | COMP2INNSSEL | | COMP2SPEED | | Res. | Res. | COMP2EN | |
| | リセット値 | 0 | 0 | | | | | | | | | | | | | | | 0 | | | | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | | | | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。



17 液晶ディスプレイコントローラ (LCD)

17.1 概要

LCD コントローラは、最大 8 つのコモン端子と最大 32 のセグメント端子を持つモノクロ受動液晶ディスプレイ (LCD) のデジタルコントローラ/ドライバであり、128 (4x32) または 224 (8x28) LCD 画素 (ピクセル) を駆動します。端子の正確な数は、データシートに示されているように、デバイスのピン数に依存します。

LCD は、可視または不可視に調整できる複数のセグメント (ピクセルまたは完全なシンボル) で構成されます。各セグメントは、2 つの電極の間に揃えられた液晶分子の層で構成されます。閾値電圧より大きな電圧が液晶にかけられると、セグメントは可視状態になります。液晶への電気泳動効果 (ディスプレイを劣化させる) を避けるには、セグメント電圧は交流である必要があります。セグメント間の波形は、直流 (DC) にならないように生成される必要があります。

用語集

バイアス : LCD を駆動するときに使用される電圧レベルの数。1/ (LCD ディスプレイの駆動に使用される電圧レベルの数 - 1) として定義されます。

増幅回路 : コントラストコントローラ回路

コモン : 複数のセグメント (32セグメント) に接続される共通端子。

デューティ比 : 1/ (特定の LCD ディスプレイのコモン端子の数) として定義される数。

フレーム : セグメントに書き込まれる波形の 1 周期。

フレームレート : 1 秒間のフレーム数。すなわち、LCD セグメントが 1 秒間に駆動される回数。

LCD : (Liquid Crystal Display : 液晶ディスプレイ) セグメントに直接つながる端子を持つ受動ディスプレイパネル。

セグメント : 最小表示要素 (LCD ディスプレイ上に文字を作り出すための 1 つのバーまたはドット)。

17.2 LCD の主な機能

- 柔軟性の高いフレームレート制御。
- スタティック、1/2、1/3、1/4、および 1/8 デューティをサポートします。
- スタティック、1/2、1/3、および 1/4 バイアスをサポートします。
- ダブルバッファメモリにより、LCD_RAM レジスタ内のデータをアプリケーションファームウェアによっていつでも更新でき、表示されるデータの整合性に影響を与えません。
 - ピクセル情報 (アクティブ/非アクティブ) を含む最大 16 x 32 ビットレジスタの LCD データ RAM。
- V_{LCDmin} から V_{LCDmax} までソフトウェアで選択可能な LCD 出力電圧 (コントラスト)。
- 外部アナログコンポーネントは不要です。
 - ステップアップコンバータが組み込まれていて、 V_{DD} より高い内部 V_{LCD} 電圧を生成します。
 - 外部と内部の V_{LCD} 電圧ソースをソフトウェアで選択。外部ソースの場合、内部増幅回路は無効になり、消費電力を削減します。
 - 抵抗ネットワークが組み込まれていて、中間 V_{LCD} 電圧を生成します。
 - 抵抗ネットワークの構造はソフトウェアによって設定可能であり、LCD パネルに必要な容量電荷に合わせて消費電力を調整できます。
- コントラストは、2 つの方法で調整できます。
 - 内部ステップアップコンバータを使用しているときには、ソフトウェアによって V_{LCD} を V_{LCDmin} と V_{LCDmax} の間で調整できます。
 - フレーム間のデッドタイムをプログラム可能 (最大 8 フェーズ周期)。
- 低電力モードのフルサポート: LCD コントローラは、SLEEP、LOW-POWER RUN、LOW-POWER SLEEP、および STOP モードで表示でき、または完全に無効にして消費電力を低減できます。
- 消費電力と EMI (電磁妨害) を削減するために、位相反転が組み込まれています。
- LCD データ RAM を更新するときには、ソフトウェアを同期するために、フレーム開始割り込み。
- 点滅機能:
 - 最大 1、2、3、4、8、またはすべてのピクセルを設定可能な周波数で点滅するようにプログラムできます。
 - ソフトウェアで点滅周波数を調整して、約 0.5 Hz、1 Hz、2 Hz、または 4 Hz を設定可能。
- 使用される LCD セグメントとコモンピンは、GPIO オルタネート機能として設定する必要があり、使用されないセグメントとコモンピンは汎用 I/O または別のペリフェラルオルタネート機能に使用できます。
- V_{LCD} レールデカップリング機能

注: LCD が内部ステップアップコンバータに依存しているときには、 V_{LCD} ピンをコンデンサ経由で V_{SS} に接続してください。平均値は 1 μF です (詳細については、製品データシートの C_{EXT} 値を参照してください)。

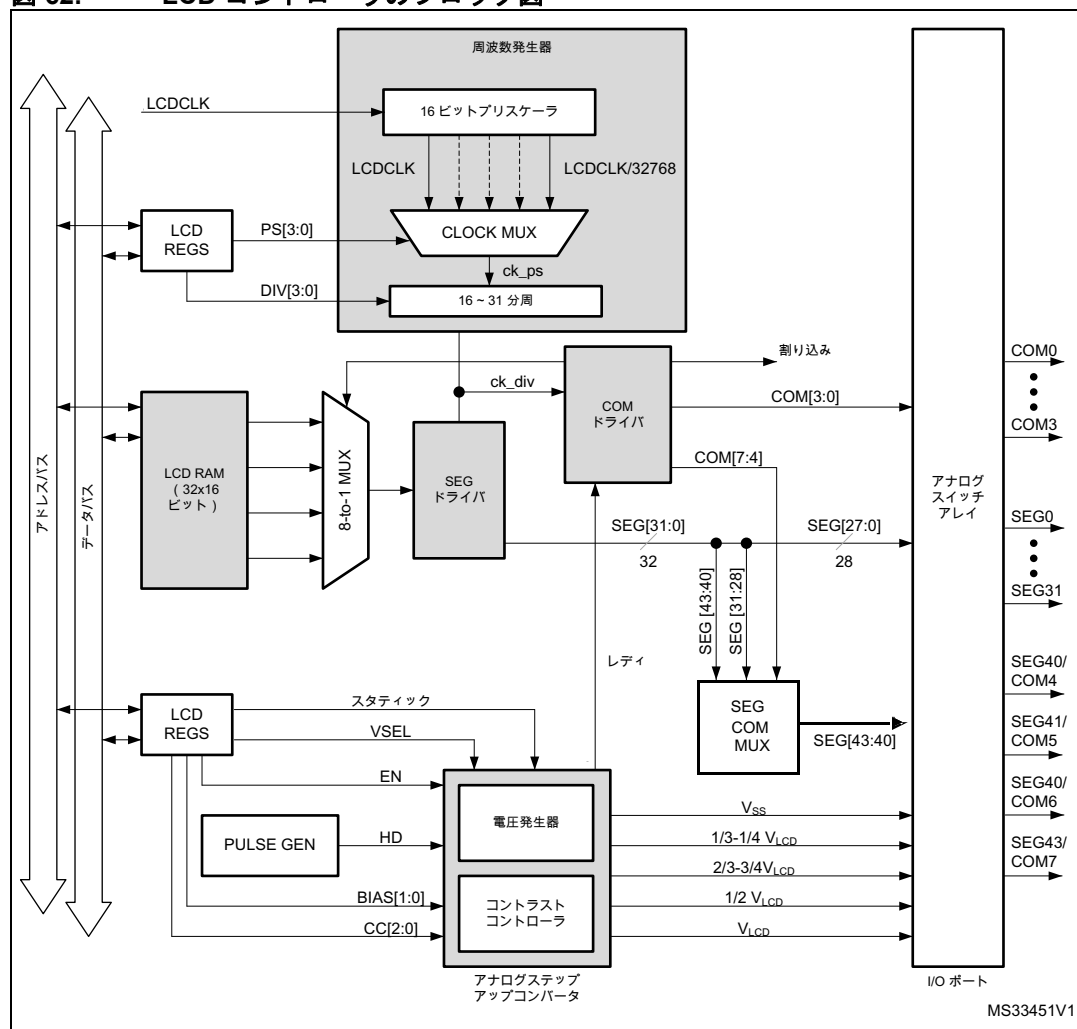
注: LCD ペリフェラルを使用しない場合は、 V_{LCD} ピンを V_{DDA} に接続してください。

17.3 LCD の機能詳細

17.3.1 概要

LCD コントローラには 5 つのメインブロックがあります (図 62 を参照)。

図 62. LCD コントローラのブロック図



注： LCDCLK は RTCCLK と同じです。このマニュアルの RCC セクションの RTC/LCD クロックの説明を参照してください。

周波数発生器により、32 kHz から 1 MHz までの LCD 入力クロック周波数 (LCDCLK) から開始されるさまざまな LCD フレームレートを設定できます。

次の 3 つのクロックソースを使用して、LCD クロック (LCDCLK/RTCCLK) を提供できます。

- 32 kHz 低速外部 RC (LSE)
- 37 kHz 低速内部 RC (LSI)
- 2、4、8、または 16 分周の高速外部 (HSE) による 1 MHz クロック

17.3.2 周波数発生器

このクロックソースが安定していなければ、正確な LCD タイミングを得られず、LCD セグメント間の DC 電圧オフセットを最小化できません。入力クロック LCDCLK は、1 から $2^{15} \times 31$ までの値で分周できます（[セクション 17.6.2 : LCD フレーム制御レジスタ \(LCD_FCR\) \(351 ページ\)](#) を参照）。周波数発生器は、プリスケアラ（16 ビットリップルカウンタ）と 16 ～ 31 のクロック分周器で構成されます。LCD_FCR レジスタの PS[3:0] ビットは、 $2^{PS[3:0]}$ で分周された LCDCLK を選択します。よりきめ細かな分解能が必要な場合は、LCD_FCR レジスタの DIV[3:0] ビットを使用して、クロックをさらに 16 ～ 31 分周できます。このように周波数を大まかに調整した後、クロックをカウンタで直線的に調整することによって微調整できます。周波数発生器ブロックの出力は f_{ck_div} であり、LCD コントローラ全体のタイムベースを構成します。ck_div 周波数は、フレーム周波数ではなく（スタティックデューティの場合のみ等しくなります）、LCD フェーズ周波数に等しくなります。フレーム周波数 (f_{frame}) は f_{ck_div} をアクティブなコモン端子の数で除算することによって（またはデューティの場合は乗算することによって）求められます。したがって、周波数発生器の入力クロック周波数 (f_{LCDCLK}) と出力クロック周波数 f_{ck_div} の関係は、次のとおりです。

$$f_{ckdiv} = \frac{f_{LCDCLK}}{2^{PS} \times (16 + DIV)}$$

$$f_{frame} = f_{ckdiv} \times duty$$

このため、周波数発生器は柔軟性が非常に高くなっています。フレームレートの計算例を [表 63](#) に示します。

表 63. フレームレートの計算例

| LCDCLK | PS[3:0] | DIV[3:0] | ? | ????? | f_{frame} |
|------------|---------|----------|------|--------|-------------|
| 32.768 kHz | 3 | 1 | 136 | 1/8 | 30.12 Hz |
| 32.768 kHz | 4 | 1 | 272 | 1/4 | 30.12 Hz |
| 32.768 kHz | 4 | 6 | 352 | 1/3 | 31.03 Hz |
| 32.768 kHz | 5 | 1 | 544 | 1/2 | 30.12 Hz |
| 32.768 kHz | 6 | 1 | 1088 | スタティック | 30.12 Hz |
| 32.768 kHz | 1 | 4 | 40 | 1/8 | 102.40 Hz |
| 32.768 kHz | 2 | 4 | 80 | 1/4 | 102.40 Hz |
| 32.768 kHz | 2 | 11 | 108 | 1/3 | 101.14 Hz |
| 32.768 kHz | 3 | 4 | 160 | 1/2 | 102.40 Hz |
| 32.768 kHz | 4 | 4 | 320 | スタティック | 102.40 Hz |
| 1.00 MHz | 6 | 3 | 1216 | 1/8 | 102.80 Hz |
| 1.00 MHz | 7 | 3 | 2432 | 1/4 | 102.80 Hz |
| 1.00 MHz | 7 | 10 | 3328 | 1/3 | 100.16 Hz |
| 1.00 MHz | 8 | 3 | 4864 | 1/2 | 102.80 Hz |
| 1.00 MHz | 9 | 3 | 9728 | スタティック | 102.80 Hz |

フレーム周波数は、約30 Hz から約100 Hz の範囲内で選択する必要があります。消費電力と許容可能なリフレッシュレートを考慮して選択する必要があります。さらに、専用の点滅プリスケアラは点滅周波数を選択します。この周波数は、次のように定義されます。

$$f_{\text{BLINK}} = f_{\text{ck_div}} / 2^{(\text{BLINKF} + 3)},$$

$$\text{BLINKF}[2:0] = 0, 1, 2, \dots, 7$$

設定できる点滅周波数は、0.5 Hz、1 Hz、2 Hz、または 4 Hz の範囲内です。

17.3.3 コモンドライバ

コモン信号は、コモンドライバブロックによって生成されます (図 62 を参照)。

COM 信号バイアス

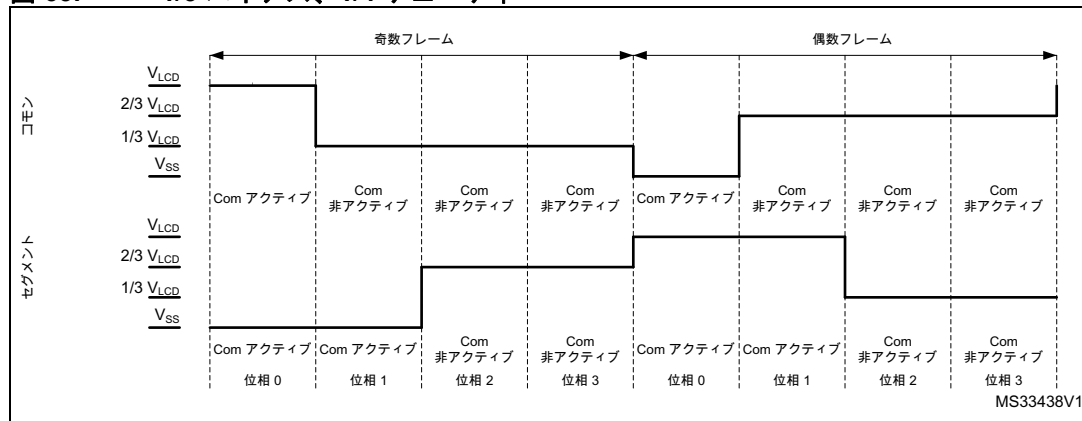
各 COM 信号の波形は同じですが、位相は異なります。対応するフレームサイクルの位相でのみ最大信振幅 V_{LCD} または V_{SS} になりますが、他の位相での信号振幅は次のとおりです。

- 1/4 バイアスの場合、 $1/4 V_{\text{LCD}}$ または $3/4 V_{\text{LCD}}$
- 1/3 バイアスの場合、 $1/3 V_{\text{LCD}}$ または $2/3 V_{\text{LCD}}$
- 1/2 バイアスの場合、 $1/2 V_{\text{LCD}}$

1/2、1/3、および 1/4 バイアスモードの選択は、LCD_CR レジスタの BIAS ビットで行うことができます。

同じ位相中に対応するコモンラインとセグメントラインの両方がアクティブのときに、すなわち、コモンとセグメントの電圧差がこの位相で最大になったときに、ピクセルはアクティブになります。EMI を削減するために、コモン信号は位相反転されます。図 63 に示されているように、位相反転の場合、各奇数サイクルの終わりに $1/2 V_{\text{LCD}}$ の平均電圧になります。

図 63. 1/3 バイアス、1/4 デューティ



1/2 バイアス (BIAS = 01) の場合、 V_{LCD} ピンは、偶数および奇数フレームについて、VLCDrail2 上で $1/2 V_{\text{LCD}}$ に等しい中間電圧を生成します (図 66 を参照)。

COM 信号デューティ

LCD_CR レジスタの DUTY[2:0] ビットに応じて、COM 信号は、スタティックデューティ (図 65 を参照)、1/2 デューティ (図 66 を参照)、1/3 デューティ (図 67 を参照)、1/4 デューティ (図 68)、または 1/8 デューティ (図 69 を参照) で生成されます。

COM[n] $n[0 \sim 7]$ は、奇数フレームの位相 n でアクティブなので、COM ピンは V_{LCD} に駆動されます。

偶数フレームの位相 n では、COM ピンは V_{SS} に駆動されます。

1/3 または 1/4 バイアスの場合、次のようになります。

- COM[n] は n 以外の位相では非アクティブなので、COM ピンは、奇数フレームでは 1/3 (1/4) V_{LCD} に駆動され、偶数フレームでは 2/3 (3/4) V_{LCD} に駆動されます。

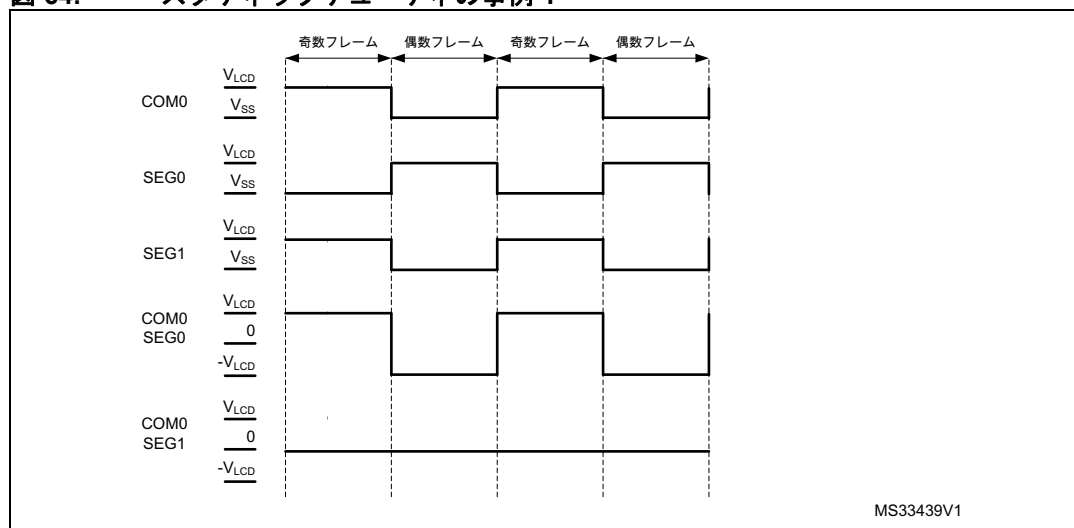
1/2 バイアスの場合、次のようになります。

- COM[n] が n 以外の位相で非アクティブな場合、COM ピンは常に (奇数フレームと偶数フレーム)、1/2 V_{LCD} に駆動されます。

スタティックデューティが選択されると、セグメントラインは多重化されず、各セグメント出力が 1 ピクセルに対応します。このように、最大 32 ピクセルだけが駆動されます。COM[0] は、COM[7:1] が使用されていないときには常にアクティブであり、 V_{SS} に駆動されます。

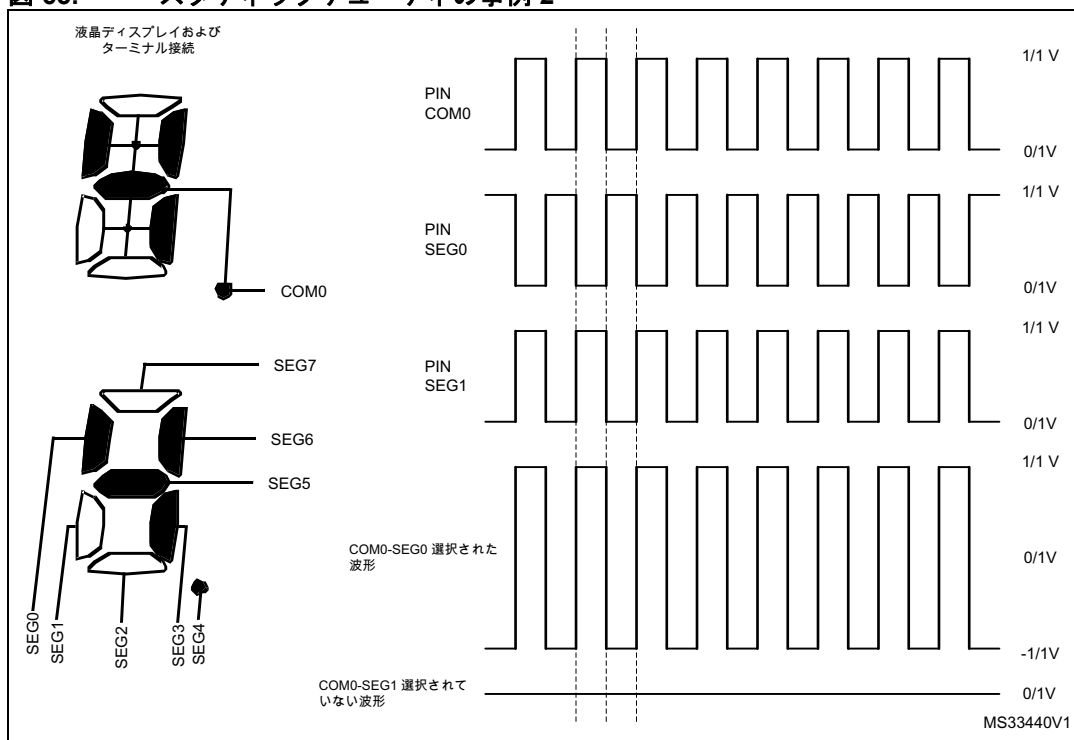
LCD_CR レジスタの LCDEN ビットがリセットされると、すべてのコモンラインが V_{SS} にプルダウンされ、LCD_SR レジスタの ENS フラグは 0 になります。スタティックデューティは、COM[0] が常にアクティブであり、セグメントラインとコモンラインについて V_{LCD} と V_{SS} の 2 つの電圧レベルだけが使用されることを意味します。ピクセルは、対応する SEG ラインが COM と反対の電圧の場合にアクティブであり、電圧が等しいときに非アクティブです。このようにして、LCD のコントラストは最大になります (図 64、図 65 を参照)。図 64 では、ピクセル 0 はアクティブであり、ピクセル 1 は非アクティブです。

図 64. スタティックデューティの事例 1



各フレームの位相は 1 つだけなので、 f_{frame} は f_{LCD} に等しくなります。1/4 デューティが選択された場合、フレームには 4 つの位相があり、COM[0] は位相 0 でアクティブであり、COM[1] は位相 1 でアクティブであり、COM[2] は位相 2 でアクティブであり、COM[3] は位相 3 でアクティブです。

図 65. スタティックデューティの事例 2

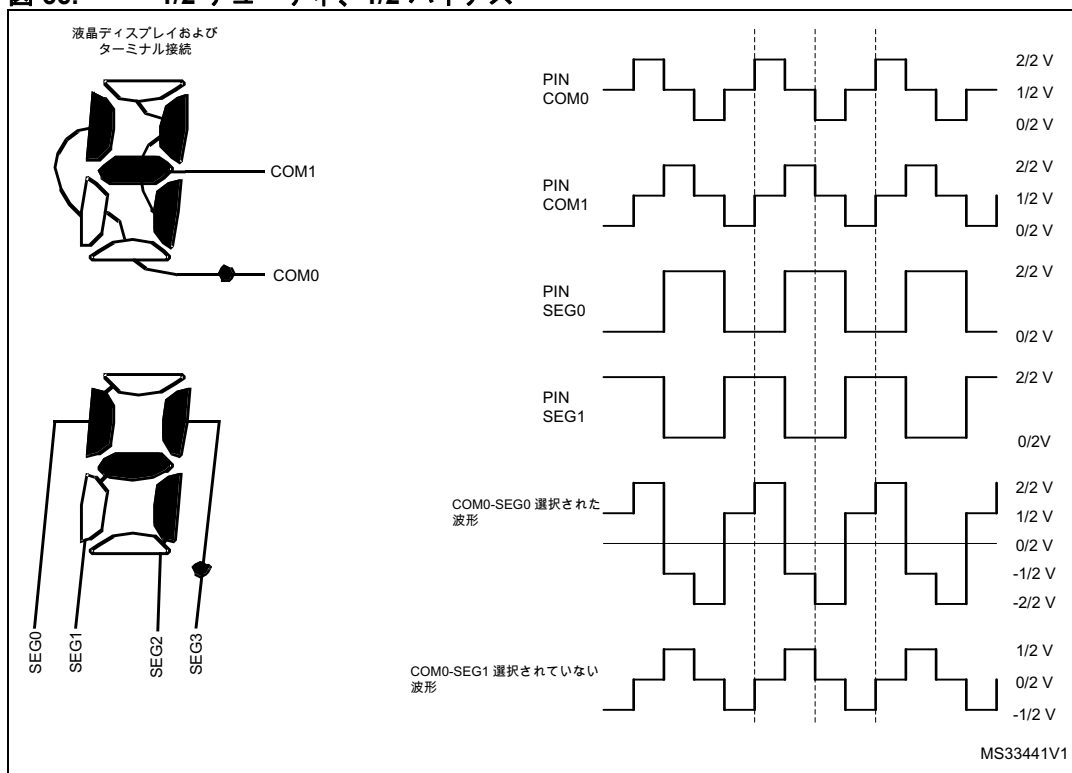


このモードでは、セグメント端子が多重化され、それぞれが 4 ピクセルを制御します。ピクセルは、対応する SEG ラインと COM ラインの両方が同じ位相でアクティブなときだけアクティブになります。1/4 デューティの場合、COM[0] に接続されているピクセル 0 を非アクティブにするには、COM[0] がアクティブなときに、位相 0 中に SEG[0] を非アクティブにする必要があります。COM[1] に接続されているピクセル 32 をアクティブにするには、COM[1] がアクティブなときに、位相 1 中に SEG[0] をアクティブにする必要があります（図 68 を参照）。COM[0] に接続されているピクセル 0 から 31 までをアクティブにするには、COM[0] がアクティブなときに、位相 0 中に SEG[0:31] をアクティブにする必要があります。これらの考え方は、他のピクセルにも適用できます。

8 ~ 1 Mux

COM[0] がアクティブなとき、コモンドライバブロックは、最初の 2 つの RAM レジスタ位置の内容を選択するために、図 62 に示されている 8 ~ 1 mux も駆動します。COM[7] がアクティブなとき、8 ~ 1 mux の出力は、最後の 2 つの RAM 位置の内容です。

図 66. 1/2 デューティ、1/2 バイアス



17.3.4 セグメントドライバ

セグメントドライバブロックは、コモンドライバブロックによって各位相で駆動される 8 ~ 1 mux からのピクセルデータに従って、SEG ラインを制御します。

1/4 または 1/8 デューティの場合

COM[0] がアクティブなとき、COM[0] に接続されているピクセルに関するピクセル情報（アクティブ／非アクティブ）（最初の 2 つの LCD_RAM 位置の内容）は、8 ~ 1 mux に送られます。

SEG[n] ピン n [0 ~ 31] は、奇数フレームの位相 0 で V_{SS} に駆動されます（COM[0] がアクティブのときにピクセル n がアクティブであることを示します）。

SEG[n] ピンは、偶数フレームの位相 0 で V_{LCD} に駆動されます。ピクセル n が非アクティブの場合、SEG[n] ピンは、奇数フレームでは $2/3$ ($2/4$) V_{LCD} に、または偶数フレームでは $1/3$ ($2/4$) V_{LCD} に駆動されます（ V_{LCD} パッドで電流反転）（図 63 を参照）。

1/2 バイアスでは、ピクセルが非アクティブの場合、SEG[n] ピンは、奇数フレームでは V_{LCD} に、偶数フレームでは V_{SS} に駆動されます。

LCD コントローラが無効なとき（LCD_CR レジスタの LCDEN ビットがクリア）、SEG ラインは V_{SS} にプルダウンされます。

図 67. 1/3 デューティ、1/3 バイアス

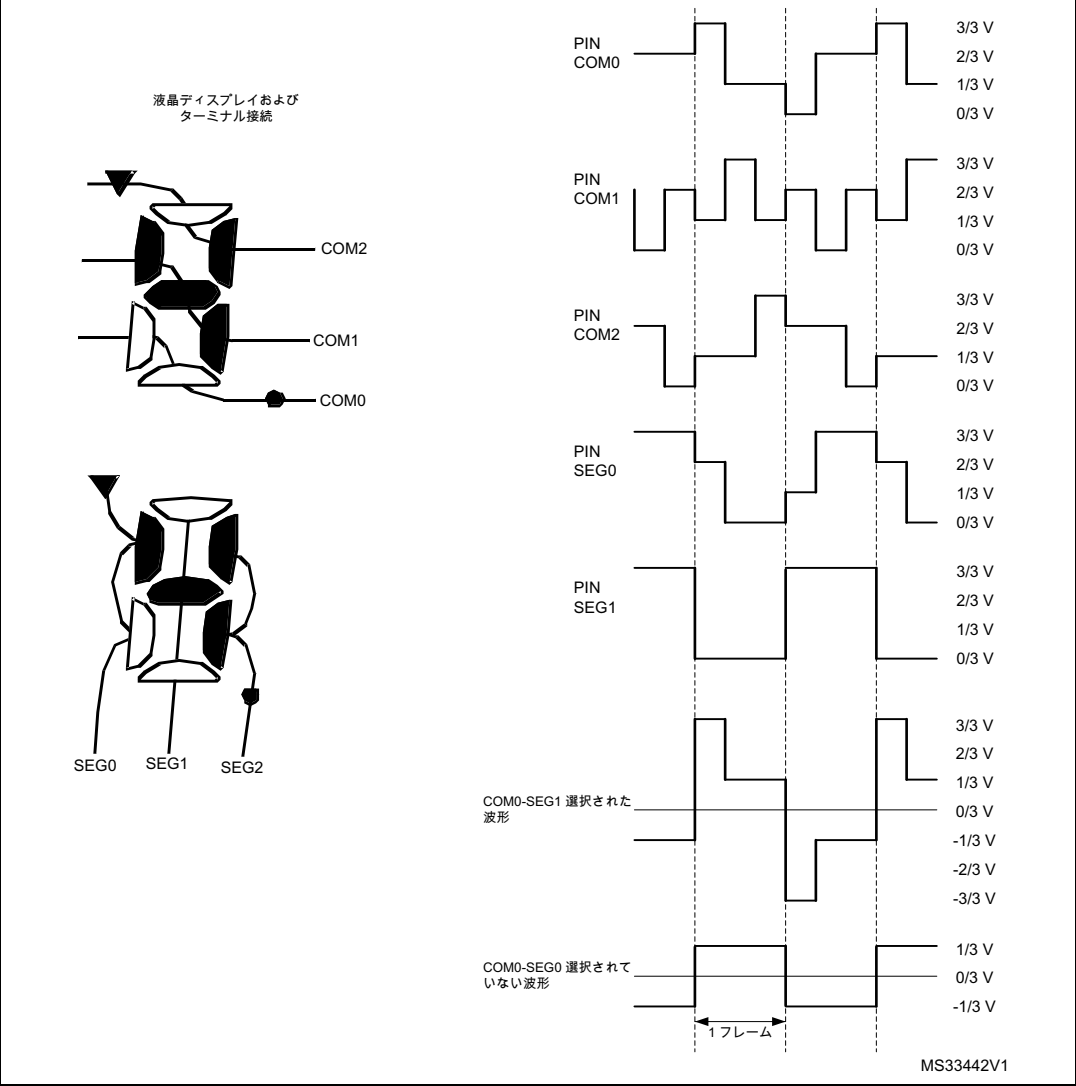


図 68. 1/4 デューティ、1/3 バイアス

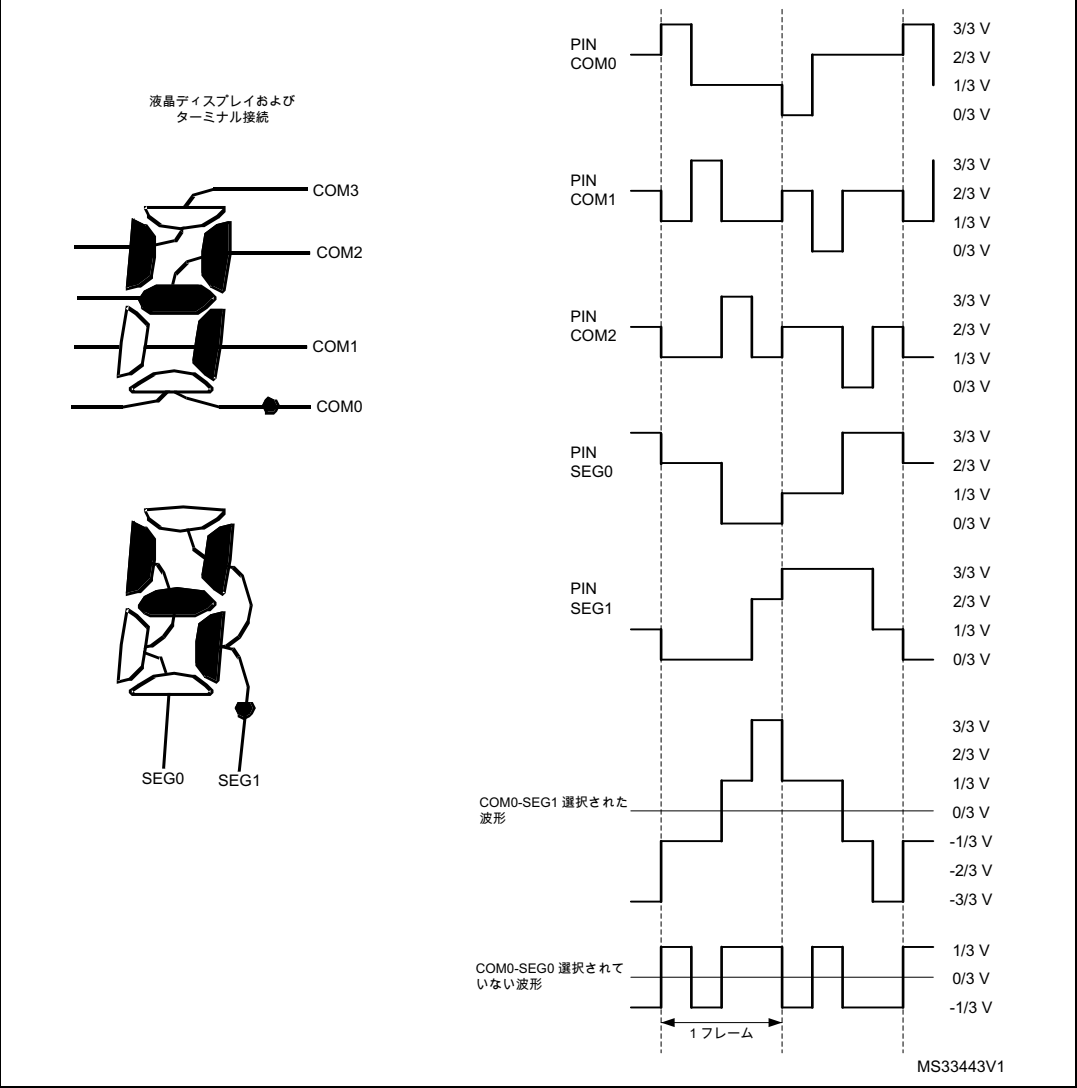
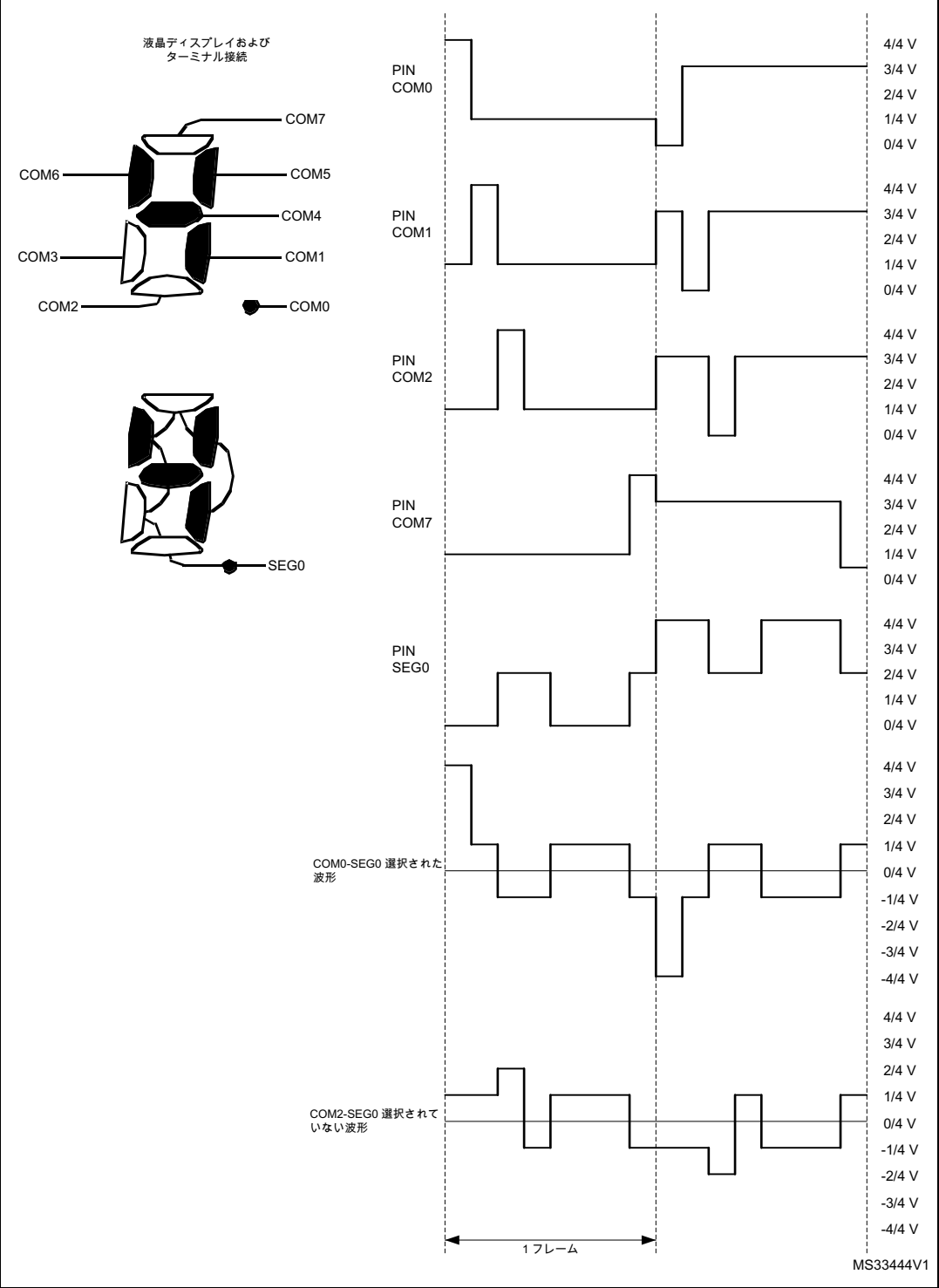


図 69. 1/8 デューティ、1/4 バイアス



点滅

セグメントドライバはプログラム可能な点滅機能も実装して、一部のピクセルを特定の周波数で連続的に点灯することができます。点滅モードは、LCD_FCR レジスタの BLINK[1:0] ビットで設定でき、最大 1、2、4、8、またはすべてのピクセルを点滅することができます（[セクション 17.6.2 : LCD フレーム制御レジスタ \(LCD_FCR\)](#) を参照）。点滅周波数は、LCD_FCR レジスタの BLINKF[2:0] ビットを使用して、8 つの値から選択できます。

[表 64](#) に、さまざまな点滅周波数の例を示します（ck_div 周波数の係数として）。

表 64. 点滅周波数

| BLINKF[2:0] ビット | | | ck_div 周波数 (LCDCLK 周波数 32.768 kHz) | | | |
|--------------------|---|---|------------------------------------|---------|---------|---------|
| | | | 32 Hz | 64 Hz | 128 Hz | 256 Hz |
| 0 | 0 | 0 | 4.0 Hz | N/A | N/A | N/A |
| 0 | 0 | 1 | 2.0 Hz | 4.0 Hz | N/A | N/A |
| 0 | 1 | 0 | 1.0 Hz | 2.0 Hz | 4.0 Hz | N/A |
| 0 | 1 | 1 | 0.5 Hz | 1.0 Hz | 2.0 Hz | 4.0 Hz |
| 1 | 0 | 0 | 0.25 Hz | 0.5 Hz | 1.0 Hz | 2.0 Hz |
| 1 | 0 | 1 | N/A | 0.25 Hz | 0.5 Hz | 1.0 Hz |
| 1 | 1 | 0 | N/A | N/A | 0.25 Hz | 0.5 Hz |
| 1 | 1 | 1 | N/A | N/A | N/A | 0.25 Hz |

17.3.5 電圧発生器とコントラスト制御

LCD 電源

LCD 電源は、内部ステップアップコンバータから、または V_{LCD} ピンに印加される外部電圧から取ることができます。内部または外部電圧ソースは、LCD_CR レジスタの VSEL ビットを使用して選択できます。外部ソースを選択した場合、内部ブースト回路（ステップアップコンバータ）は無効になり、消費電力を削減します。

ステップアップコンバータを V_{LCD} ソースとして選択したときには、LCD_FCR（[セクション 17.6.2](#) を参照）レジスタ内の CC[2:0] (Contrast Control) ビットによって、 V_{LCD} 値を V_{LCDmin} から V_{LCDmax} までの幅広い値のセットから選択できます。新しい V_{LCD} 値は、新しいフレームの開始ごとに有効になります。

外部電源が V_{LCD} ソースとして選択されたときには、 V_{LCD} 電圧を V_{LCDmin} から V_{LCDmax} の範囲から選ぶ必要があります（データシートを参照）。その場合、フレーム間のデッドタイムをプログラムすることによってコントラストを制御できます（[デッドタイム \(345 ページ\)](#) を参照）。

LCD 中間電圧

LCD 電圧発生器は、1/3 (1/4) バイアスの場合、 V_{SS} と V_{LCD} の間で最大 3 つの中間電圧レベル（1/3 V_{LCD} 、2/3 V_{LCD} 、または 1/4 V_{LCD} 、2/4 V_{LCD} 、3/4 V_{LCD} ）を生成し、1/2 バイアスの場合は V_{SS} と V_{LCD} の間で 1 つの電圧レベル（1/2 V_{LCD} ）を生成します。

1/2 バイアスの場合、1 つの電圧レベル（1/2 V_{LCD} ）が生成され、ノード b の電圧は 1/2 V_{LCD} です。

1/3 バイアスの場合、2 つの中間電圧レベル（1/3 V_{LCD} 、2/3 V_{LCD} ）が生成されます。

- ノード a は 1/3 V_{LCD}

- ノード b は $2/3 V_{LCD}$
- 1/4 バイアスの場合、3 つの中間電圧レベル ($1/4 V_{LCD}$ 、 $1/2 V_{LCD}$ 、および $3/4 V_{LCD}$) が生成されます。
- ノード a は $1/4 V_{LCD}$
- ノード b は $1/2 V_{LCD}$
- ノード c は $3/4 V_{LCD}$

LCD ドライブ選択

1. 高駆動および低駆動抵抗ネットワーク

内部抵抗ネットワークは、すべての中間電圧を生成するために使用されます (図 70 を参照)。実際には、低値抵抗 (R_{LN}) の回路と高値抵抗 (R_{HN}) の回路のそれぞれが、遷移時の電流を増加するためと、スタティック状態の消費電力を削減するために使用されます。

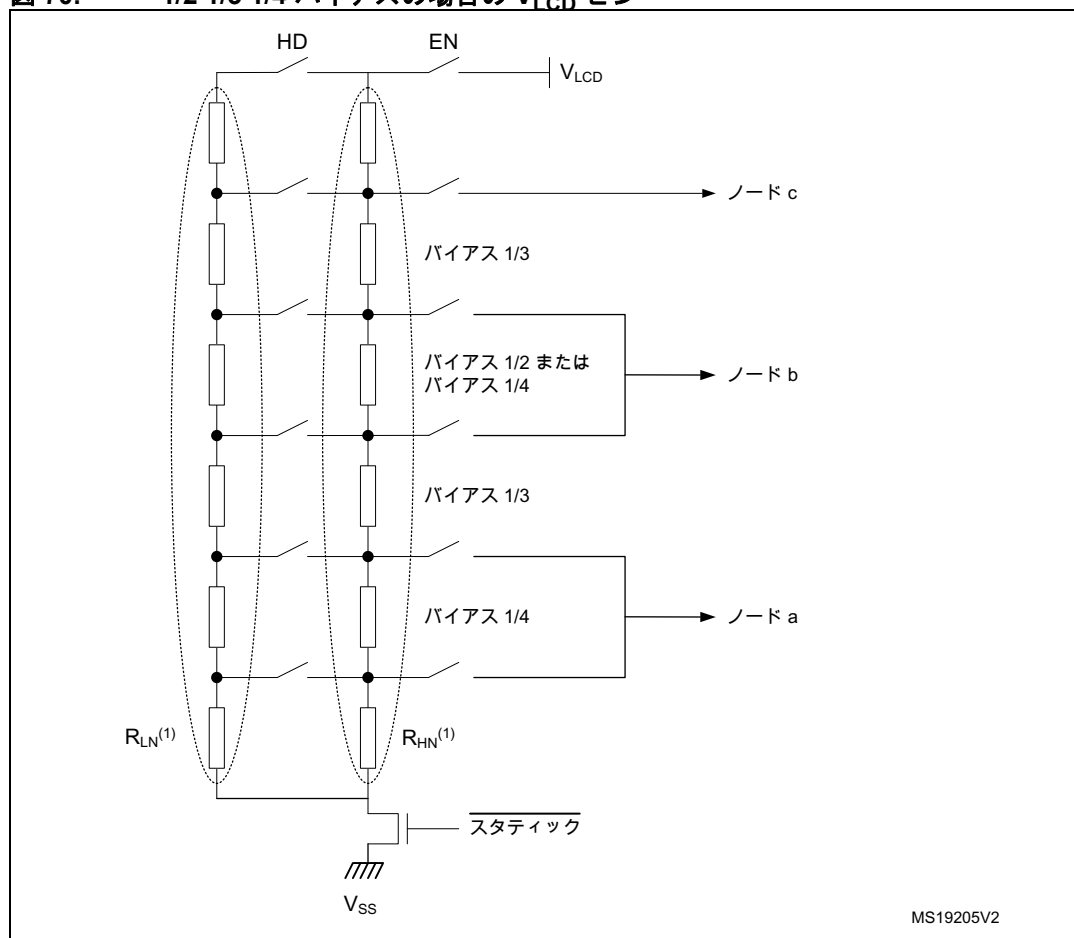
EN スイッチは、次のルールに従います。

LCD_CR レジスタの LCDEN ビットがセットされた場合、EN スイッチは ON します。

LCD_CR レジスタの LCDEN ビットをクリアすると、フレーム全体が奇数プラス偶数とみなされる 0 以外の中間電圧レベルを避けるために、EN スイッチは偶数フレームの終わりで開きます。

LCD_FCR レジスタの PON[2:0] (Pulse ON 期間) ビットは、コモンおよびセグメントラインのレベルが変化したときに HD (高駆動) スイッチを通じて R_{LN} が有効 (図 62 を参照) になる時間を設定します。駆動時間が短いと消費電力を削減できますが、内部抵抗が高いディスプレイは、十分なコントラストを達成するために長い駆動時間を必要とすることがあります。

図 70. 1/2 1/3 1/4 バイアスの場合の V_{LCD} ピン



1. R_{LN} : 低値抵抗ネットワーク。 R_{HN} : 高値抵抗ネットワーク。

LCD_FCR 設定レジスタの HD ビットを使用して、 R_{LN} 分周器を常にオンにしておくことができます (セクション 17.6.2 を参照)。

HD スイッチは次のルールに従います。

- LCD_FCR レジスタの HD ビットと PON[2:0] ビットがリセットされた場合、HD スイッチは OFF します。
- LCD_FCR レジスタの HD ビットがリセットされ、LCD_FCR の PON[2:0] ビットが 00 以外の場合、HD スイッチは PON[2:0] ビットで定義されたパルス数だけ閉じます。
- LCD_FCR レジスタの HD ビットが 1 の場合、HD スイッチは常に ON しています。

LCDEN ビットが有効になった後、LCD_SR レジスタの RDY ビットがセットされて、電圧レベルが安定していることと、LCD コントローラが動作を開始できることを示します。

外部デカップリング

V_{LCD} レールデカップリング機能を持つデバイス (デバイスのデータシートを参照) は、内部 VLCD レール (LCD_VLCD1、LCD_VLCD2、LCD_VLCD3) をオプションのコンデンサに接続できます。これにより、セグメントおよびコモンラインの信号波形が向上し、期待値に近い VSEG-VCOM rms 値が得られます。LCD_FCR レジスタで、より低い PON[2:0] 値を選択できるため、この機能は消費電力の点で特に有用です。

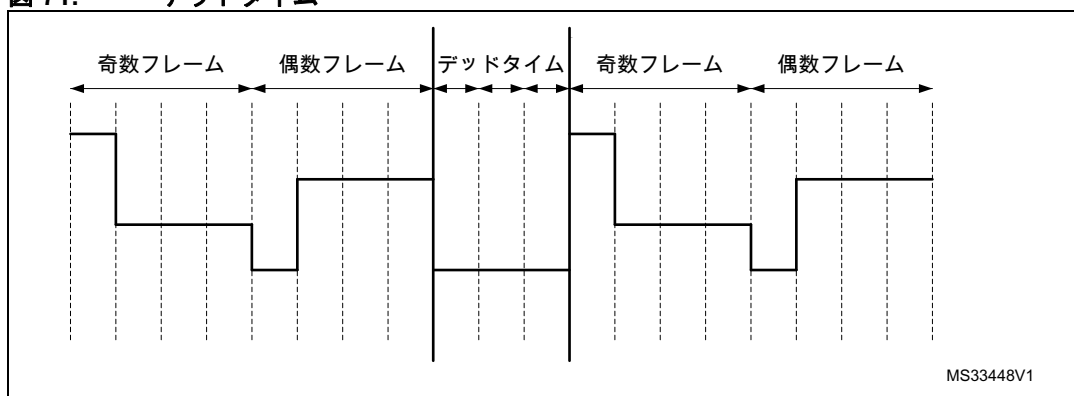
V_{LCD} レールを [表 39](#) に示されているように専用の GPIO に接続するには、SYSCFG_CFGR2 レジスタの CAPA[2:0] ビットを設定します ([セクション 10.2.2 : SYSCFG ペリフェラルモード設定レジスタ \(SYSCFG_CFGR2\)](#) を参照)。

効果的であるためには、これらのデカップリングコンデンサの値を LCD ガラスおよび PCB 容量に従って調整する必要があります。ガイドラインとして、ユーザはデカップリングコンデンサの値を LCD 容量の約 10 倍に設定できます。

デッドタイム

CC[2:0] ビットの使用に加えて、各フレーム間のデッドタイムをプログラムすることによって、コントラストを制御できます。デッドタイム中、COM および SEG の値は V_{SS} になります。LCD_FCR レジスタの DEAD[2:0] ビットを使用して、最大 8 フェーズ期間の時間をプログラムできます。このデッドタイムは、フレームレートを変更せずにコントラストを下げます。

図 71. デッドタイム



17.3.6 ダブルバッファメモリ

ダブルバッファメモリを使用することにより、LCD コントローラは表示される情報のコヒーレンスを確保でき、割り込みを使用して LCD_RAM の変更を制御する必要がありません。

アプリケーションソフトウェアは APB インタフェースを通じて最初のバッファレベル (LCD_RAM) にアクセスできます。LCD_RAM を変更すると、LCD_SR レジスタの UDR フラグをセットします。この UDR フラグ (Update Display Request) は、更新された情報を 2 番目のバッファレベル (LCD_DISPLAY) に移動するように要求します。

この動作はフレームと同期して (次のフレームの開始時に) 行われ、更新が完了するまで、LCD_RAM は書き込み保護され、UDR フラグはハイのままです。更新が完了すると、別のフラグ (UDD - Update Display Done) がセットされ、LCD_FCR レジスタの UDDIE ビットがセットされている場合は割り込みを生成します。

LCD_DISPLAY を更新するための所要時間は、最悪の場合、1 奇数フレームと 1 偶数フレームです。

ディスプレイが有効 (LCDEN = 1) になるまで、更新は発生しません (UDR = 1 かつ UDD = 0)。

17.3.7 COM および SEG の多重化

出力ピン対デューティモード

出力ピンは、次のもので構成されます。

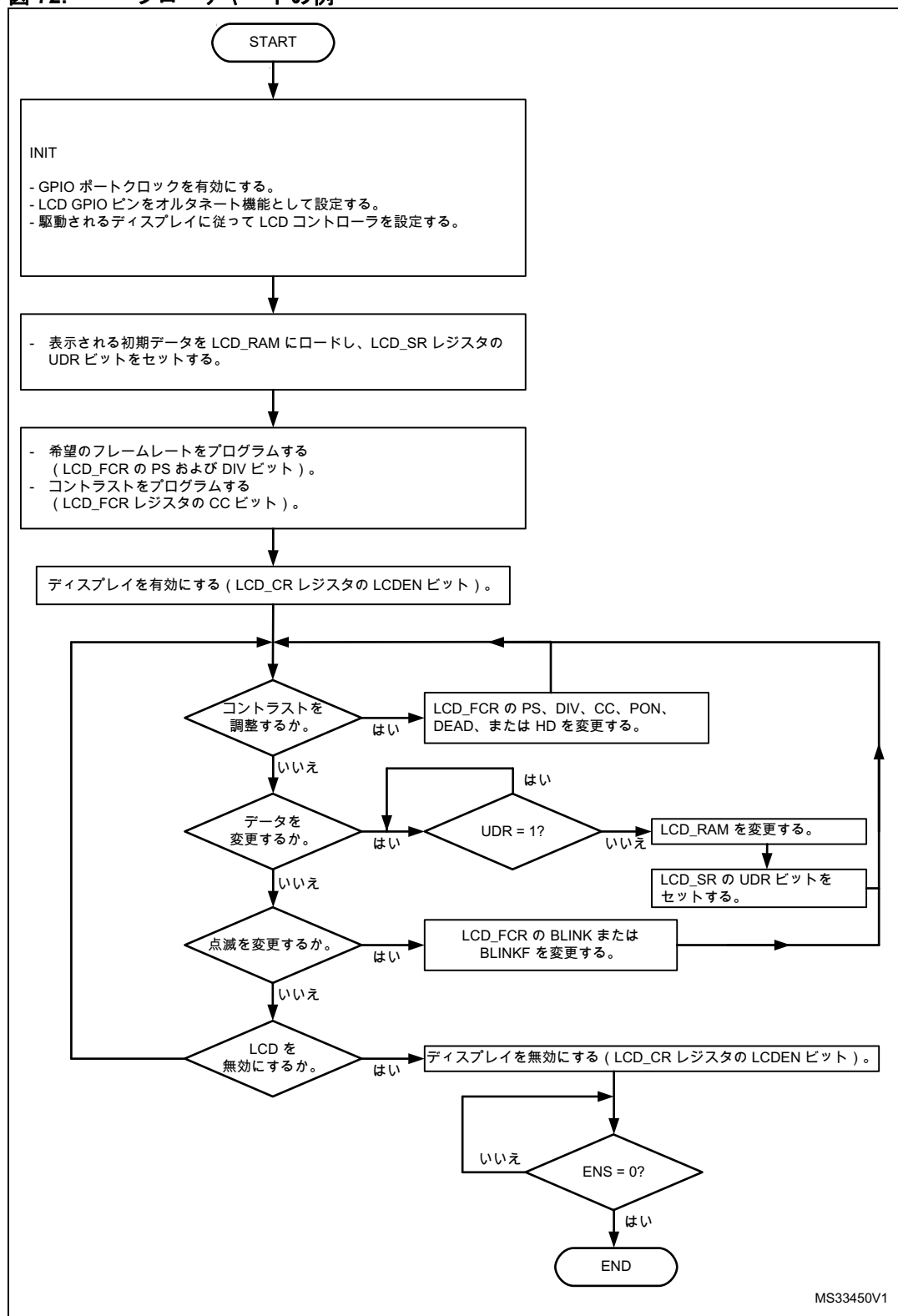
- SEG[31:0]
- COM[3:0]

デューティ設定に応じて、COM および SEG 出力ピンは異なる機能を持ちます。

- スタティック、1/2、1/3、および 1/4 デューティモードでは、最大 32 本の SEG ピンと、それぞれ 1、2、3、および 4 本の COM ピンがあります。
- 1/8 デューティモード (DUTY[2:0] = 100) では、COM[7:4] 出力は SEG[31:28] ピンで使用でき、使用可能なセグメント数が 28 に減ります。

17.3.8 フローチャート

図 72. フローチャートの例



17.4 LCD 低電力モード

LCD コントローラは、STOP モードで表示するか、完全に無効にして、消費電力を削減できます。

表 65. 低電力モードでの LCD の動作

| モード | 説明 |
|---------|---------------------|
| STOP | LCD は、アクティブです。 |
| STANDBY | LCD は、アクティブではありません。 |

17.5 LCD 割り込み

次の表に、LCD 割り込みリクエストの一覧を示します。

表 66. LCD 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イベントフラグ/ 割り込みのクリア方法 | 割り込み有効制御ビット |
|------------------|---------|------------------------|-------------|
| フレーム開始 (SOF) | SOF | SOFC = 1 を書き込む | SOFIE |
| ディスプレイ更新終了 (UDD) | UDD | UDDC = 1 を書き込む | UDDIE |

フレーム開始 (SOF)

SOFIE (Start Of Frame Interrupt Enable) ビットがセットされている場合、LCD フレーム開始割り込みが実行されます ([セクション 17.6.2 : LCD フレーム制御レジスタ \(LCD_FCR\)](#) を参照)。SOF は、対応する割り込み処理ベクトルを実行するときに LCD_CLR レジスタの SOFC ビットに 1 を書き込むことによってクリアされます。

ディスプレイ更新終了 (UDD)

UDDIE (Update Display Done Interrupt Enable) ビットがセットされている場合、LCD ディスプレイ更新割り込みが実行されます ([セクション 17.6.2 : LCD フレーム制御レジスタ \(LCD_FCR\)](#) を参照)。UDD は、対応する割り込み処理ベクトルを実行するときに LCD_CLR レジスタの UDDC ビットに 1 を書き込むことによってクリアされます。

製品実装に応じて、これらの割り込みイベントのすべてで同じ割り込みベクトルを共有するか (LCD グローバル割り込み)、2 つの割り込みベクトル (LCD SOF 割り込みと LCD UDD 割り込み) にグループ化することができます。詳細については、 [表 45 : ベクタテーブル](#) を参照してください。

LCD 割り込みを有効にするには、次のシーケンスが必要です。

1. NVIC で LCD IRQ チャンネルを設定し、有効にします。
2. 割り込みを生成するように LCD を設定します。

17.6 LCD レジスタ

ペリフェラルレジスタには、ワード（32 ビット）単位でアクセスする必要があります。

17.6.1 LCD 制御レジスタ（LCD_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

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

- ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 7 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 6:5 **BIAS[1:0]**：バイアスセレクタ
- これらのビットによって、使用されるバイアスが決まります。11 は禁止されています。
- 00：バイアス 1/4
 - 01：バイアス 1/2
 - 10：バイアス 1/3
 - 11：予約済み
- ビット 4:2 **DUTY[2:0]**：デューティ選択
- これらのビットによって、デューティサイクルが決まります。値 101、110、および 111 は禁止されています。
- 000：スタティックデューティ
 - 001：1/2 デューティ
 - 010：1/3 デューティ
 - 011：1/4 デューティ
 - 100：1/8 デューティ
 - 101：予約済み
 - 110：予約済み
 - 111：予約済み
- ビット 1 **VSEL**：電圧ソース選択
- VSEL ビットによって、LCD の電圧ソースが決まります。
- 0：内部ソース（電圧ステップアップコンバータ）
 - 1：外部ソース（V_{LCD} ピン）
- ビット 0 **LCDEN**：LCD コントローラ有効
- このビットは、LCD コントローラ／ドライバを有効にするために、ソフトウェアによってセットされます。次のフレームの開始時に LCD をオフにするためにソフトウェアによってクリアされます。LCD が無効になると、すべての COM および SEG ピンが V_{SS} に駆動されます。
- 0：LCD コントローラ無効
 - 1：LCD コントローラ有効

注： **VSEL、BIAS、および DUTY ビットは、LCD が有効なときには（LCD_SR の ENS ビットが 1）、書き込み保護されます。**

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

17.6.2 LCD フレーム制御レジスタ (LCD_FCR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|------|------|---------|------|------|-----------|-----|-----|----------|----------|-----|-------|------|------------|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | PS[3:0] | | | | DIV[3:0] | | | | BLINK[1:0] | |
| | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BLINKF[2:0] | | | CC[2:0] | | | DEAD[2:0] | | | PON[2:0] | | | UDDIE | Res. | SOFIE | HD |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | r/w | r/w |

ビット 31:26 予約済みであり、リセット値のままにしておかなければなりません。

ビット 25:22 **PS[3:0]** : PS 16 ビットプリスケアラ

これらのビットは、PS 16 ビットプリスケアラの分周比を定義するために、ソフトウェアによって書き込まれます。

ck_ps = LCDCLK/(2)。を参照 [セクション 17.3.2](#)。

0000 : ck_ps = LCDCLK

0001 : ck_ps = LCDCLK/2

0002 : ck_ps = LCDCLK/4

...

1111 : ck_ps = LCDCLK/32768

ビット 21:18 **DIV[3:0]** : DIV クロック分周器

これらのビットは、DIV 分周器の分周比を定義するために、ソフトウェアによって書き込まれます。を参照 [セクション 17.3.2](#)。

0000 : ck_div = ck_ps/16

0001 : ck_div = ck_ps/17

0002 : ck_div = ck_ps/18

...

1111 : ck_div = ck_ps/31

ビット 17:16 **BLINK[1:0]** : 点滅モード選択

00 : 点滅は無効

01 : SEG[0]、COM[0] で点滅有効 (1 ピクセル)

10 : SEG[0]、すべての COM で点滅有効 (プログラムされたデューティに応じて最大 8 ピクセル)

11 : すべての SEG とすべての COM で点滅有効 (すべてのピクセル)

ビット 15:13 **BLINKF[2:0]** : 点滅周波数選択

000 : f_{LCD}/8

001 : f_{LCD}/16

010 : f_{LCD}/32

011 : f_{LCD}/64

100 : f_{LCD}/128

101 : f_{LCD}/256

110 : f_{LCD}/512

111 : f_{LCD}/1024

ビット 12:10 CC[2:0] : コントラスト制御

これらのビットは、 V_{LCD} 最大電圧の 1 つを指定します (V_{DD} とは無関係に)。2.60 V から 3.51V までの範囲です。

000 : V_{LCD0}
001 : V_{LCD1}
010 : V_{LCD2}
011 : V_{LCD3}
100 : V_{LCD4}
101 : V_{LCD5}
110 : V_{LCD6}
111 : V_{LCD7}

V_{LCDx} 値については、製品データシートを参照してください。

ビット 9:7 DEAD[2:0] : デッドタイム期間

これらのビットは、フレーム間のデッドタイムの長さを設定するために、ソフトウェアによって書き込まれます。フレームレートを変更せずにコントラストを下げるために、デッドタイム中、COM および SEG 電圧レベルは 0 V に保持されます。

000 : デッドタイムなし
001 : 1 位相周期のデッドタイム
010 : 2 位相周期のデッドタイム
.....
111 : 7 位相周期のデッドタイム

ビット 6:4 PON[2:0] : パルス ON 期間

これらのビットは、パルス期間を ck_ps パルス数で定義するために、ソフトウェアによって書き込まれます。パルスが短いと消費電力を削減できますが、内部抵抗が高いディスプレイは、十分なコントラストを達成するために長いパルスを必要とすることがあります。

パルスはプリスケールされた LCD クロック周期の 1/2 より長くはならないことに注意してください。

000 : 0
001 : 1/ ck_ps
010 : 2/ ck_ps
011 : 3/ ck_ps
100 : 4/ ck_ps
101 : 5/ ck_ps
110 : 6/ ck_ps
111 : 7/ ck_ps

LCDCLK = 32.768 kHz、PS=0x03 での PON 期間の例 :

000 : 0 μ s
001 : 244 μ s
010 : 488 μ s
011 : 782 μ s
100 : 976 μ s
101 : 1.22 ms
110 : 1.46 ms
111 : 1.71 ms

ビット 3 UDDIE : ディスプレイ更新終了割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : LCD ディスプレイ更新終了割り込み無効
1 : LCD ディスプレイ更新終了割り込み有効

ビット 2 予約済みであり、リセット値のままにしておかなければなりません。

ビット 1 **SOFIE** : フレーム開始割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : LCD フレーム開始割り込み無効

1 : LCD フレーム開始割り込み有効

ビット 0 **HD** : 高駆動有効

このビットは、低抵抗分圧器を有効にするために、ソフトウェアによって書き込まれます。高い内部抵抗を持つディスプレイは、十分なコントラストを達成するために、より長い駆動時間を必要とすることがあります。このようなとき、ある程度の消費電力の増加が許される場合には、このビットが便利です。

0 : 永続的な高駆動無効。

1 : 永続的な高駆動有効。HD=1 のときには、PON ビットを 001 にプログラムする必要があります。

注 : *このレジスタ内のデータはいつでも更新できますが、新しい値は次のフレームの開始時にのみ適用されます (デバイスの動作にすぐに影響する UDDIE、SOFIE を除きます)。*

CC[2:0] ビットの新しい値もすぐに適用されますが、デバイスに対する効果は、電圧発生器によって次のフレームの開始時に遅延されます。

このレジスタを読み出すと、現在のフレームの表示に使用された設定ではなく、レジスタに書き込まれた最後の値が得られます。

17.6.3 LCD ステータスレジスタ (LCD_SR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0020

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|-------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FCRSF | RDY | UDD | UDR | SOF | ENS |
| | | | | | | | | | | r | r | r | rs | r | r |

ビット 31:6 予約済みであり、リセット値のままにしておかなければなりません。

ビット 5 **FCRSF** : LCD フレーム制御レジスタ同期フラグ

このビットは、LCDCLK ドメインで LCD_FCR レジスタが更新されるたびに、ハードウェアによってセットされます。LCD_FCR レジスタに書き込んだときに、ハードウェアによってクリアされます。

0 : LCD フレーム制御レジスタは同期されていません。

1 : LCD フレーム制御レジスタは同期されました。

ビット 4 **RDY** : レディフラグ

このビットは、ハードウェアによってセット／クリアされます。ステップアップコンバータのステータスを示します。

0 : レディでない状態

1 : ステップアップコンバータは有効であり、正しい電圧を提供する準備ができています。

ビット 3 **UDD** : ディスプレイ更新終了

このビットは、ハードウェアによってセットされます。LCD_CLR レジスタの UDDC ビットに 1 を書き込むことによってクリアされます。ビットのセットは、クリアに優先します。

0 : イベントはありません。

1 : ディスプレイ更新リクエストが終了しました。LCD_FCR レジスタの UDDIE ビットがセットされた場合、UDD 割り込みが生成されます。

注 : デバイスが STOP モードの場合 (PCLK が提供されない)、UDDIE = 1 の場合でも UDD は割り込みを生成しません。

ディスプレイが有効でない場合、UDD 割り込みは発生しません。

ビット 2 **UDR** : ディスプレイ更新リクエスト

ソフトウェアが LCD_RAM を変更するたびに、更新されたデータを 2 番目のレベルバッファに転送するために、UDR ビットをセットする必要があります。UDR ビットは、更新の終了までセットされたままであり、この間、LCD_RAM は書き込み保護されます。

0 : 効果なし

1 : ディスプレイ更新リクエスト

注 : ディスプレイが無効なときには、すべての LCD_DISPLAY 位置について更新が行われます。ディスプレイが有効なときには、コモンがアクティブな位置についてのみ (DUTY に依存)、更新が行われます。たとえば、DUTY = 1/2 の場合、COM0 および COM1 の LCD_DISPLAY が更新されます。

注 : このビットに 0 を書き込むか、すでに 1 のときに 1 を書き込んでも、効果はありません。このビットは、ハードウェアによってのみクリアできます。LCDEN = 1 のときのみクリアできます。

ビット 1 SOF : フレーム開始フラグ

このビットは、ディスプレイデータが更新されると同時に、新しいフレームの開始時にハードウェアによってセットされます。LCD_CLR レジスタの SOFC ビットに 1 を書き込むことによってクリアされます。ビットのクリアは、セットに優先します。

0 : イベントはありません。

1 : フレーム開始イベントが発生しました。SOFIE ビットがセットされている場合、LCD フレーム開始割り込みが生成されます。

ENS : LCD 有効ステータス

このビットは、ハードウェアによってセット／クリアされます。LCD コントローラのステータスを示します。

0 : LCD コントローラ無効

1 : LCD コントローラ有効

注 : *ENS ビットは、LCD_CR の LCDEN ビットが 0 から 1 になると同時にセットされます。非アクティブ化のときには、LCD の実際のステータスを反映するので、最後に表示されたフレームの終了時に 0 になります。*

17.6.4 LCD クリアレジスタ (LCD_CLR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

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

ビット 31:2 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **UDDC** : ディスプレイ更新終了クリア

このビットは、LCD_SR レジスタの UDD フラグをクリアするために、ソフトウェアによって書き込まれます。

0 : 効果なし

1 : UDD フラグをクリアします。

ビット 2 予約済みであり、リセット値のままにしておかなければなりません。

ビット 1 **SOFC** : フレーム開始フラグクリア

このビットは、LCD_SR レジスタの SOF フラグをクリアするために、ソフトウェアによって書き込まれます。

0 : 効果なし

1 : SOF フラグをクリアします。

ビット 0 予約済みであり、リセット値のままにしておかなければなりません。

17.6.5 LCD ディスプレイメモリ (LCD_RAM)

アドレスオフセット : 0x14 から 0x50

リセット値 : 0x0000 0000

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

ビット 31:0 SEGMENT_DATA[31:0]

各ビットは、LCD ディスプレイの 1 ピクセルに対応します。

0 : ピクセルは非アクティブ

1 : ピクセルはアクティブ

17.6.6 LCD レジスタマップ

次の表に LCD レジスタの一覧を示します。

表 67. LCD レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-------------------|------|------|------|------|------|------|---------|------|------|----------|------|------|------------|------|-------------|------|-------------|------|---------------|------|--------------|------|-------|------|------|-----------|---------------|------|------|------|-------|------|
| 0x00 | LCD_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BIAS[1:0] | DUTY [2:0] | | VSEL | | LCDEN | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x04 | LCD_FCR | Res. | Res. | Res. | Res. | Res. | Res. | PS[3:0] | | | DIV[3:0] | | | BLINK[1:0] | | BLINKF[2:0] | | CC [2:0] | | DEAD [2:0] | | PON [2:0] | | UDDIE | | Res. | SOFIE | | HD | | | | |
| | リセット値 | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x08 | LCD_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FCRSF | RDY | UDD | UDR | SOF | ENS | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x0C | LCD_CLR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | 0 | | | |
| 0x14 | LCD_RAM (COM0) | Res. | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| 0x18 | | Res. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x1C | LCD_RAM (COM1) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| 0x20 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x24 | LCD_RAM (COM2) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| 0x28 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x2C | LCD_RAM (COM3) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| 0x30 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |

表 67. LCD レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 0x34 | LCD_RAM (COM4) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| 0x38 | LCD_RAM (COM4) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x3C | LCD_RAM (COM5) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| 0x40 | LCD_RAM (COM5) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x44 | LCD_RAM (COM6) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| 0x48 | LCD_RAM (COM6) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x4C | LCD_RAM (COM7) | S31 | S30 | S29 | S28 | S27 | S26 | S25 | S24 | S23 | S22 | S21 | S20 | S19 | S18 | S17 | S16 | S15 | S14 | S13 | S12 | S11 | S10 | S09 | S08 | S07 | S06 | S05 | S04 | S03 | S02 | S01 | S00 |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| 0x50 | LCD_RAM (COM7) | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレス表については、[セクション 2.2.2](#) を参照してください。

18 タッチセンシングコントローラ (TSC)

18.1 概要

タッチセンシングコントローラを使用することで、どんなアプリケーションにも簡単に静電容量検出機能を追加することができます。静電容量検出技術とは、誘電材料（ガラス、プラスチックなど）により直接触れることがないよう保護されている電極の近くに近づけた指を検出することができる技術です。指（または何らかの導電性物体）によって発生する静電容量の変化は、表面電荷移動取得原理に基づき、実証済みの方法で測定されます。

タッチセンシングコントローラは、STMTouch タッチ検出ファームウェアライブラリによってフルサポートされています。この自由に利用できるライブラリにより、タッチ検出機能がエンドアプリケーションに確実に実装されます。

18.2 TSC の主な機能

タッチセンシングコントローラは、以下のような主な機能を備えています。

- 実績のある安定した表面電荷移動取得原理を採用。
- 最大 24 本の静電容量検出チャンネルに対応。
- 最大 8 本の静電容量検出チャンネルの 同時取得により、非常に優れた応答時間を実現。
- スペクトル拡散機能により、ノイズの多い環境でのシステムの安定性が向上。
- 電荷移動取得シーケンスのハードウェア完全管理
- プログラム可能な電荷移動周波数
- プログラム可能なサンプリングコンデンサの I/O ピン
- プログラム可能なチャンネルの I/O ピン
- 最大カウント値をプログラムできるので、チャンネル障害時の取得時間の延長を回避。
- 割り込み機能を持つ専用の取得完了フラグおよび最大カウントエラーフラグ
- 最高 3 本の静電容量検出チャンネルに対して 1 個のサンプリングコンデンサとすることで、システム構成部品の個数を削減。
- 近接、タッチキー、リニア、およびロータリタイプのタッチセンサに対応。
- STMTouch タッチ検出ファームウェアライブラリで動作。

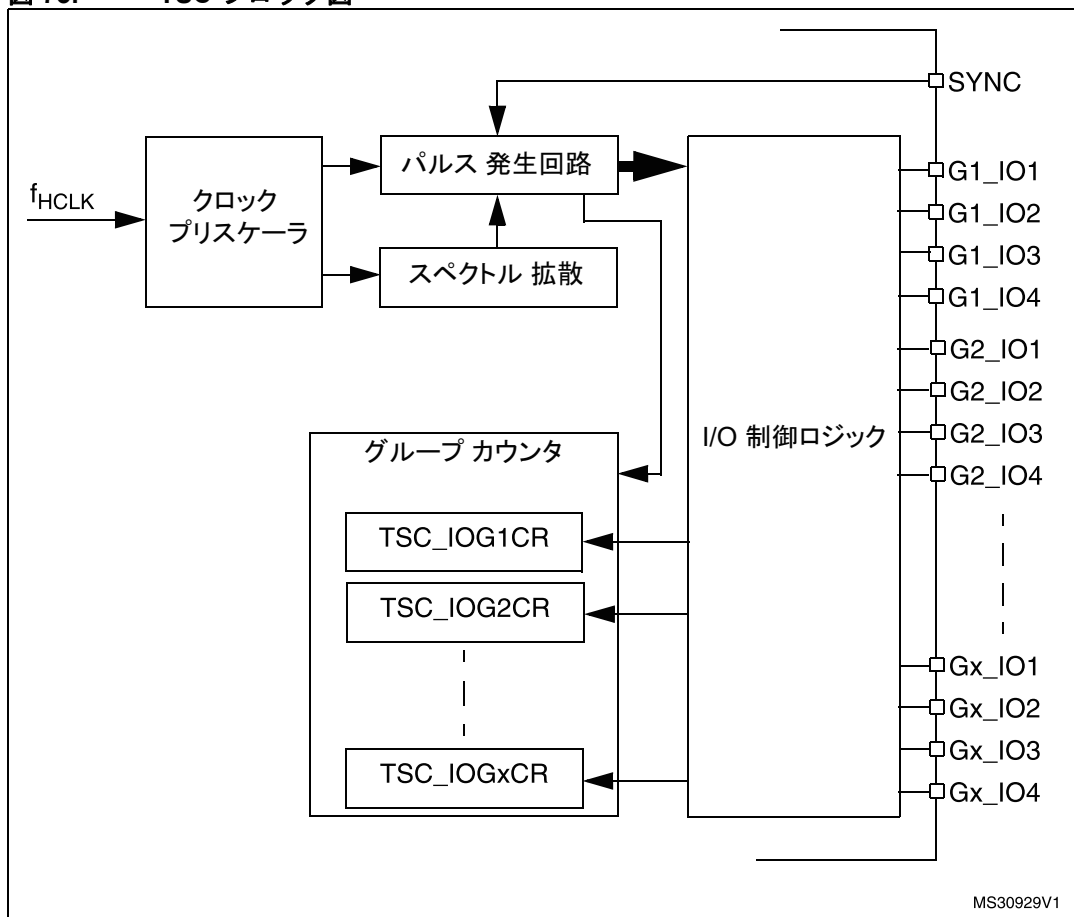
注： 静電容量検出チャンネルの本数は、パッケージのサイズ、および使用可能なI/Oがあるかどうかによって決まります。

18.3 TSC の機能説明

18.3.1 TSC ブロック図

タッチセンシングコントローラのブロック図を [図 73 : TSC ブロック図](#) に示します。

図 73. TSC ブロック図



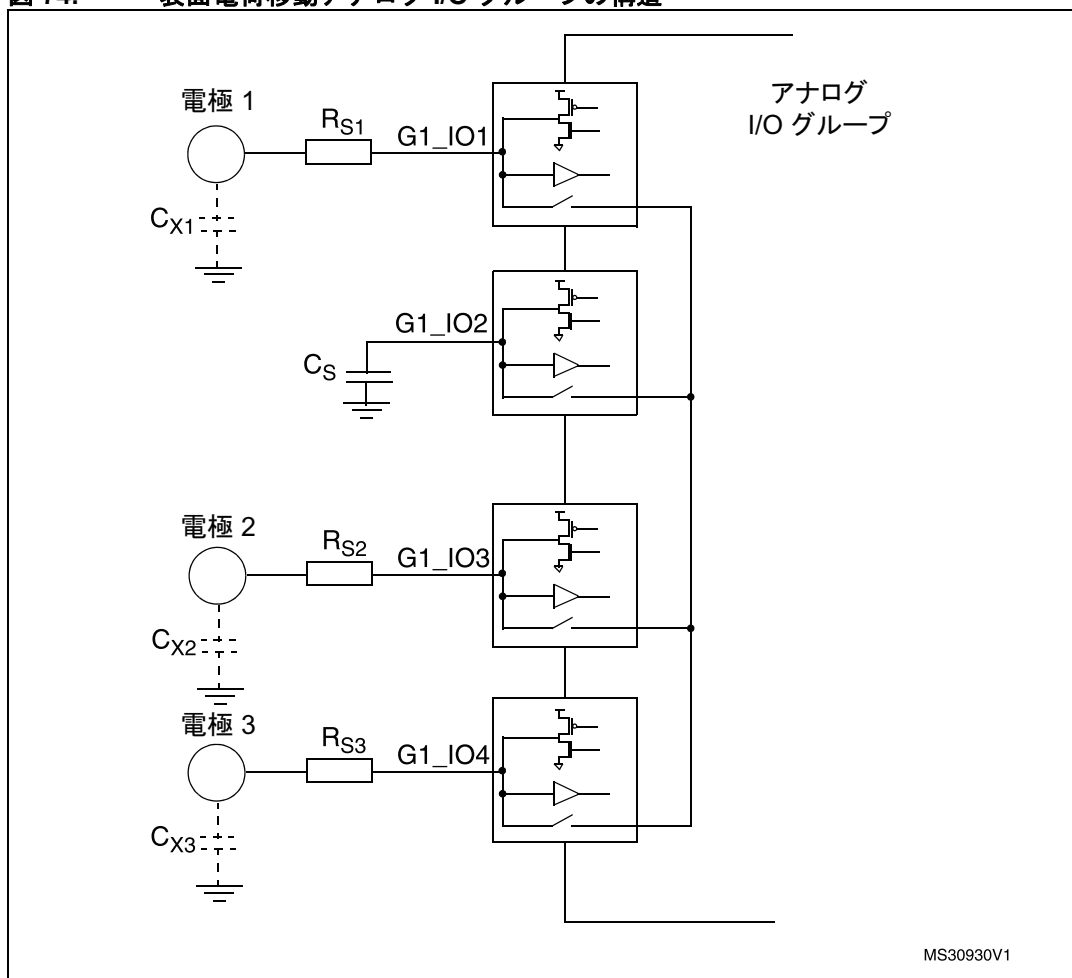
18.3.2 表面電荷移動取得の概要

表面電荷移動取得は、静電容量を計測するための実績のある、安定した効果的な方法です。この方法では、シングルエンド電極型で動作するために最低限必要な数の外部コンポーネントが使用されます。この取得は、4 つの GPIO から成るアナログ I/O グループを中心に設計されています ([図 74](#) を参照)。いくつかのアナログ I/O グループを使用することができ、それにより複数の静電容量検出チャネルを同時に取得し、サポートできる静電容量検出チャネルの数を増やすことができます。1 つのアナログ I/O グループ内で行われる静電容量検出チャネルの取得はシーケンシャルです。

GPIO のうちの 1 つは、サンプリングコンデンサ C_S 専用です。1 つのアナログ I/O グループでは、一度に 1 つのサンプリングコンデンサ I/O だけを有効にしてください。

残りの GPIO は電極に使用され、一般にチャネルと呼ばれます。特定のニーズ (近接検出など) については、1 つのアナログ I/O グループで同時に 1 つ以上のチャネルを有効にすることができます。

図 74. 表面電荷移動アナログ I/O グループの構造



注: Gx IOy の x はアナログ I/O グループの番号、y は選択したグループ内での GPIO の番号を表します。

表面電荷移動取得の基本動作は、電極の静電容量 (C_X) を充電することと、蓄積された電荷の一部をサンプリングコンデンサ (C_S) へ移動することです。このシーケンスは、 C_S の電圧が所定の閾値(弊社の場合、 V_{IH}) に達するまで繰り返されます。閾値に達するために必要な電荷移動の回数は、電極の静電容量の大きさで直接表されます。

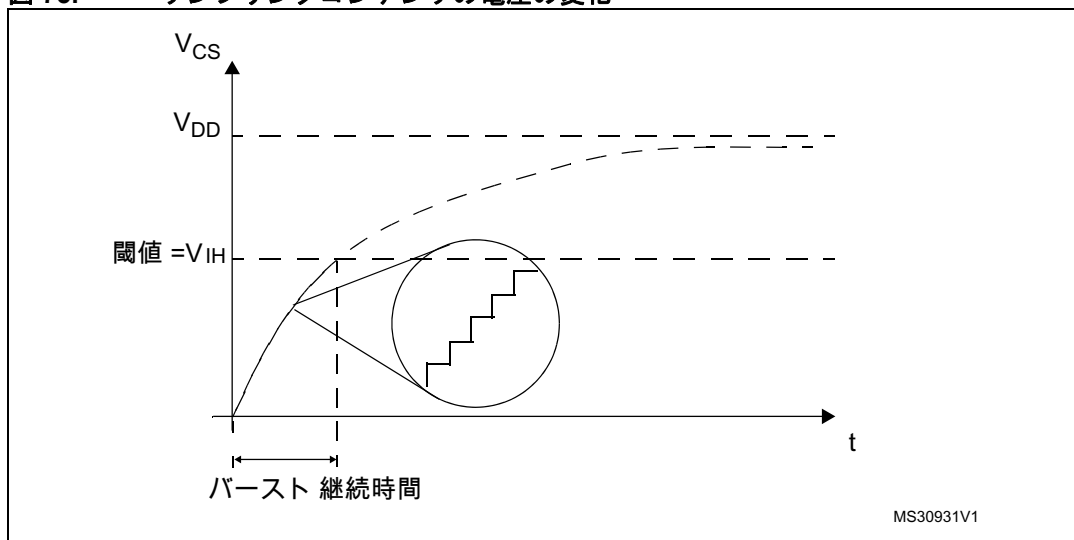
表 68 に、静電容量検出チャンネル 1 の電荷移動取得シーケンスの詳細を示します。 C_S の電圧が所定の閾値に達するまで、状態 3~7 が繰り返されます。その他のチャンネルの取得にも同じシーケンスが適用されます。電極のシリアルレジスタ R_S によって、この方法の ESD（静電気放電）イミュニティが向上します。

表 68. 取得シーケンスの概要

| 状態 | G1_IO1 (電極) | G1_IO2 (サンプリング) | G1_IO3 (電極) | G1_IO4 (電極) | 状態の説明 |
|----|----------------------------------|-------------------------------------|--------------------------|----------------|--|
| 1 | 入力 フローティング (アナログ スイッチ閉) | 出力オープン ドレインロー (アナログ スイッチ閉) | 入力フローティング (アナログスイッチ閉) | | すべての C _X および C _S を放電。 |
| 2 | 入力フローティング | | | | デッドタイム |
| 3 | 出力プッシュ プルハイ | 入力フローティング | | | C _{X1} を充電。 |
| 4 | 入力フローティング | | | | デッドタイム |
| 5 | 入力フローティング (アナログス イッチ閉) | | 入力フローティング | | C _{X1} から C _S への 電荷移動 |
| 6 | 入力フローティング | | | | デッドタイム |
| 7 | 入力フローティング | | | | C _S の電圧を測定。 |

サンプリングコンデンサ C_S の電圧の経時変化を以下に示します。

図 75. サンプリングコンデンサの電圧の変化



18.3.3 リセットおよびクロック

TSC クロックソースは AHB クロック (HCLK) です。2 個のプログラム可能なプリスケアラを使用して、パルス発生回路およびスペクトル拡散内部クロックを生成します。

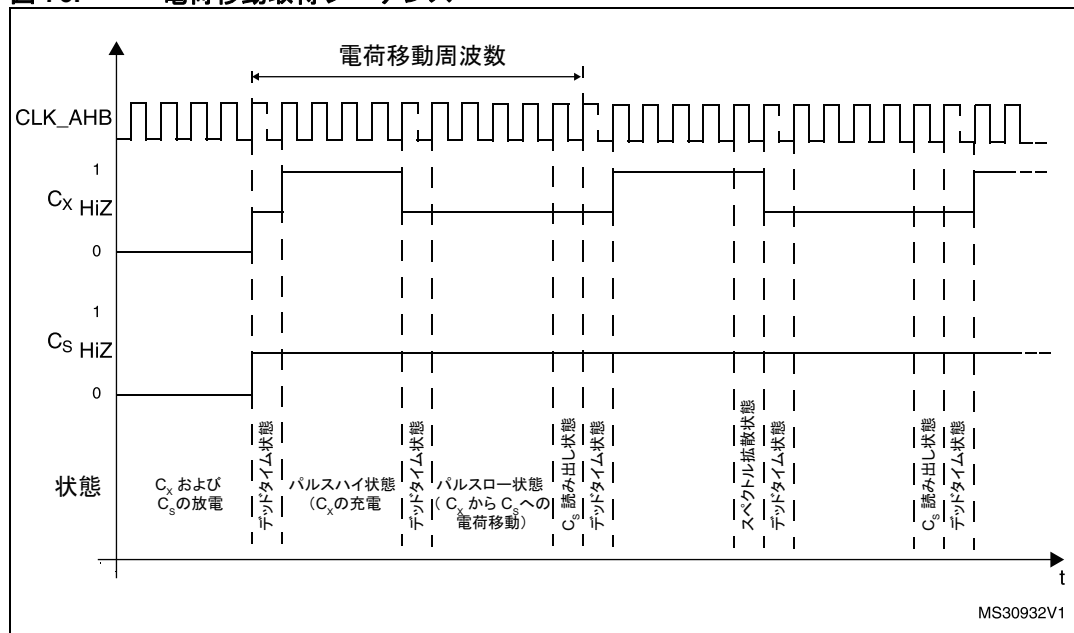
- パルス発生回路クロック (PGCLK) は、TSC_CR レジスタの PGPSCLK[2:0] ビットを使用して定義されます。
- スペクトル拡散クロック (SSCLK) は、TSC_CR レジスタの SSPSC ビットを使用して定義されます。

リセットおよびクロックコントローラ (RCC) は、タッチセンシングコントローラのクロックを有効にし、このペリフェラルをリセットするための専用ビットを提供します。詳細については、[セクション 7: リセットおよびクロック制御 \(RCC\)](#) を参照してください。

18.3.4 電荷移動取得シーケンス

電荷移動取得シーケンスの例を [図 76](#) に示します。

図 76. 電荷移動取得シーケンス



より高い柔軟性を得るために、電荷移動周波数は完全に設定可能です。パルスのハイ状態 (C_X の充電) およびパルスのロー状態 (C_X から C_S への電荷の移動) のどちらも、継続時間は TSC_CR レジスタの CTPH[3:0] ビットおよび CTPL[3:0] ビットを使用して定義することができます。パルスのハイ状態およびロー状態の継続時間の標準範囲は、500 ns ~ 2 μ s です。電極の静電容量を正確に測定できるようにするには、 C_X が常にフル充電となるように、パルスのハイ状態の継続時間を設定する必要があります。

電荷移動取得シーケンスを最適化するため、パルスのハイ状態とロー状態の間にデッドタイムが挿入されます。デッドタイムとは、サンプリングコンデンサの I/O とチャネルの I/O の両方が入力フローティング状態にある時間のことです。この状態の継続時間は HCLK の 2 周期です。

スペクトル拡散機能が有効であれば、パルスのハイ状態の最後に SSCLK クロックの周期の変数が追加されます。

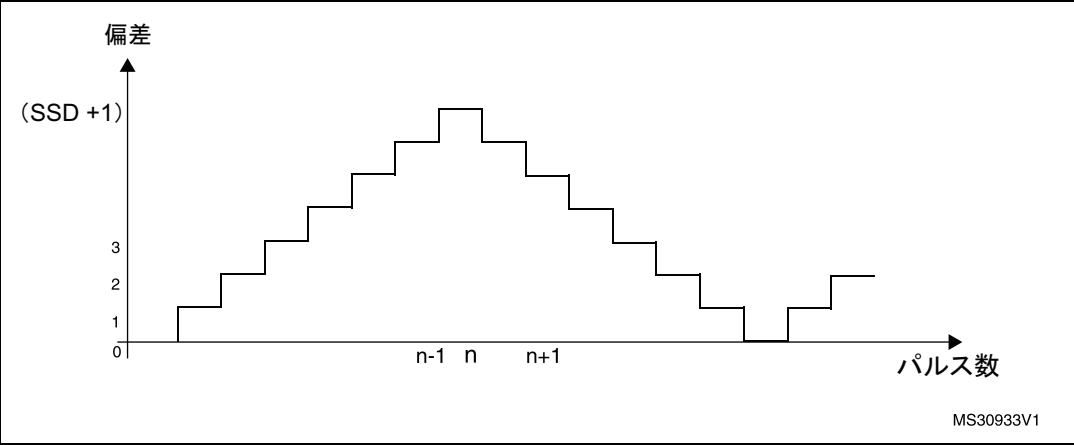
サンプリングコンデンサの I/O の読み出しは、 C_S の電圧が所定の閾値に達したかどうかを確認するために、パルスのロー状態の最後に行われ、その継続時間は HCLK の 1 周期分です。

18.3.5 スペクトル拡散機能

スペクトル拡散機能により、電荷移動周波数の変動を生成することができます。これは、ノイズの多い環境において電荷移動取得の安定性を向上させ、さらに誘導放出を低減するために行われます。周波数の最大変動幅は、公称電荷移動周期の 10～50 % です。たとえば、公称電荷移動周波数が 250 KHz (4 μ s) の場合、典型的なスペクトル拡散偏差は 10 % (400 ns) であり、この値から最小電荷移動周波数 (227 KHz 以下) が導き出されます。

実際には、スペクトル拡散とは、以下に示す原理を用いて、SSCLK 周期の変数を パルスのハイ状態に追加することを言います。

図 77. スペクトル拡散の変動の原理



次の表に、異なるHCLK 設定での最大周波数偏差を示します。

表 69. スペクトル拡散偏差と AHB クロック周波数

| f_{HCLK} | スペクトル拡散の刻み | スペクトル拡散最大偏差 |
|------------|------------|-------------|
| 24 MHz | 41.6 ns | 10666.6 ns |
| 32 MHz | 27.7 ns | 7111.1 ns |

スペクトル拡散機能は、TSC_CR レジスタの SSE ビットを使用して有効／無効にできます。周波数偏差は、TSC_CR レジスタの SSPSC および SSD[6:0] ビットを介して、デバイス HCLK クロック周波数および選択された電荷移動周波数を収容できるように設定することも可能です。

18.3.6 最大カウントエラー

最大カウントエラーは、静電容量検出チャネルの障害により取得時間が長くなるのを防ぎます。アナログ I/O グループのカウンタに対して最大カウント値を指定するのがこのエラーです。この最大カウント値は TSC_CR レジスタの MCV[2:0] ビットを使用して指定されます。取得グループのカウンタがこの最大値に到達するとすぐに、処理中の取得は停止し、取得完了 (EOAF ビット) および最大カウントエラー (MCEF ビット) の両方のフラグがセットされます。対応する取得完了 (EOAIE ビット) や最大カウントエラー (MCEIE ビット) の割り込みイネーブルビットがセットされている場合は、割り込みも生成することができます。

18.3.7 サンプリングコンデンサ I/O および チャネル I/O のモード選択

タッチセンシングコントローラによる GPIO の制御を可能にするには、標準の GPIO レジスタおよび GPIOxAFR レジスタを介して、対応するオルタネート機能を有効にする必要があります。

TSC によって制御される GPIO モードは、TSC_IOSCR および TSC_IOCCR レジスタを使用して定義されます。

処理中の取得がない場合、タッチセンシングコントローラによって制御されるすべての I/O はデフォルト状態です。取得の処理中は、未使用 I/O (サンプリングコンデンサ I/O としても、チャネル I/O としても定義されない) のみがデフォルト状態です。TSC_CR レジスタの IODEF ビットは、デフォルト状態にある I/O の設定を定義します。次の表にモードに応じた I/O の設定を示します。

表 70. モードおよび IODEF ビットの値に応じた I/O の状態

| IODEF ビット | 取得ステータス | 未使用 I/O モード | 電極 I/O モード | サンプリング コンデンサ I/O モード |
|-----------------------|---------|----------------|----------------|----------------------------|
| 0 (出力プッシュプル ロー) | なし | 出力プッシュプル ロー | 出力プッシュプル ロー | 出力プッシュプル ロー |
| 0 (出力プッシュプル ロー) | 処理中 | 出力プッシュプル ロー | - | - |
| 1 (入力 フローティング) | なし | 入力フローティング | 入力フローティング | 入力フローティング |
| 1 (入力 フローティング) | 処理中 | 入力フローティング | - | - |

未使用 I/O モード

未使用 I/O は、TSC ペリフェラルによって制御される GPIO に対応しますが、電極 I/O としても、サンプリングコンデンサ I/O としても定義されません。

サンプリングコンデンサ I/O モード

TSC ペリフェラルによるサンプリングコンデンサ I/O の制御を可能にするには、まず対応する GPIO をオルタネート出力オープンドレインモードにセットし、次に TSC_IOSCR レジスタ の対応する Gx_IOy ビットをセットする必要があります。

1 つのアナログ I/O グループでは、一度に 1 つのサンプリングコンデンサのみを有効にしてください。

チャネル I/O モード

TSC ペリフェラルによるチャネル I/O の制御を可能にするには、まず対応する GPIO をオルタネート出力プッシュプルモードにセットし、次に TSC_IOCCR レジスタ の対応する Gx_IOy ビットをセットする必要があります。

より高い等価電極表面が要求される、あるいは取得プロセスを高速化する近接検出では、同じアナログ I/O グループに属するいくつかのチャネルを有効にし、同時に取得することが可能です。

注： **取得フェーズの間、および TSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCR または TSC_IOCCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。**

18.3.8 取得モード

タッチセンシングコントローラには 2 種類の取得モードがあります。

- 通常の取得モード: TSC_CR レジスタの START ビットがセットされるとすぐに取得を開始します。
- 同期取得モード: TSC_CR レジスタの START ビットをセットすることで取得が有効になりますが、取得の開始は、立ち下がり／立ち上がりエッジおよび SYNC 入力ピンのハイレベルが検出されたときのみです。このモードは、CPU 負荷を追加することなく、静電容量検出チャンネルの取得を外部信号に同期させる場合に役立ちます。

TSC_ILOGCSR レジスタの GxE ビットで、どの I/O グループを有効 (対応するカウンタがカウントされる状態) にするかを指定します。無効なアナログ I/O グループの C_S 電圧は監視されず、このグループは取得完了フラグのトリガに加わりません。ただし、無効なアナログ I/O グループにいくつかのチャンネルが含まれる場合は、それらはパルスを出力します。

有効なアナログ I/O グループの C_S 電圧が所定の閾値に達すると、TSC_ILOGCSR レジスタの対応する GxS ビットがセットされます。すべての有効なアナログ I/O グループの取得が完了した (すべての有効なアナログ I/O グループのすべての GxS ビットがセットされた) 時点で、TSC_ISR レジスタの EOAF フラグがセットされます。TSC_IER レジスタの EOAIIE ビットがセットされている場合、割り込みリクエストが生成されます。

最大カウントエラーが検出された場合、処理中の取得は停止し、TSC_ISR レジスタの EOAF フラグと MCEIF フラグの両方がセットされます。対応するビット (TSCIER レジスタの EOAIIE および MCEIE ビット) がセットされていれば、両方のイベントに対して割り込みリクエストを生成することができます。最大カウントエラーが検出された場合、有効なアナログ I/O グループの残りの GxS ビットはセットされませんので注意してください。

割り込みフラグをクリアするには、TSC_ICR レジスタの対応する EOAIIC ビットと MCEIC ビットをセットする必要があります。

アナログ I/O グループのカウンタは、新しい取得が開始された時点でクリアされます。カウンタは、取得完了と同時に、対応するチャンネルで生成された電荷移動サイクルの数に更新されます。

18.3.9 I/O ヒステリシスおよびアナログスイッチの制御

より高い柔軟性を提供するために、タッチセンシングコントローラを使用することで、各 Gx_IOy のシュミットトリガヒステリシスおよびアナログスイッチを制御することもできます。タッチセンシングコントローラが有効であると仮定すれば、I/O 制御モードの状態 (標準の GPIO レジスタまたはその他のペリフェラルによって制御されている、など) にかかわらず、この制御は実行可能です。この制御は、異なる取得シーケンスを実行したり、他の目的のために実行する際に役立つことがあります。

システムのイミュニティを向上させる目的で、TSC_IOHCR レジスタの対応する Gx_IOy ビットをリセットすることによって、TSC によって制御される GPIO のシュミットトリガヒステリシスを無効にする必要があります。

18.3.10 静電容量検出 GPIO

次の表に、静電容量検出 GPIO の概要を示します。

表 71. 静電容量検出 GPIO

| グループ | 静電容量検出グループ名 | ピン名 | グループ | 静電容量検出グループ名 | ピン名 |
|------|-------------|------|------|-------------|------|
| 1 | TSC_G1_IO1 | PA0 | 5 | TSC_G5_IO1 | PB3 |
| | TSC_G1_IO2 | PA1 | | TSC_G5_IO2 | PB4 |
| | TSC_G1_IO3 | PA2 | | TSC_G5_IO3 | PB6 |
| | TSC_G1_IO4 | PA3 | | TSC_G5_IO4 | PB7 |
| 2 | TSC_G2_IO1 | PA4 | 6 | TSC_G6_IO1 | PB11 |
| | TSC_G2_IO2 | PA5 | | TSC_G6_IO2 | PB12 |
| | TSC_G2_IO3 | PA6 | | TSC_G6_IO3 | PB13 |
| | TSC_G2_IO4 | PA7 | | TSC_G6_IO4 | PB14 |
| 3 | TSC_G3_IO1 | PC5 | 7 | TSC_G7_IO1 | PC0 |
| | TSC_G3_IO2 | PB0 | | TSC_G7_IO2 | PC1 |
| | TSC_G3_IO3 | PB1 | | TSC_G7_IO3 | PC2 |
| | TSC_G3_IO4 | PB2 | | TSC_G7_IO4 | PC3 |
| 4 | TSC_G4_IO1 | PA9 | 8 | TSC_G8_IO1 | PC6 |
| | TSC_G4_IO2 | PA10 | | TSC_G8_IO2 | PC7 |
| | TSC_G4_IO3 | PA11 | | TSC_G8_IO3 | PC8 |
| | TSC_G4_IO4 | PA12 | | TSC_G8_IO4 | PC9 |

18.4 TSC 低電力モード

表 72. 低電力モードが TSC に与える影響

| モード | 説明 |
|---------|--|
| SLEEP | 影響しません。 TSC 割り込みによって、デバイスは SLEEP モードを終了します。 |
| STOP | TSC レジスタは停止状態です。 |
| STANDBY | STOP または STANDBY モードを終了するまで、TSC は動作を停止します。 |

18.5 TSC 割り込み

表 73. 割り込み制御ビット

| 割り込みイベント | イネーブル 制御ビット | イベントフラ グ | クリアフラグ ビット | SLEEPモー ドの終了 | STOPモー ドの終了 | STANDBY モードの終了 |
|-----------|----------------|-------------|---------------|-----------------|----------------|-------------------|
| 取得完了 | EOAIE | EOAIF | EOAIC | あり | なし | なし |
| 最大カウントエラー | MCEIE | MCEIF | MCEIC | あり | なし | なし |

18.6 TSC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの [セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスすることができます。

18.6.1 TSC 制御レジスタ (TSC_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-----------|-----|------------|-----|-----------|------|------|------|----------|-----|-----|-------|-------------|-----|-------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| CTPH[3:0] | | | | CTPL[3:0] | | | | SSD[6:0] | | | | | | SSE | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SSPSC | | PGPSC[2:0] | | Res. | Res. | Res. | Res. | MCV[2:0] | | | IODEF | SYNC POL | AM | START | TSCE |
| r/w | r/w | r/w | r/w | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:28 **CTPH[3:0]** : 電荷移動パルスハイ

これらのビットは、ソフトウェアによってセット/クリアされます。これらのビットは、電荷移動パルスのハイ状態 (C_X の充電) の継続時間を定義します。

0000 : 1x t_{PGCLK}

0001 : 2x t_{PGCLK}

...

1111 : 16x t_{PGCLK}

注 : 取得処理中はこれらのビットを変更しないでください。

ビット 27:24 **CTPL[3:0]** : 電荷移動パルスロー

これらのビットは、ソフトウェアによってセット/クリアされます。これらのビットは、電荷移動パルスのロー状態 (C_X から C_S への電荷の移動) の継続時間を定義します。

0000 : 1x t_{PGCLK}

0001 : 2x t_{PGCLK}

...

1111 : 16x t_{PGCLK}

注 : 取得処理中はこれらのビットを変更しないでください。

ビット 23:17 **SSD[6:0]** : スペクトル拡散偏差

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、SSCLK クロックの周期の変数を電荷移動パルスのハイ状態に追加する際のスペクトル拡散偏差を定義します。

0000000 : $1 \times t_{SSCLK}$
0000001 : $2 \times t_{SSCLK}$
...
1111111 : $128 \times t_{SSCLK}$

注: 取得処理中はこれらのビットを変更しないでください。

ビット 16 **SSE** : スペクトル拡散イネーブル

このビットは、スペクトル拡散機能を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : スペクトル拡散機能は無効です。
1 : スペクトル拡散機能は有効です。

注: 取得処理中はこのビットを変更しないでください。

ビット 15 **SSPSC** : スペクトル拡散プリスケアラ

このビットはソフトウェアによってセット／クリアされます。このビットは、スペクトル拡散クロック (SSCLK) の生成に使用される AHB クロック分周器を選択します。

0 : f_{HCLK}
1 : $f_{HCLK} / 2$

注: 取得処理中はこのビットを変更しないでください。

ビット 14:12 **PGPSC[2:0]** : パルス生成回路プリスケアラ

これらのビットはソフトウェアによってセット／クリアされます。これらのビットは、パルス発生回路クロック (PGCLK) の生成に使用される AHB クロック分周器を選択します。

000 : f_{HCLK}
001 : $f_{HCLK} / 2$
010 : $f_{HCLK} / 4$
011 : $f_{HCLK} / 8$
100 : $f_{HCLK} / 16$
101 : $f_{HCLK} / 32$
110 : $f_{HCLK} / 64$
111 : $f_{HCLK} / 128$

注: 取得処理中はこれらのビットを変更しないでください。

ビット 11:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:5 **MCV[2:0]** : 最大カウント値

これらのビットは、ソフトウェアによってセット／クリアされます。これらのビットは、最大カウントエラーが生成される前に生成され得る電荷移動パルスの最大数を定義します。

000 : 255
001 : 511
010 : 1023
011 : 2047
100 : 4095
101 : 8191
110 : 16383
111 : 予約済み

注: 取得処理中はこれらのビットを変更しないでください。

- ビット 4 **IODEF** : I/O デフォルトモード
- このビットはソフトウェアによってセット／クリアされます。このビットは、処理中の取得がない場合に、すべての TSC I/O の設定を定義します。処理中の取得がある場合、このビットは、すべての未使用 I/O (サンプリングコンデンサ I/O やチャネル I/O として定義されていない) の設定を定義します。
- 0 : I/O は強制的に出力プッシュプルローにされます。
- 1 : I/O は入力フローティング状態です。
- 注 :** 取得処理中はこのビットを変更しないでください。
- ビット 3 **SYNCPOL** : 同期ピンの極性
- このビットは、同期入力ピンの極性を選択するために、ソフトウェアによってセット／クリアされます。
- 0 : 立ち下がりエッジのみ
- 1 : 立ち上がりエッジおよびハイレベル
- ビット 2 **AM** : 取得モード
- このビットは、取得モードを選択するために、ソフトウェアによってセット／クリアされます。
- 0 : 通常の取得モード (START ビットがセットされた直後に取得を開始)
- 1 : 同期取得モード (START ビットがセットされ、かつ選択された信号が SYNC 入力ピンで検出された場合に、取得を開始)
- 注 :** 取得処理中はこのビットを変更しないでください。
- ビット 1 **START** : 新しい取得を開始します。
- このビットは、新しい取得を開始するために、ソフトウェアによってセットされます。このビットは、取得が完了した直後にハードウェアによって、または処理中の取得をキャンセルするためにソフトウェアによってクリアされます。
- 0 : 取得は開始されていません。
- 1 : 新しい取得を開始します。
- ビット 0 **TSCE** : タッチセンシングコントローライネーブル
- このビットは、タッチセンシングコントローラを有効／無効にするために、ソフトウェアによってセット／クリアされます。
- 0 : タッチセンシングコントローラは無効です。
- 1 : タッチセンシングコントローラは有効です。
- 注 :** タッチセンシングコントローラが無効の場合、TSC レジスタの設定は無効です。

18.6.2 TSC 割り込み有効レジスタ (TSC_IER)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEIE** : 最大カウントエラー割り込みイネーブル

このビットは、最大カウントエラーの割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

- 0 : 最大カウントエラー割り込みは無効です。
- 1 : 最大カウントエラー割り込みは有効です。

ビット 0 **EOAIE** : 取得完了割り込みイネーブル

このビットは、取得完了割り込みを有効／無効にするために、ソフトウェアによってセット／クリアされます。

- 0 : 取得完了割り込みは無効です。
- 1 : 取得完了割り込みは有効です。

18.6.3 TSC 割り込みクリアレジスタ (TSC_ICR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEIC** : 最大カウントエラー割り込みクリア

このビットは、最大カウントエラーフラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

- 0 : 影響しません。
- 1 : TSC_ISR レジスタの対応する MCEF をクリアします。

ビット 0 **EOAIC** : 取得完了割り込みクリア

このビットは、取得完了フラグをクリアするためにソフトウェアによってセットされ、フラグがリセットされた時点でハードウェアによってクリアされます。“0”を書き込んでも、ビットの値は変化しません。

- 0 : 影響しません。
- 1 : TSC_ISR レジスタの対応する EOAF をクリアします。

18.6.4 TSC 割り込みステータスレジスタ (TSC_ISR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

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

ビット 31:2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **MCEF** : 最大カウントエラーフラグ

このビットは、アナログ I/O グループのカウンタが指定された最大カウント値に達すると、ハードウェアによってセットされます。このビットは、ソフトウェアで TSC_ICR レジスタの MCEIC ビットに“1”を書き込むことによってクリアされます。

- 0 : 最大カウントエラー (MCE) は検出されていません。
- 1 : 最大カウントエラー (MCE) が検出されました。

ビット 0 **EOAF** : 取得完了フラグ

このビットは、すべての有効なグループの取得が完了した (すべての有効なアナログ I/O グループのすべての GxS ビットがセットされた、または最大カウントエラーが検出された) 時点で、ハードウェアによってセットされます。このビットは、ソフトウェアで TSC_ICR レジスタの EOAIIC ビットに“1”を書き込むことによってクリアされます。

- 0 : 取得は処理中または開始されていません。
- 1 : 取得は完了しました。

18.6.5 TSC I/O ヒステリシス制御レジスタ (TSC_IOHCR)

アドレスオフセット : 0x10

リセット値 : 0xFFFF FFFF

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| G8_IO4 | G8_IO3 | G8_IO2 | G8_IO1 | G7_IO4 | G7_IO3 | G7_IO2 | G7_IO1 | G6_IO4 | G6_IO3 | G6_IO2 | G6_IO1 | G5_IO4 | G5_IO3 | G5_IO2 | G5_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| G4_IO4 | G4_IO3 | G4_IO2 | G4_IO1 | G3_IO4 | G3_IO3 | G3_IO2 | G3_IO1 | G2_IO4 | G2_IO3 | G2_IO2 | G2_IO1 | G1_IO4 | G1_IO3 | G1_IO2 | G1_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **Gx_IOy** : Gx_IOy シュミットトリガヒステリシスモード

これらのビットは、Gx_IOy シュミットトリガヒステリシスを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : Gx_IOy シュミットトリガヒステリシスは無効です。

1 : Gx_IOy シュミットトリガヒステリシスは有効です。

注 : これらのビットは、I/O 制御モードの状態にかかわらず (標準の GPIO レジスタによって制御されていても)、I/O シュミットトリガヒステリシスを制御します。

18.6.6 TSC I/O アナログスイッチ制御レジスタ (TSC_IOASCR)

アドレスオフセット : 0x18

リセット値 : 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| G8_IO4 | G8_IO3 | G8_IO2 | G8_IO1 | G7_IO4 | G7_IO3 | G7_IO2 | G7_IO1 | G6_IO4 | G6_IO3 | G6_IO2 | G6_IO1 | G5_IO4 | G5_IO3 | G5_IO2 | G5_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| G4_IO4 | G4_IO3 | G4_IO2 | G4_IO1 | G3_IO4 | G3_IO3 | G3_IO2 | G3_IO1 | G2_IO4 | G2_IO3 | G2_IO2 | G2_IO1 | G1_IO4 | G1_IO3 | G1_IO2 | G1_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **Gx_IOy** : Gx_IOy アナログスイッチイネーブル

これらのビットは、Gx_IOy アナログスイッチを有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : Gx_IOy アナログスイッチは無効 (開) です。

1 : Gx_IOy アナログスイッチは有効 (閉) です。

注 : これらのビットは、I/O 制御モードの状態にかかわらず (標準の GPIO レジスタによって制御されていても)、I/O アナログスイッチを制御します。

18.6.7 TSC I/O サンプリング制御レジスタ (TSC_IOSCR)

アドレスオフセット : 0x20

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| G8_IO4 | G8_IO3 | G8_IO2 | G8_IO1 | G7_IO4 | G7_IO3 | G7_IO2 | G7_IO1 | G6_IO4 | G6_IO3 | G6_IO2 | G6_IO1 | G5_IO4 | G5_IO3 | G5_IO2 | G5_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| G4_IO4 | G4_IO3 | G4_IO2 | G4_IO1 | G3_IO4 | G3_IO3 | G3_IO2 | G3_IO1 | G2_IO4 | G2_IO3 | G2_IO2 | G2_IO1 | G1_IO4 | G1_IO3 | G1_IO2 | G1_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **Gx_IOy** : Gx_IOy サンプリングモード

これらのビットは、Gx_IOy をサンプリングコンデンサ I/O として設定するために、ソフトウェアによってセット/クリアされます。1 つのアナログ I/O グループでは、1 つの I/O だけをサンプリングコンデンサとして定義してください。

0 : Gx_IOy は未使用です。

1 : Gx_IOy はサンプリングコンデンサとして使用されます。

注 : 取得処理中はこれらのビットを変更しないでください。

取得フェーズの間、およびTSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOSCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。

18.6.8 TSC I/O チャネル制御レジスタ (TSC_IOCOCR)

アドレスオフセット : 0x28

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| G8_IO4 | G8_IO3 | G8_IO2 | G8_IO1 | G7_IO4 | G7_IO3 | G7_IO2 | G7_IO1 | G6_IO4 | G6_IO3 | G6_IO2 | G6_IO1 | G5_IO4 | G5_IO3 | G5_IO2 | G5_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| G4_IO4 | G4_IO3 | G4_IO2 | G4_IO1 | G3_IO4 | G3_IO3 | G3_IO2 | G3_IO1 | G2_IO4 | G2_IO3 | G2_IO2 | G2_IO1 | G1_IO4 | G1_IO3 | G1_IO2 | G1_IO1 |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **Gx_IOy** : Gx_IOy チャネルモード

これらのビットは、Gx_IOy をチャネル I/O として設定するために、ソフトウェアによってセット/クリアされます。

0 : Gx_IOy は未使用です。

1 : Gx_IOy はチャネルとして使用されます。

注 : 取得処理中はこれらのビットを変更しないでください。

取得フェーズの間、およびTSC ペリフェラルのオルタネート機能が有効でない場合でも、TSC_IOCOCR ビットがセットされたらすぐに、タッチセンシングコントローラによって、対応する GPIO アナログスイッチが自動的に制御されます。



18.6.9 TSC I/O グループ制御ステータスレジスタ (TSC_ILOGCSR)

アドレスオフセット : 0x30

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|-----|-----|-----|-----|-----|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | G8S | G7S | G6S | G5S | G4S | G3S | G2S | G1S |
| | | | | | | | | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | G8E | G7E | G6E | G5E | G4E | G3E | G2E | G1E |
| | | | | | | | | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:16 **GxS** : アナログ I/O グループ x ステータス

これらのビットは、対応する有効なアナログ I/O グループ x で取得が完了した時点で、ハードウェアによってセットされます。これらのビットは、新しい取得が開始した時点で、ハードウェアによってクリアされます。

0 : アナログ I/O グループ x での取得は、処理中または開始されていません。

1 : アナログ I/O グループ x での取得は完了しました。

注 : 最大カウントエラーが検出された場合、有効なアナログ I/O グループの残りの GxS ビットはセットされません。

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **GxE** : アナログ I/O グループ x イネーブル

これらのビットは、対応するアナログ I/O グループ x での取得を有効 (カウンタがカウントされる状態) にするために、ソフトウェアによってセット/クリアされます。

0 : アナログ I/O グループ x での取得は無効です。

1 : アナログ I/O グループ x での取得は有効です。

18.6.10 TSC I/O グループ x カウンタレジスタ (TSC_ILOGxCR) (x = 1~8)

アドレスオフセット : 0x30 + 0x04 x アナログ I/O グループ番号

リセット値 : 0x0000 0000

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

ビット 31:14 予約済みであり、リセット値に保持する必要があります。

ビット 13: **CNT[13:0]** : カウンタ値

これらのビットは、アナログ I/O グループ x の取得を完了させる (C_S の電圧が閾値に達する) ために、そこで生成された電荷移動サイクルの数を表します。

18.6.11 TSC レジスタマップ

表 74. TSC レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|------------|-----------|--------|--------|--------|-----------|--------|--------|--------|----------|--------|--------|--------|--------|--------|--------|--------|--------|------------|-----------|--------|--------|--------|--------|--------|--------------|--------|--------|--------|---------|--------|--------|--------|
| 0x0000 | TSC_CR | CTPH[3:0] | | | | CTPL[3:0] | | | | SSD[6:0] | | | | | | | SSE | SSPSC | PGPSC[2:0] | | | Res. | Res. | Res. | Res. | MCV [2:0] | | | IODEF | SYNCPOL | AM | START | TSCE |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0004 | TSC_IER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MCEIE | FOAIF | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | |
| 0x0008 | TSC_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MCEIC | FOAIC | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | |
| 0x000C | TSC_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MCEIF | FOAIF | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | |
| 0x0010 | TSC_IOHCR | G8_I04 | G8_I03 | G8_I02 | G8_I01 | G7_I04 | G7_I03 | G7_I02 | G7_I01 | G6_I04 | G6_I03 | G6_I02 | G6_I01 | G5_I04 | G5_I03 | G5_I02 | G5_I01 | G4_I04 | G4_I03 | G4_I02 | G4_I01 | G3_I04 | G3_I03 | G3_I02 | G3_I01 | G2_I04 | G2_I03 | G2_I02 | G2_I01 | G1_I04 | G1_I03 | G1_I02 | G1_I01 |
| | リセット値 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0x0014 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0018 | TSC_IOASCR | G8_I04 | G8_I03 | G8_I02 | G8_I01 | G7_I04 | G7_I03 | G7_I02 | G7_I01 | G6_I04 | G6_I03 | G6_I02 | G6_I01 | G5_I04 | G5_I03 | G5_I02 | G5_I01 | G4_I04 | G4_I03 | G4_I02 | G4_I01 | G3_I04 | G3_I03 | G3_I02 | G3_I01 | G2_I04 | G2_I03 | G2_I02 | G2_I01 | G1_I04 | G1_I03 | G1_I02 | G1_I01 |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x001C | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0020 | TSC_IOSCR | G8_I04 | G8_I03 | G8_I02 | G8_I01 | G7_I04 | G7_I03 | G7_I02 | G7_I01 | G6_I04 | G6_I03 | G6_I02 | G6_I01 | G5_I04 | G5_I03 | G5_I02 | G5_I01 | G4_I04 | G4_I03 | G4_I02 | G4_I01 | G3_I04 | G3_I03 | G3_I02 | G3_I01 | G2_I04 | G2_I03 | G2_I02 | G2_I01 | G1_I04 | G1_I03 | G1_I02 | G1_I01 |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0024 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0028 | TSC_IOCRR | G8_I04 | G8_I03 | G8_I02 | G8_I01 | G7_I04 | G7_I03 | G7_I02 | G7_I01 | G6_I04 | G6_I03 | G6_I02 | G6_I01 | G5_I04 | G5_I03 | G5_I02 | G5_I01 | G4_I04 | G4_I03 | G4_I02 | G4_I01 | G3_I04 | G3_I03 | G3_I02 | G3_I01 | G2_I04 | G2_I03 | G2_I02 | G2_I01 | G1_I04 | G1_I03 | G1_I02 | G1_I01 |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x002C | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0030 | TSC_IQGCSR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | G8S | G7S | G6S | G5S | G4S | G3S | G2S | G1S | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | G8E | G7E | G6E | G5E |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 |
| 0x0034 | TSC_IQG1CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0038 | TSC_IQG2CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x003C | TSC_IQG3CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



表 74. TSC レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-----------|----|----|----|---|---|---|---|---|---|---|---|---|---|
| 0x0040 | TSC_I0G4CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0044 | TSC_I0G5CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0048 | TSC_I0G6CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x004C | TSC_I0G7CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0050 | TSC_I0G8CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[13:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

19 AESハードウェアアクセラレータ (AES)

AES は、AES搭載のカテゴリ 2の マイクロコントローラでのみ使用できます。

19.1 概要

AES ハードウェアアクセラレータは、AES アルゴリズムを使用した暗号化および復号化データの両方に対して使用できます。次の規格に完全に準拠して実装されています。

- 連邦情報処理規格公報 (FIPS PUB 197, 2001 November 26) によって規定されている高度暗号化標準 (AES : Advanced Encryption Standard)。

アクセラレータは 128 ビットのキー長を使用して 128 ビットブロックの暗号化と復号化を行います。キー派生も実行できます。複数のデータブロックを同じキーを使用して処理するときに CPU または DMA による書き込み操作を最小限にするために、暗号化または復号化キーは内部レジスタに格納されます。

デフォルトでは、ECB (電子コードブック) モードが選択されます。CBC (暗号ブロック連鎖) または CTR (カウンタ) モードの連鎖アルゴリズムもハードウェアによってサポートされます。

AES は、着信データと発信データの DMA 転送をサポートします (2 つの DMA チャンネルが必要です)。

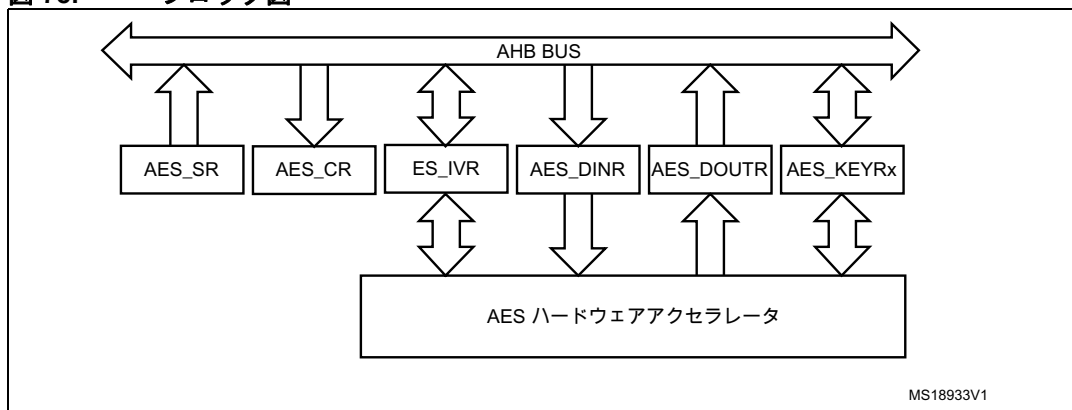
19.2 AES の主な機能

- AES Rijndael ブロック暗号アルゴリズムを使用した暗号化／復号化
- NIST FIPS 197 に準拠した AES 暗号化／復号化アルゴリズムの実装
- 暗号化または派生キーを格納するための内部 128 ビットレジスタ (4x 32 ビットレジスタ)
- ECB (電子コードブック)、CBC (暗号ブロック連鎖)、および CTR (カウンタ) モードをサポート
- キースケジューラ
- 復号化のためのキー派生
- 128 ビットデータブロックの処理
- 128 ビットのキー長
- 1 つの 128 ビットブロックの暗号化または復号化に 213 クロックサイクル (入力および出力フェーズを含む)
- 1x32 ビットの入力バッファと 1x32 ビットの出カバッファ
- 32 ビットのデータ幅のみをサポートするレジスタアクセス
- AES が CBC モードで設定されているときには初期化ベクトル用に 1 つの 128 ビットレジスタ、または CTR モードが選択されているときには 32 ビットのカウンタ初期化
- 2 つのチャンネル (着信データ用と発信データ用) を使用したダイレクトメモリアクセス (DMA) のサポートによる自動データフロー制御

19.3 AES の機能詳細

図 78 に、AES アクセラレータのブロック図を示します。

図 78. ブロック図



AES アクセラレータは 128 ビット（4 ワード）のデータブロックを 128 ビットの長さのキーを使用して処理し、CBC または CTR 連鎖モードが選択されたときには初期化ベクトルも使用します。

4 つの動作モードを備えています。

- モード 1：AES_KEYRx レジスタに格納された暗号化キーを使用した暗号化。
- モード 2：AES を有効にする前に AES_KEYRx レジスタに格納された暗号化キーから処理されるキー派生の最後に、このレジスタに内部格納されるキー派生。このモードは、AES 連鎖モード選択に依存しません。
- モード 3：AES_KEYRx レジスタに格納された特定の（あらかじめ計算された）復号化キーを使用した復号化。
- モード 4：AES_KEYRx レジスタに格納された暗号化キーを使用したキー派生 + 復号化（連鎖アルゴリズムを実行するために AES がカウンタモードに設定されたときには使用されません）。

動作モードは、AES_CR レジスタの MODE[1:0] ビット をプログラムすることによって選択されます。モードの変更は、AES が無効なとき（AES_CR レジスタのビット EN=0）に行う必要があります。KEY レジスタ（AES_KEYRx）は、AES を有効にする前に格納される必要があります。

ECB、CBC、または CTR モードのいずれを暗号ソリューションに使用するかを選択するには、AES が無効なとき（AES_CR レジスタのビット EN=0）に、AES_CR レジスタと AES_IVR レジスタ（CBC および CTR 連鎖モードの場合のみ使用）の CHMOD[1:0] ビット に書き込む必要があります。

AES は有効（ビット EN=1）になると、入力フェーズになり、ソフトウェアがモード 1、3、または 4 の場合に AES_DINR に入力データワード（4 ワード）を書き込むのを待ちます。データは、プレーンテキストメッセージまたは暗号メッセージのいずれかに対応します。データでエンタープライズされたキーを AES プロセッサに送信するために、AES_DINR レジスタへの 2 つの連続した書き込みの間にウェイトサイクルが自動的に挿入されます。

モード 2 の場合、キー派生処理は、AES_CR レジスタの EN ビットがセットされた直後に開始されます。AES を有効にする前に、AES_KEYRx レジスタに暗号化された KEY がロードされる必要があります。キー派生処理の終了時（CCF フラグがセットされます）、派生キーは AES_KEYRx レジスタで使用可能になり、AES はハードウェアによって無効にされます。このモードでは、AES が有効で、CCF フラグがハードウェアによって 1 にセットされない限り、AES_KEYRx レジスタを読み出してはなりません。

AES_SR レジスタのステータスフラグ CCF（計算完了フラグ）は、計算フェーズが完了するとセットされます。AES_CR レジスタのビット CCFIE=1 の場合、割り込みが生成されます。ソフトウェア

は AES_DOUTR レジスタから（モード 1、3、4 の場合）、または AES_KEYRx レジスタから（モード 2 が選択された場合）データを読み出すことができます。

AES_DOUTR レジスタの読み出しは DMA によって自動的に管理され、計算フェーズの終了時にソフトウェアの動作を必要としないため、AES_CR レジスタの DMAOUTEN = 1 のときには、フラグ CCF は意味を持ちません。

モード 1、3、または 4 では、操作は出力フェーズの終了時にソフトウェアが AES_DOUTR レジスタから出力データ 4 ワードを連続的に読み出したときに終了します。モード 2（キー派生モード）では、データは AES_KEYRx レジスタに自動的に格納され、AES はハードウェアによって無効にされます。その後、ソフトウェアはモード 3（復号化モード）を選択してから AES を有効にして、この派生キーを使用して復号化を開始できます。

入力および出力フェーズでは、ソフトウェアはデータバイトを連続的に読み出す、または書き込む必要がありますが（モード 2 の場合を除く）、AES は各読み出しまたは書き込み操作間の遅延発生（この時点で別の割り込みを処理している場合など）に対して耐性があります。

AES_SR レジスタの RDERR および WRERR フラグは、予期しない読み出しまたは書き込み操作が検出されたときにセットされます。AES_CR レジスタの ERRIE ビットがセットされている場合、割り込みが生成されます。エラー検出後も AES は無効にならず、通常通りに処理を続行します。

汎用 DMA を使用して入力ワードを書き込み、出力ワードを読み出すことも可能です（図 93 および図 94 を参照）。

AES_CR レジスタの EN ビットをリセットすることによって、いつでも AES を再初期化できます。その後、EN=1 をセットすることによって AES を始めから再開でき、最初の入力データバイトが書き込まれるのを待ちます（モード 2 の場合、キー派生処理は EN ビットがセットされると同時に、AES_KEYRx レジスタに格納された値から開始されます）。

19.4 暗号化キーと復号化キー

AES_KEYRx レジスタは暗号化または復号化キーを格納するために使用されます。この 4 つのレジスタは、リトルエンディアン設定で構成されています。レジスタ AES_KEYR0 にはキーの 32 ビットの LSB をロードする必要があります。結果として、AES_KEYR3 には 128 ビットのキーの 32 ビットの MSB をロードする必要があります。

暗号化または復号化のキーは、AES が無効なとき（AES_CR レジスタの EN = 0）に、これらのレジスタに格納される必要があります。これらのエンディアン形式は固定です。

モード 2（キー派生）では、AES_KEYRx に暗号化キーをロードする必要があります。その後、AES を有効にする必要があります。計算フェーズの終了時、派生キーは AES_KEYRx レジスタに自動的に格納され、以前の暗号化キーを上書きします。AES は、派生キーが使用可能になると、ハードウェアによって無効にされます。ソフトウェアが AES をモード 3（復号化モード）に切り替える必要がある場合、AES_KEYRx レジスタの内容が（以前にモード 2 によって計算された）復号化キーに対応する場合、書き込む必要はありません。

モード 4（キー派生 + 復号化）では、AES_KEYRx レジスタは暗号化キーだけを含みます。派生キーは、これらのレジスタへの書き込みなしで、内部で計算されます。

19.5 AES 連鎖アルゴリズム

AES ハードウェアでは 3 つのアルゴリズムがサポートされ、AES が無効なとき（ビット EN=0）、AES_CR レジスタの CHMOD[1:0] ビットによって選択できます。

- ECB（電子コードブック）
- CBC（暗号ブロック連鎖）
- CTR（カウンタモード）

19.5.1 ECB（電子コードブック）

これはデフォルトのモードです。このモードは、AES_IVR レジスタを使用しません。連鎖操作はありません。メッセージはブロックに分割され、各ブロックが個別に暗号化されます。

図 79 と 図 80 に、電子コードブックの暗号化と復号化の アルゴリズムの原理をそれぞれ示します。

図 79. ECB 暗号化モード

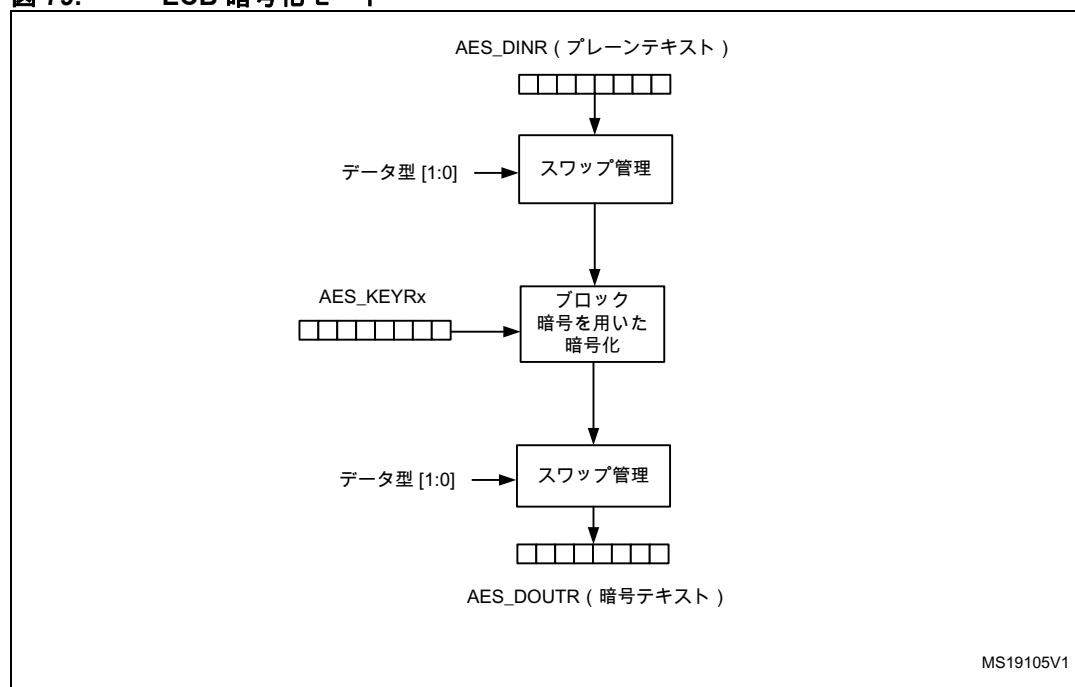
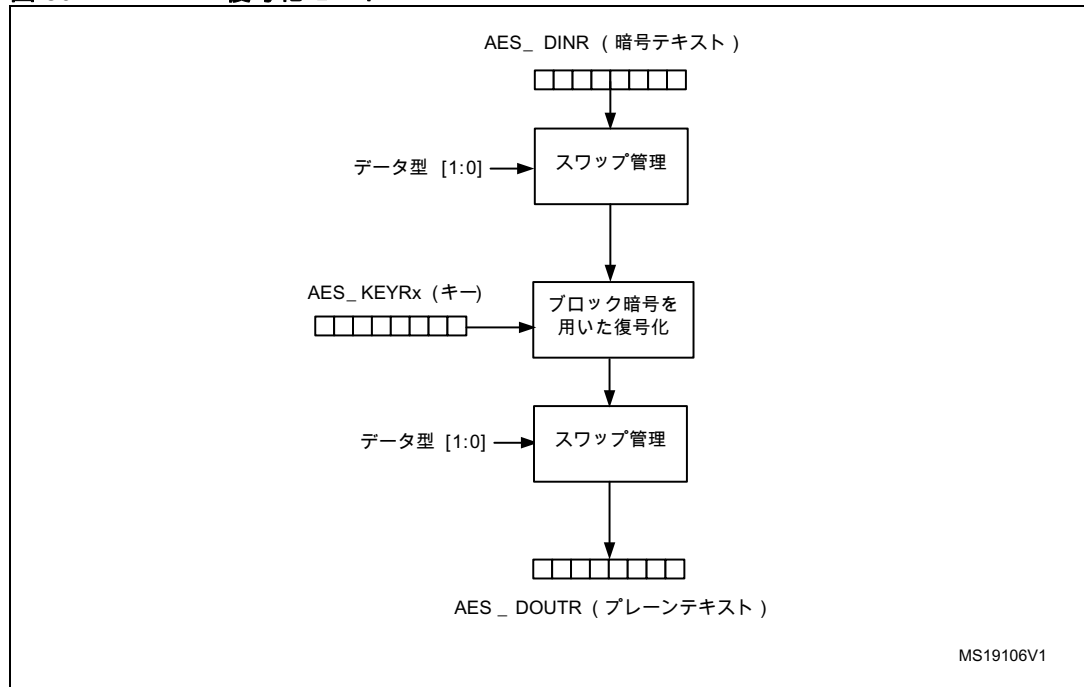


図 80. ECB 復号化モード



19.5.2 CBC (暗号ブロック連鎖)

CBC (暗号ブロック連鎖) モードでは、プレーンテキストの各ブロックが前の暗号テキストブロックと XOR されてから暗号化されます。各メッセージを一意にするために、最初のブロック処理時に初期化ベクトル (AES_IVRx) が使用されます。

暗号化モードでは管理ブロックの交換後に、復号化モードではその前に、初期化ベクトルが XOR されます (図 81 および 図 82 を参照)。

図 81. CBC モードの暗号化

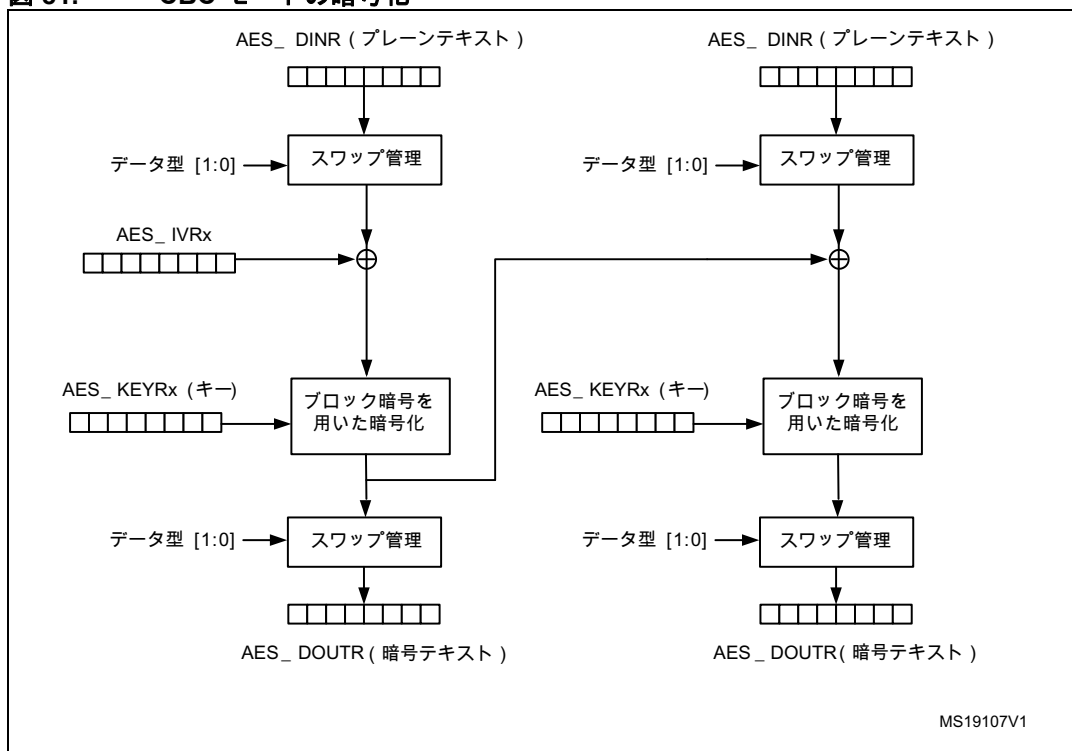
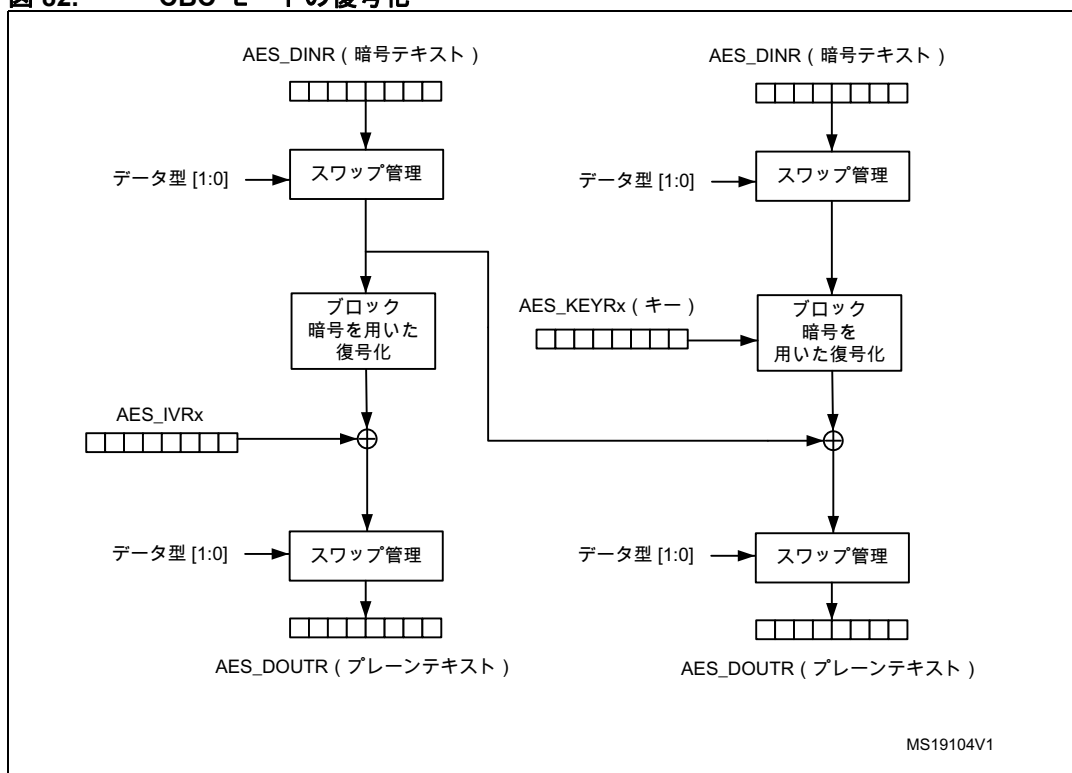


図 82. CBC モードの復号化



注: AES が有効なときに AES_IVR を読み出すと、値 0x00000000 が返されます。

特定のメッセージでのサスペンドモード

優先順位の高い別のメッセージを処理する必要がある場合には、メッセージをサスペンドすることが可能です。最も優先順位の高いメッセージの送信終了時、サスペンドされたメッセージを暗号化または復号化モードで再開できます。この機能は、AES_DOUTR および AES_DINR レジスタへの CPU アクセスによってデータ転送が行われたときだけ使用できます。DMA コントローラがデータ転送を管理しているときには、使用しないことが推奨されます。

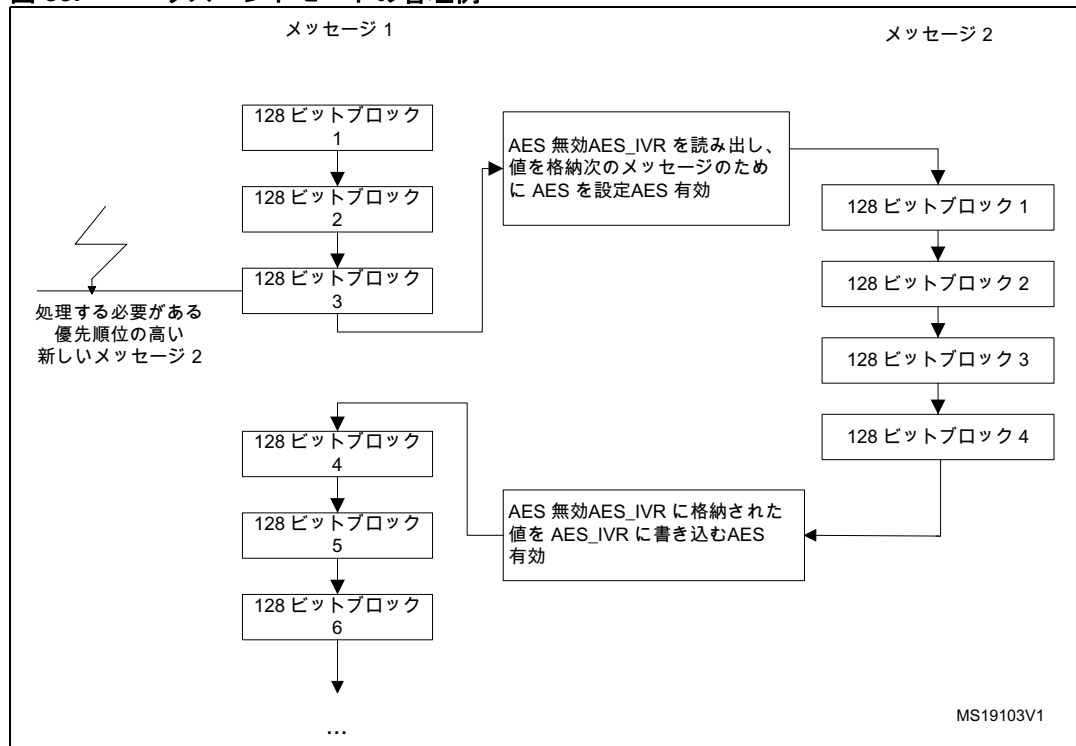
正しい操作のためには、ブロックの処理の終了時にメッセージをサスペンドする必要があります (AES_DOUTR レジスタの 4 番目の読み出しの後、次に処理するブロックの入力に対する次の AES_DINR 書き込みアクセスの前)。

AES_CR レジスタのビット EN = 0 を書き込んで、AES を無効にする必要があります。ソフトウェアは、メッセージ中断の前に、連鎖 XOR 操作に使用する最新の値を含んだ AES_IVRx を読み出す必要があります。この値は、中断されたメッセージを再開する必要があるときにすぐに再利用できるように、AES_IVRx レジスタに書き込むことによって格納される必要があります (AES が無効のときに)。これによって連鎖操作が途切れることはなく、AES が再び有効になるとすぐにメッセージ処理を再開して、次の 128 ビットデータブロックを送信できます。

この動作は、AES 設定 (暗号化または復号化モード) にかかわらず有効です。

図 83 に、メッセージ 1 より短く、優先順位の高いメッセージ 2 を送信するために、メッセージ 1 をサスペンドする例を示します。128 ビットブロックの処理の終了時、AES は無効になります。連鎖操作が中断されないように、AES_IVR レジスタは、後でメッセージ再開時に取得する値を格納するために読み戻されます。次に、メッセージ 2 を送信するように AES が設定され、処理を開始するために有効化されます。メッセージ 2 の処理の終了時、AES を再び無効にする必要があります。AES_IVRx レジスタに、前回メッセージ 1 が中断されたときに格納された値をロードする必要があります。次に、ソフトウェアは、メッセージ 1 を再開するために AES が有効になるとすぐに、ブロック 4 に対応する入力値から再開する必要があります。

図 83. サスペンドモードの管理例



19.5.3 CTR (カウンタモード)

カウンタモードでは、XOR 操作のノンス値に加えて、32 ビットカウンタが暗号テキストまたはプレーンテキストとともに使用されます (図 84 および 図 85 を参照)。

図 84. CTR モードの暗号化

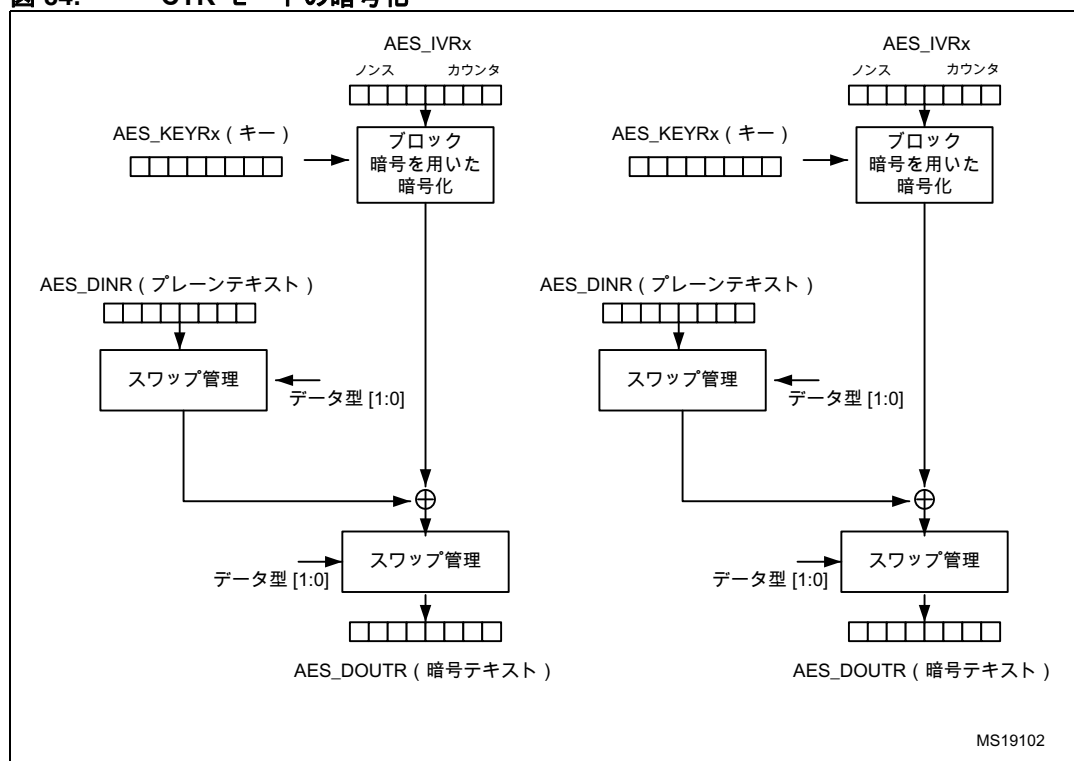
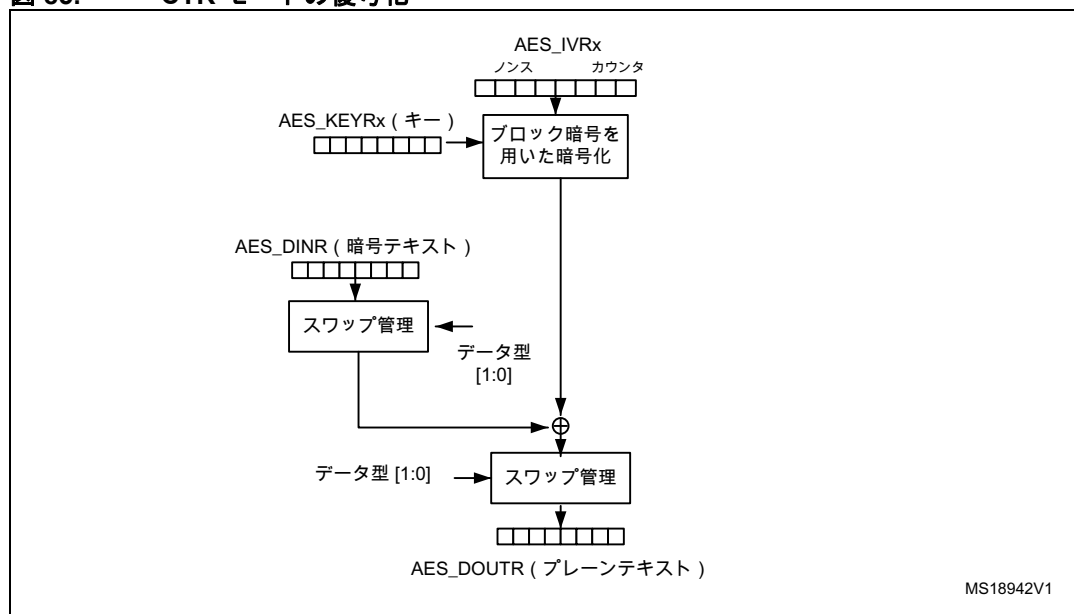
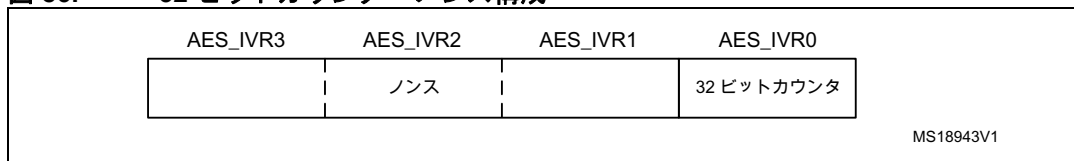


図 85. CTR モードの復号化



ノンス値と 32 ビットカウンタは、AES_IVRx レジスタからアクセスでき、[図 86](#) のように構成されます。

図 86. 32 ビットカウンタ + ノンス構成



カウンタモードでは、長時間繰り返されない一意なシーケンスを保証するために、カウンタは処理されるブロックごとに初期値からインクリメントされます。32 ビットカウンタであるため、ノンスメッセージは、AES が無効にされたときに格納された初期値に保たれます。128 ビットの初期化ベクトルレジスタの 32 ビットの LSB だけがカウンタを表します。CBC モード（最初のデータブロックを処理するときに一度だけ AES_IVRx レジスタを使用）とは逆に、カウンタモードでは、各データブロックを処理するたびに AES_IVRx レジスタが使用されます。

カウンタモードでは、キー派生 + 復号化モードは適用できません。

注： AES の正常な動作を保証するために、AES_IVRx レジスタは AES が無効なとき（ビット EN = 0）に、書き込まれる必要があります。

AES が有効なときに読み出すと、値 0x00000000 が返されます。

AES が無効なときに読み出すと、最新のカウンタ値が返されます（サスペンドモードの管理に役立ちます）。

CTR モードでは、キー派生 + 復号化は何の目的も果たしません。したがって、AES_CR レジスタの MODE[1:0] = 11 をセットすることは禁じられており、この設定をセットしようと試みると、強制的に MODE[1:0] = 10（CTR モードの復号化に対応）になります。これは、[図 85](#) に示されているように、AES プロセッサの暗号化ブロックを使用してメッセージを復号化します。

CTR モードでのサスペンドモード

CBC モードと同様に、メッセージを中断して、優先順位の高いメッセージを送信してから、中断されたメッセージを再開することが可能です。サスペンドモード機能の詳細については、[図 83](#) および [章 19.5.2](#) を参照してください。

19.6 データ型

データは AES_DINR レジスタに書き込むことによって、32 ビット（ワード）ごとに AES プロセッサに入力されます。AES は 128 ビットのデータブロックを処理します。1 つの 128 ビットデータブロックを処理するには、AES_DINR または AES_DOUTR レジスタを MSB から順に 4 回読み出す、または書き込む必要があります。

システムメモリ構成はリトルエンディアンです。すなわち、使用されるデータ型（ビット、バイト、16 ビットハーフワード、32 ビットワード）に関わらず、最下位データが最下位アドレス位置を占めます。

したがって、暗号化または復号化されるデータの種類のに応じて、AES プロセッサに入力する前に、システムメモリから AES_DINR に書き込まれるデータに対するビット、バイト、またはハーフワードスワッピング操作が必要であり、AES_DOUTR レジスタからシステムメモリに読み出される AES データについても同じスワッピングが必要です。

AES_CR レジスタの DATATYPE ビットは、AES プロセッサに送信する前に AES_DINR レジスタに適用され、プロセッサから送信されるデータに対する AES_DOUTR レジスタに適用されるさまざまなスワップモードを提供します（[図 87](#) を参照）。



注 : スワッピング操作は、AES_DOUTR および AES_DINR レジスタのみを対象とします。AES_KEYRx
 および AES_IVRx レジスタは、選択されたスワップモードの影響を受けません。これらはリトルエン
 ディアン設定に固定です (セクション 19.4 および セクション 19.12 を参照)。

図 87. データ型に応じた 128 ビットブロックの構成

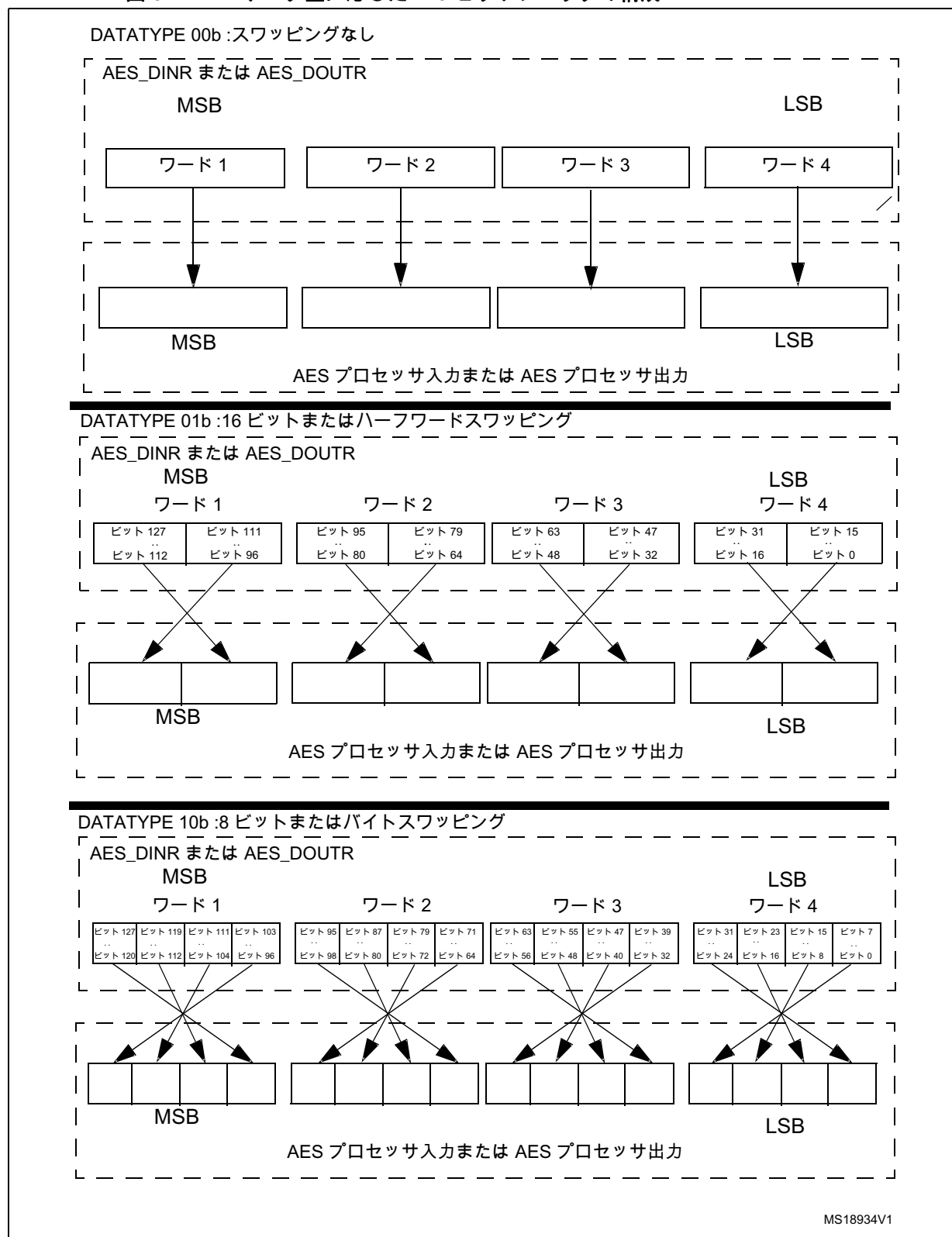
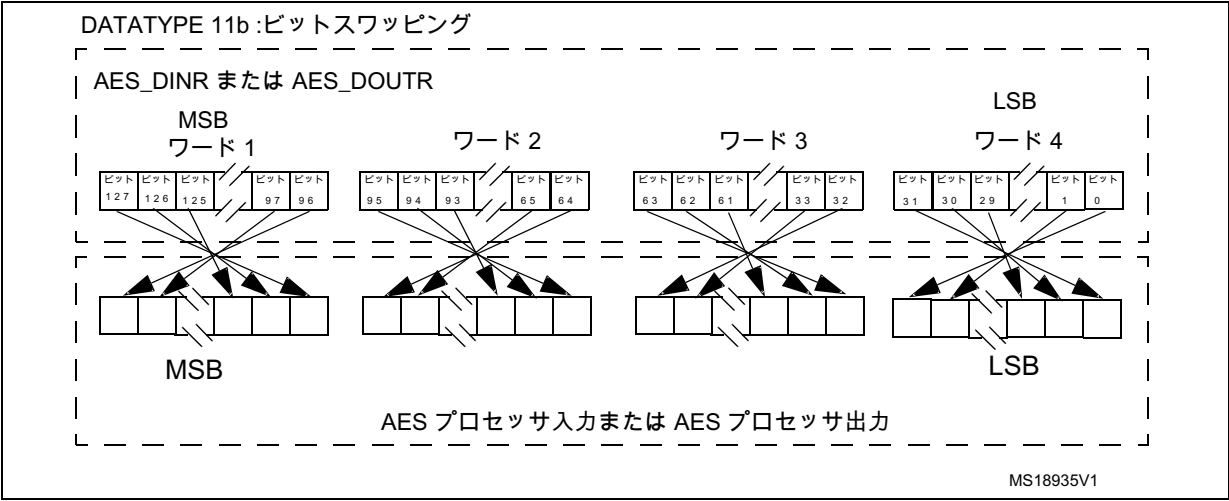


図 88. データ型に応じた 128 ビットブロックの構成 (続き)

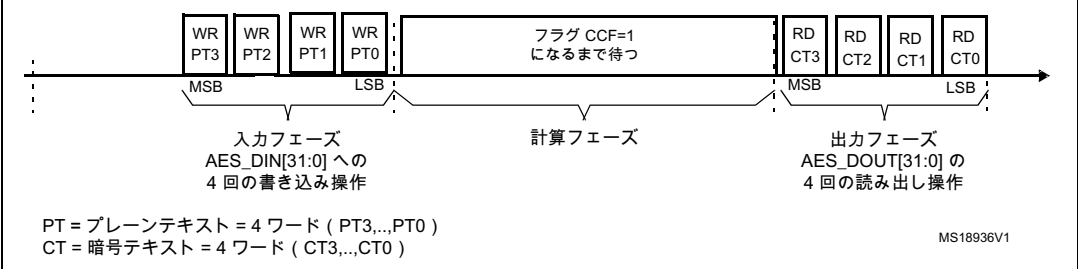


19.7 動作モード

19.7.1 モード 1 : 暗号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0]=00 をプログラムすることによってモード 1 を設定し、CHMOD[1:0] ビットをプログラムすることによって、実行する必要がある連鎖モードのタイプを選択します。
3. CTR または CBC モードが選択された場合は、AES_KEYRx レジスタ (128 ビット暗号化キー) および AES_IVRx レジスタに書き込みます。ECB モードの場合、AES_IVRx レジスタは使用されません。
4. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
5. [図 89 : モード 1 : 暗号化 \(390 ページ\)](#) に示されているように、AES_DINR レジスタに 4 回書き込んで、プレーンテキストを入力します (MSB から順に)。
6. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
7. [図 89 : モード 1 : 暗号化 \(390 ページ\)](#) に示されているように、AES_DOUTR レジスタを 4 回読み出して (MSB から順に)、暗号テキストを取得します。
8. ステップ 5、6、7 を繰り返して、同じ暗号化キーですべてのブロックを処理します。

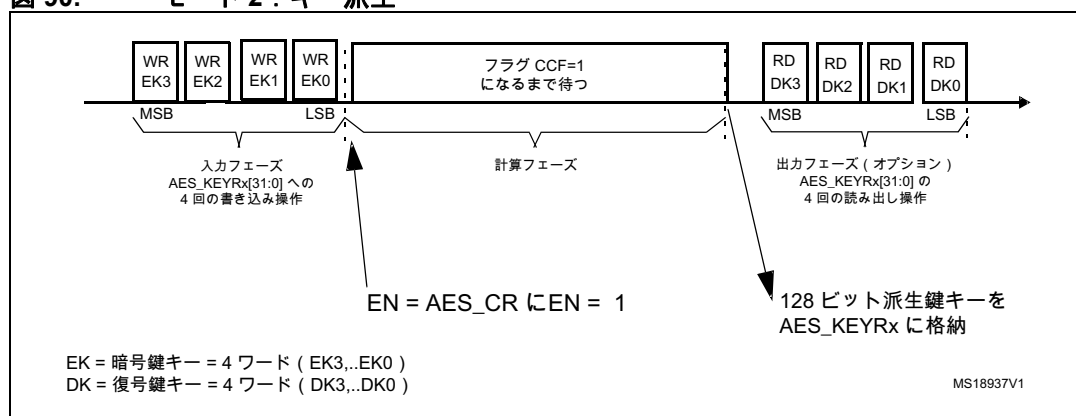
図 89. モード 1 : 暗号化



19.7.2 モード 2 : キー派生

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0]=01 をプログラムすることによって、モード 2 を設定します。このキー派生モードは選択された連鎖アルゴリズムには無関係なので、この場合、CHMOD[1:0] ビットは重要ではありませんのでご注意ください。
3. AES_KEYRx レジスタに暗号化キーを書き込んで、派生キーを取得します。AES_IVRx への書き込みは効果がありません。
4. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
5. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
6. 派生キーは、AES_KEYRx レジスタに自動的に格納されます。必要な場合は、AES_KEYRx レジスタを読み出して、復号化キーを取得します。AES はハードウェアによって無効にされます。派生キーの計算を再開するには、ステップ 3、4、5、および 6 を繰り返します。

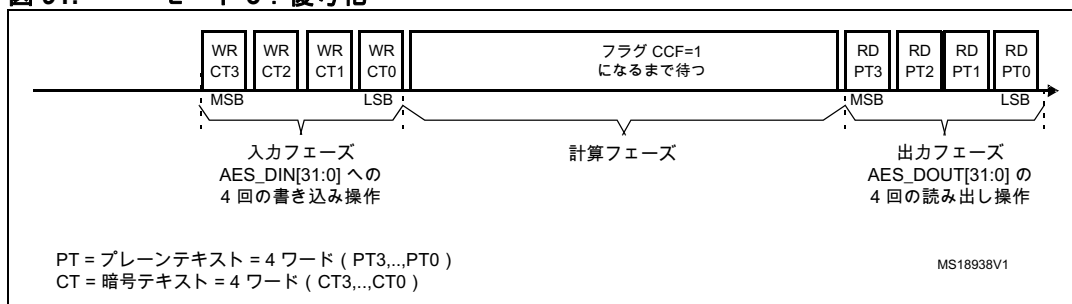
図 90. モード 2 : キー派生



19.7.3 モード 3 : 復号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0]=10 をプログラムすることによってモード 3 を設定し、CHMOD[1:0] ビットをプログラムすることによって、実行する必要がある連鎖モードのタイプを選択します。
3. AES_KEYRx レジスタに復号化キーを書き込みます (モード 2 : キー派生を使用して復号化キーがすでに AES_KEYRx レジスタに格納されている場合、このステップは省略できます)。CTR または CBC モードが選択された場合は、AES_IVRx レジスタに書き込みます。ECB モードの場合、AES_IVRx レジスタは使用されません。
4. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
5. [図 91 : モード 3 : 復号化 \(392 ページ\)](#) に示されているように、AES_DINR レジスタに 4 回書き込んで、暗号テキストを入力します (MSB から順に)。
6. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
7. [図 91 : モード 3 : 復号化 \(392 ページ\)](#) に示されているように、AES_DOUTR レジスタを 4 回読み出して、プレーンテキストを取得します (MSB から順に)。
8. ステップ 5、6、7 を繰り返して、AES_KEYRx レジスタに格納された同じ派生キーを使用して、すべてのブロックを処理します。

図 91. モード 3 : 復号化

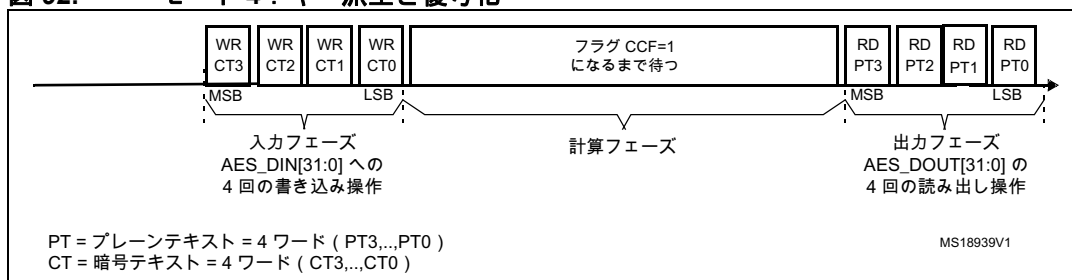


19.7.4 モード 4 : キー派生と復号化

1. AES_CR レジスタの EN ビットをリセットすることによって、AES を無効にします。
2. AES_CR レジスタの MODE[1:0]=11 をプログラムすることによって、モード 4 を設定します。AES が CTR モードに設定されているときには、このモードは禁止されています。ソフトウェアが MODE[1:0] = 11 と CHMOD[1:0] = 10 を書き込んだ場合、強制的に CTR 復号化モードになります。
3. AES_KEYRx レジスタに暗号化キーを書き込みます。CBC モードが選択された場合は、AES_IVRx レジスタに書き込みます。
4. AES_CR レジスタの EN ビットをセットすることによって、AES を有効にします。
5. [図 92 : モード 4 : キー派生と復号化 \(392 ページ\)](#) に示されているように、AES_DINR レジスタに 4 回書き込んで、暗号テキストを入力します (MSB から順に)。
6. AES_SR レジスタの CCF フラグがセットされるまで待ちます。
7. [図 92 : モード 4 : キー派生と復号化 \(392 ページ\)](#) に示されているように、AES_DOUTR レジスタを 4 回読み出して、プレーンテキストを取得します (MSB から順に)。
8. ステップ 5、6、7 を繰り返して、同じ暗号化キーですべてのブロックを処理します。

注 : AES_KEYRx レジスタは処理のすべてのフェーズで暗号化キーを含み、派生キーはこれらのレジスタに格納されません。暗号化キーから開始された派生キーは、AES に内部で格納され、AES_KEYRx レジスタにコピーは格納されません。

図 92. モード 4 : キー派生と復号化



19.8 AES DMA インタフェース

AES アクセラレータは、DMA コントローラに接続するインタフェースを備えています。

DMA は、ワードを転送するように設定する必要があります。

AES は、2 つの異なる DMA リクエストチャネルに関連付けることができます。

- 入力用の DMA リクエストチャネル：AES_CR レジスタの DMAINEN ビットがセットされると、AES は INPUT フェーズ中、AES_DINR レジスタにワードを書き込む必要があるたびに、DMA リクエスト (AES_IN) を開始します。DMA チャネルは、32 ビットデータサイズで memory-to-peripheral モードに設定する必要があります。
- 出力用の DMA リクエストチャネル：DMAOUTEN ビットが有効なとき、AES は OUTPUT フェーズ中、AES_DOUTR レジスタからワードを読み出す必要があるたびに、DMA リクエスト (AES_OUT) を開始します。DMA チャネルは、32 ビットデータサイズで peripheral-to-memory モードに設定する必要があります。

各フェーズで 4 つの DMA リクエストがアサートされます。これらを 図 93 および 図 94 に示します。

DMA リクエストは、AES が無効にされない限り、生成されます。したがって、128 ビットデータブロックの処理終了時のデータ出力フェーズの後、AES は、次のデータブロックがある場合、新しいデータ入力フェーズに自動的に切り替わります。

注： モード 2 (キー派生) の場合、AES_KEYRx レジスタへのアクセスは、ソフトウェアによって CPU を使用して行うことができます。この目的のために提供される DMA チャネルはありません。結果として、AES_CR レジスタの DMAINEN ビットと DMAOUTEN ビットは、このモードでは何の効果もありません。

DMAOUTEN = 1 のとき、CCF フラグは関係せず、この場合、ソフトウェアはこのフラグを読み出す必要はありません。このビットはハイのままになることがあり、アプリケーションが AES を無効にして、DMA 管理をキャンセルし、データ入力またはデータ出力フェーズのために CPU アクセスを使用する必要がある場合、ソフトウェアによってクリアしなければならないことがあります。

図 93. 入力フェーズ中の DMA リクエストとデータ転送 (AES_IN)

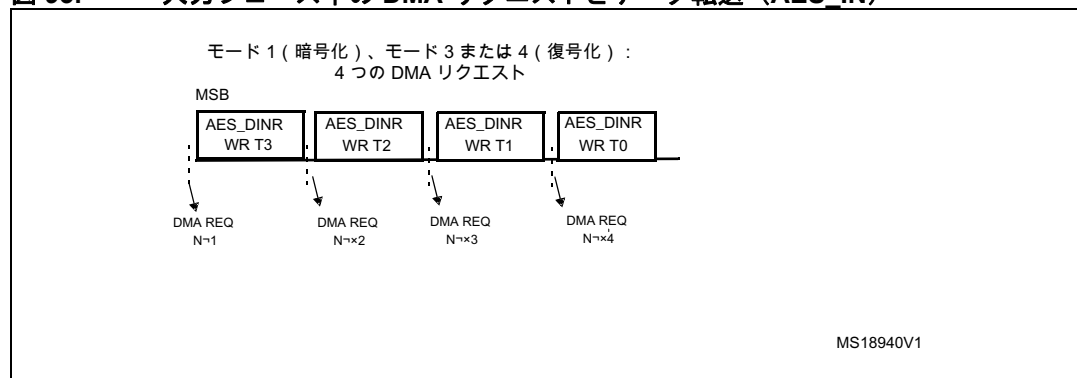
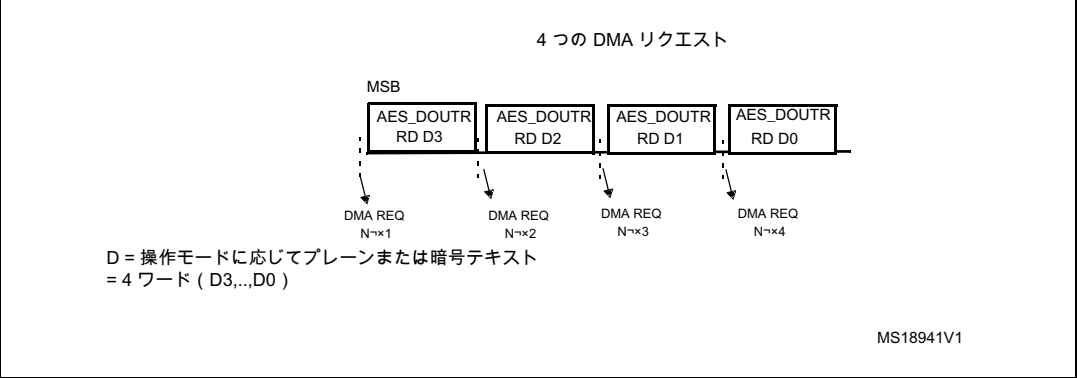


図 94. 出力フェーズ中の DMA リクエスト (AES_OUT)



19.9 エラーフラグ

AES_SR レジスタの RDERR フラグは、計算フェーズまたは入力フェーズ中に予期しない読み出し操作が検出されたときにセットされます。

AES_SR レジスタの WRERR フラグは、出力フェーズまたは計算フェーズ中に予期しない書き込み操作が検出されたときにセットされます。

フラグは、AES_CR レジスタのそれぞれのビットをセットすることによってクリアできます (CCF フラグをクリアするには CCFC ビット、WERR および RDERR フラグをクリアするには ERRC ビット)。

AES_CR レジスタの ERRIE ビットが事前にセットされていた場合、エラーフラグの 1 つがセットされたときに割り込みを生成できます。

エラーが検出された場合、AES はハードウェアによって無効にされず、通常通りに処理を続行します。

19.10 処理時間

次の表に、各動作モードで 128 ビットブロックの処理に必要な時間を示します。

表 75. 処理時間 (クロックサイクル数)

| 動作モード | 入力フェーズ | 計算フェーズ | 出力フェーズ | 合計 |
|--------------------|--------|--------|--------|-----|
| モード 1 : 暗号化 | 8 | 202 | 4 | 214 |
| モード 2 : キー派生 | - | 80 | - | 80 |
| モード 3 : 復号化 | 8 | 202 | 4 | 214 |
| モード 4 : キー派生 + 復号化 | 8 | 276 | 4 | 288 |

19.11 AES 割り込み

表 76. AES 割り込みリクエスト

| 割り込みイベント | イベント フラグ | イネーブル 制御ビット | 待機からの 終了 |
|----------------|-------------|----------------|-------------|
| AES 計算完了フラグ | CCF | CCFIE | あり |
| AES 読み出しエラーフラグ | RDERR | ERRIE | あり |
| AES 書き込みエラーフラグ | WRERR | ERRIE | あり |

19.12 AES レジスタ

19.12.1 AES 制御レジスタ（AES_CR）

アドレスオフセット：0x00

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|------|----|----|----------|---------|-------|-------|------|------|------------|----|-----------|----|---------------|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 予約済み | | | DMAOUTEN | DMAINEN | ERRIE | CCFIE | ERRC | CCFC | CHMOD[1:0] | | MODE[1:0] | | DATATYPE[1:0] | | EN |
| r | r | r | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

- ビット 31:13 予約済み、0 として読み出される
- ビット 12 **DMAOUTEN**:データ出力フェーズの DMA 管理を有効にします。
- 0 : DMA（データ出力フェーズ時）無効
- 1 : DMA（データ出力フェーズ時）有効
- DMAOUTEN ビットがセットされた場合、モード 1、3、または 4 では、出力データフェーズの DMA リクエストが生成されます。モード 2（キー派生）では、このビットは効果がありません。
- ビット 11 **DMAINEN** : データ入力フェーズの DMA 管理を有効にします。
- 0 : DMA（データ入力フェーズ時）無効
- 1 : DMA（データ入力フェーズ時）有効
- DMAINEN ビットがセットされた場合、モード 1、3、または 4 では、データ入力フェーズの DMA リクエストが生成されます。モード 2（キー派生）では、このビットは効果がありません。
- ビット 10 **ERRIE** : エラー割り込み有効
- RDERR フラグまたは WRERRフラグ がセットされた場合、割り込みが生成されます。
- 0 : エラー割り込みは無効です。
- 1 : エラー割り込みは有効です。
- ビット 9 **CCFIE** : CCF フラグ割り込み有効
- CCF フラグがセットされている場合、割り込みが生成されます。
- 0 : CCF 割り込み無効
- 1 : CCF 割り込み有効
- ビット 8 **ERRC** : エラークリア
- このビットに 1 を書き込むと、RDERR および WRERR フラグがクリアされます。
- このビットは常にローとして読み出されます。
- ビット 7 **CCFC**: : 計算完了フラグクリア
- このビットに 1 を書き込むと、CCF フラグがクリアされます。
- このビットは常にローとして読み出されます。
- ビット 6:5 **CHMOD[1:0]** : AES連鎖モード
- 00 : ECB（電子コードブック）
- 01 : CBC（暗号ブロック連鎖）
- 10 : CTR（カウンタモード）
- 11 : 予約済み。
- AES 連鎖モードは、AES が無効のときに変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

ビット 4:3 **MODE[1:0]**: AES 動作モード

- 00: モード 1: 暗号化
- 01: モード 2: キー派生
- 10: モード 3: 復号化
- 11: モード 4: キー派生 + 復号化

動作モードは、AES が無効のときに変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

CTR モードが選択された場合、モード 4 は禁じられています。それにもかかわらず、ソフトウェアが CTR モード設定でモード 4 を設定しようと試みた場合は、強制的にモード 3 に設定されます。

ビット 2:1 **DATATYPE[1:0]**: データ型選択 (暗号ブロックに入出力されるデータについて)

- 00: 32 ビットデータ。スワッピングなし。
- 01: 16 ビットデータ、すなわちハーフワード。ワード内の各ハーフワードがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0x56AB7643 です。
- 10: 8 ビットデータ、すなわちバイト。ワード内のすべてのバイトがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0xAB564376 です。
- 11: ビットデータ。ワード内のすべてのビットがスワップされます。たとえば、AES_DINR レジスタに書き込まれた 4 つの 32 ビットデータの 1 つが 0x764356AB であった場合、暗号ブロックに与えられる値は 0xD56AC26E です。

データ型選択は、AES が無効の場合に変更する必要があります。予測不能な AES の動作を避けるために、AES が有効なときにこれらのビットに書き込むことは禁じられています。

ビット 0 **EN**: AES 有効

- 0: AES 無効
- 1: AES 有効

AES は、このビットをリセットすることによって、いつでも再初期化できます。その場合、AES は EN がセットされると、新しいブロックの処理を開始する準備が整います。

このビットは、モード 2 (キー派生) では、AES 計算の終了時にハードウェアによってクリアされます。

19.12.2 AES ステータスレジスタ (AES_SR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|----|----|----|----|----|----|----|----|----|----|----|----|-------|-------|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 予約済み | | | | | | | | | | | | | WRERR | RDERR | CCF |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 31:3 予約済み、0 として読み出される

ビット 2 **WRERR** : 書き込みエラーフラグ

このビットは、(計算またはデータ出力フェーズで) AES_DINR レジスタへの予期しない書き込み操作が検出されたときに、ハードウェアによってセットされます。AES_CR レジスタの ERRIE ビットがセットされていた場合、割り込みが生成されます。このフラグは AES には影響を与えず、WERR がセットされている場合でも AES は実行を続行します。

ソフトウェアでAES_CR レジスタの ERRC ビットをセットすることによって、クリアされます。

0 : 書き込みエラーは検出されていません。

1 : 書き込みエラーが検出されました。

ビット 1 **RDERR** : 読み出しエラーフラグ

このビットは、(計算またはデータ入力フェーズで) AES_DOUTR レジスタからの予期しない読み出し操作が検出されたときに、ハードウェアによってセットされます。AES_CR レジスタの ERRIE ビットが事前にセットされていた場合は、割り込みが生成されます。このフラグは AES には影響を与えず、RDERR がセットされている場合でも AES は実行を続行します。

ソフトウェアでAES_CR レジスタの ERRC ビットをセットすることによって、クリアされます。

0 : 読み出しエラーは検出されていません。

1 : 読み出しエラーが検出されました。

ビット 0 **CCF** : 計算完了フラグ

このビットは、計算が完了したときに、ハードウェアによってセットされます。AES_CR レジスタの CCFIE ビットが事前にセットされている場合、割り込みが生成されます。

ソフトウェアでAES_CR レジスタの CCFC ビットをセットすることによって、クリアされます。

0 : 計算が完了しました。

1 : 計算は完了していません。

注 : このビットは、DMAOUTEN = 0 のときだけ意味を持ちます。DMA_EN = 1 のときはハイを維持します。



19.12.3 AES データ入力レジスタ (AES_DINR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

ビット 31:0 **DINR[31:0]** : データ入力レジスタ。

このレジスタは、入力フェーズで4 回書き込む必要があります。

- モード 1 (暗号化) では、プレーンテキストを表す 4 ワードを MSB から LSB の順に書き込む必要があります。
- モード 2 (キー派生) では、AES_KEYRx レジスタから開始される派生キー計算のみが行われるため、このレジスタは使用されません。
- モード 3 (復号化) および 4 (キー派生 + 復号化) では、暗号テキストを表す 4 ワードを MSB から LSB の順に書き込む必要があります。

注 : このレジスタには 32 ビットデータ幅でアクセスする必要があります。

19.12.4 AES データ出力レジスタ (AES_DOUTR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

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

ビット 31:0 **DOUTR[31:0]** : データ出力レジスタ

このレジスタは読み出し専用です。

CCF フラグ (計算完了フラグ) がセットされた後、このデータレジスタを 4 回読み出すことによって、128 ビットの出力結果にアクセスできます。

- モード 1 (暗号化) では、読み出された 4 ワードは、暗号テキストを MSB から LSB の順に表します。
- モード 2 (キー派生) では、派生キーは AES_KEYRx レジスタにあるので、このレジスタを読み出す必要はありません。
- モード 3 (復号化) およびモード 4 (キー派生 + 復号化) では、読み出された 4 ワードは、プレーンテキストを MSB から LSB の順に表します。

注 : このレジスタには 32 ビットデータ幅でアクセスする必要があります。

19.12.5 AES キーレジスタ 0 (AES_KEYR0) (LSB:キー [31:0])

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| KEYR0[31:16] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| KEYR0[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **KEYR0[31:0]** : データ出力レジスタ (LSB キー [31:0])

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。
モード 1 (暗号化)、モード 2 (キー派生)、およびモード 4 (キー派生 + 復号化) では、書き込まれる値は暗号化キーを LSB から表し、キー [31:0] を意味します。
モード 3 (復号化) では、書き込まれる値は復号化キーを LSB から表し、キー [31:0] を意味します。この復号化モードで暗号化キーがレジスタに書き込まれると、AES が有効になる前に読み出すと、暗号化の値が返されます。CCF フラッグがセットされた後で読み出すと、復号化キーが返されます。
AES が有効なときにこのレジスタを読み出すと、予期しない値が返されます。

注 : モード 4 (キー派生 + 復号化) では、このレジスタは、派生キーを含みません。常に暗号化キー値を含みます。

19.12.6 AES キーレジスタ 1 (AES_KEYR1) (キー [63:32])

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

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

ビット 31:0 **KEYR1[31:0]** : AES キーレジスタ (キー [63:32])

AES_KEYR0 の説明を参照してください。

19.12.7 AES キーレジスタ 2 (AES_KEYR2) (キー [95:64])

アドレスオフセット : 0x18

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| KEYR2[31:16] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| KEYR2[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **KEYR2[31:0]** : AES キーレジスタ (キー [95:64])
AES_KEYR0 の説明を参照してください。

19.12.8 AES キーレジスタ 3 (AES_KEYR3) (MSB : キー [127:96])

アドレスオフセット : 0x1C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|--------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| KEYR3[31:16] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| KEYR3[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **KEYR3[31:0]** : AES キーレジスタ (MSB キー [127:96])
AES_KEYR0 の説明を参照してください。

19.12.9 AES 初期化ベクトルレジスタ 0 (AES_IVR0) (LSB : IVR[31:0])

アドレスオフセット : 0x20

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| IVR0[31:16] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IVR0[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **IVR0[31:0]** : 初期化ベクトルレジスタ (LSB IVR [31:0])

- このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。
- 次の場合、このレジスタの値は意味を持ちません。
- ECB (電子コードブック) モードが選択された場合。
 - キー派生に加えて、CTR または CBC モードが選択された場合。
- CTR モード (カウンタモード) では、このレジスタは 32 ビットのカウンタ値を含みます。
- AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

19.12.10 AES 初期化ベクトルレジスタ 1 (AES_IVR1) (IVR[63:32])

アドレスオフセット : 0x24

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| IVR1[31:16] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IVR1[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:0 **IVR1[31:0]** : 初期化ベクトルレジスタ (IVR [63:32])

- このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。
- 次の場合、このレジスタの値は意味を持ちません。
- ECB (電子コードブック) モードが選択された場合。
 - キー派生、またはキー派生 + 復号化モードに加えて CTR または CBC モードが選択された場合。
- CTR モード (カウンタモード) では、このレジスタはノンス値を含みます。
- AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。



19.12.11 AES 初期化ベクトルレジスタ 2 (AES_IVR2) (IVR[95:64])

アドレスオフセット : 0x28

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| IVR2[31:16] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IVR2[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:0 **IVR2[31:0]** : 初期化ベクトルレジスタ (IVR [95:64])

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

- ECB (電子コードブック) モードが選択された場合。
 - キー派生に加えて CTR または CBC モードが選択された場合、またはキー派生 + 復号化モードの場合。
- CTR モード (カウンタモード) では、このレジスタはノンス値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。

19.12.12 AES 初期化ベクトルレジスタ 3 (AES_IVR3) (MSB : IVR[127:96])

アドレスオフセット : 0x2C

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| IVR3[31:16] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| IVR3[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:0 **IVR3[31:0]** : 初期化ベクトルレジスタ (MSB IVR [127:96])

このレジスタは、AES_CR レジスタの EN ビットがセットされる前に書き込む必要があります。

次の場合、このレジスタの値は意味を持ちません。

- ECB (電子コードブック) モードが選択された場合。
 - キー派生に加えて CTR または CBC モードが選択された場合、またはキー派生 + 復号化モードの場合。
- CTR モード (カウンタモード) では、このレジスタはノンス値を含みます。

AES が有効なときにこのレジスタ読み出すと、値 0x00000000 が返されます。



表 77. AES レジスタマップ

表 77. AES レジスタマップ

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や損害などににつきましては如何なる責任にも負いません。

並行して、アナログシードと専用の PLL48CLK クロックが監視されます。シードで異常なシーケンスが発生したり、PLL48CLK クロックの周波数が低すぎる場合、ステータスビット (RNG_SR レジスタ内) がそれを示します。エラーが検出された場合、割り込みを生成することができます。

20.3.1 動作

RNG を作動させるには、次のステップに従います。

1. 必要に応じて割り込みを有効にします (RNG_CR レジスタの IE ビットをセットします)。乱数の準備ができたとき、またはエラーが発生したとき、割り込みが生成されます。
2. RNG_CR レジスタの RNGEN ビットをセットして、乱数発生を有効にします。これによってアナログ部分、RNG_LFSR およびエラー検出回路が作動します。
3. 割り込みのたびに、エラーが発生していないこと (RNG_SR レジスタの SEIS ビットおよび CEIS ビットが“0”)、乱数の準備ができていないこと (RNG_SR レジスタの DRDY ビットが“1”)を確認します。RNG_DR レジスタの内容が読み出せるようになります。

FIPS PUB 140-2 の要件に従い、RNGEN ビット設定後最初に生成された乱数は使用すべきではありませんが、次に生成される乱数との比較のため保存されます。後に生成される各乱数は、それぞれ前に生成された値と比較する必要があります。比較された 2 つの値が同一の場合、試験は不合格となります (連続乱数発生器試験)。

20.3.2 エラー管理

CEIS ビットが“1”として読み出された場合 (クロックエラー)

クロックエラーの場合、PLL48CLK クロックが正しくないため、RNG は以降、乱数を発生することはできません。RNG クロックが供給されるようにクロックコントローラが正しく設定されていることを確認し、CEIS ビットをクリアします。CEIS ビットが“0”になると、RNG が動作できるようになります。クロックエラーはその前に発生した乱数には影響しないため、RNG_DR レジスタの内容は使用できます。

SEIS ビットが“1”として読み出された場合 (シードエラー)

シードエラーの場合、SEIS ビットが“1”の間は乱数発生が中断します。RNG_DR レジスタにおける値が利用可能な場合であっても、エントロピーが十分でない可能性があるため使用することはできません。

ここでは、SEIS ビットをクリアした後、RNGEN ビットをクリアおよびセットし、RNG を再初期化および再起動します。

20.4 RNG レジスタ

RNG は、制御レジスタ、データレジスタ、ステータスレジスタと連動します。これらのレジスタは、ワード (32 ビット) 単位でアクセスする必要があります。

20.4.1 RNG 制御レジスタ (RNG_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|----|----|----|----|----|----|----|----|----|----|----|----|-------|------|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 予約済み | | | | | | | | | | | | IE | RNGEN | 予約済み | |
| | | | | | | | | | | | | rw | rw | | |

ビット 31:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **IE** : 割り込みイネーブル

- 0 : RNG 割り込みは無効です。
- 1 : RNG 割り込みは有効です。RNG_SR レジスタで DRDY = 1、SEIS = 1 または CEIS = 1 になると、割り込みは直ちに保留となります。

ビット 2 **RNGEN** : 乱数発生器イネーブル

- 0 : 乱数発生器は無効です。
- 1 : 乱数発生器は有効です。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

20.4.2 RNG ステータスレジスタ (RNG_SR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

| | | | | | | | | | | | | | | | |
|------|----|----|----|----|----|----|-------|-------|------|----|----|----|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| 予約済み | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 予約済み | | | | | | | SEIS | CEIS | 予約済み | | | | SECS | CECS | DRDY |
| | | | | | | | rc_w0 | rc_w0 | | | | | r | r | r |

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 6 **SEIS** : シードエラー割り込みステータス

このビットは SECS と同時にセットされ、"0"を書き込むとクリアされます。

- 0 : 障害シーケンスは検出されませんでした。
- 1 : 次の障害シーケンスのうち 1 つが検出されました。
 - 64 個以上のビットが連続して同じ値 (0 または 1)
 - 0 と 1 の繰り返し (0101010101...01) が 32 回以上連続

RNG_CR レジスタで IE = 1 である場合、割り込みが保留となります。

ビット 5 **CEIS** : クロックエラー割り込みステータス

このビットは CECS と同時にセットされ、"0"を書き込むとクリアされます。

- 0 : PLL48CLK クロックが正しく検出されました。
 - 1 : PLL48CLK が正しく検出されませんでした ($f_{PLL48CLK} < f_{HCLK}/16$)。
- RNG_CR レジスタで IE = 1 である場合、割り込みが保留となります。

ビット 4:3 予約済みであり、リセット値に保持する必要があります。

- ビット 2 **SECS** : シードエラーの現在のステータス
- 0 : 障害のあるシーケンスは現在のところ検出されていません。SEIS ビットがセットされている場合、これは障害のあるシーケンスが検出されたが、回復していることを意味します。
 - 1 : 次の障害シーケンスのうち 1 つが検出されました。
 - 64 個以上のビットが連続して同じ値 (0 または 1)
 - 0 と 1 の繰り返し (0101010101...01) が 32 回以上連続
- ビット 1 **CECS** : クロックエラーの現在のステータス
- 0 : PLL48CLK クロックが正しく検出されました。CEIS ビットがセットされている場合、これはクロックエラーが検出されたが、回復していることを意味します。
 - 1 : PLL48CLK が正しく検出されませんでした ($f_{PLL48CLK} < f_{HCLK}/16$)。
- ビット 0 **DRDY** : データレディ
- 0 : RNG_DR レジスタがまだ有効でなく、乱数データは利用できません。
 - 1 : RNG_DR レジスタに有効な乱数データが入っています。
- 注 :** **RNG_CR レジスタで IE = 1 である場合、割り込みが保留となります。**
RNG_DR レジスタがいったん読み出されると、新しい有効な値が計算されるまでこのビットは 0 に戻ります。

20.4.3 RNG データレジスタ（RNG_DR）

アドレスオフセット : 0x08
リセット値 : 0x0000 0000

RNG_DR レジスタは読み出し専用レジスタであり、読み出されると 32 ビットの乱数値を返します。いったん読み出されると、最長で PLL48CLK クロック 40 周期後に、このレジスタは新しい乱数値を出力します。ソフトウェアは、RNDATA 値を読み出す前に DRDY ビットがセットされていることを確認する必要があります。

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

- ビット 31: **RNDATA** : 乱数データ
- 32 ビットの乱数データです。

20.4.4 RNG レジスタマップ

表 78 に、RNG レジスタマップとリセット値を示します。

表 78. RNG レジスタマップとリセット値

| オフ セット | レジスタ名 リセット値 | レジスタサイズ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|----------------------|--------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|----|-------|------|------|
| | | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| 0x00 | RNG_CR 0x00000000 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IE | RNGEN | Res. | Res. |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | |
| 0x04 | RNG_SR 0x00000000 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SEIS | CEIS | Res. | Res. | Res. | 0 | SECS | Res. | Res. |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | | 0 | | | |
| 0x08 | RNG_DR 0x00000000 | RNDATA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

21 汎用タイマ (TIM2)

21.1 TIM2 の概要

汎用タイマは、プログラム可能なプリスケラによって駆動される 16 ビットの自動再ロードカウンタで構成されています。

入力信号のパルス長の測定（入力キャプチャ）や出力波形の生成（出力比較と PWM）など、さまざまな目的に使用できます。

パルス幅と波形の周期は、タイマプリスケラと RCC クロックコントローラプリスケラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

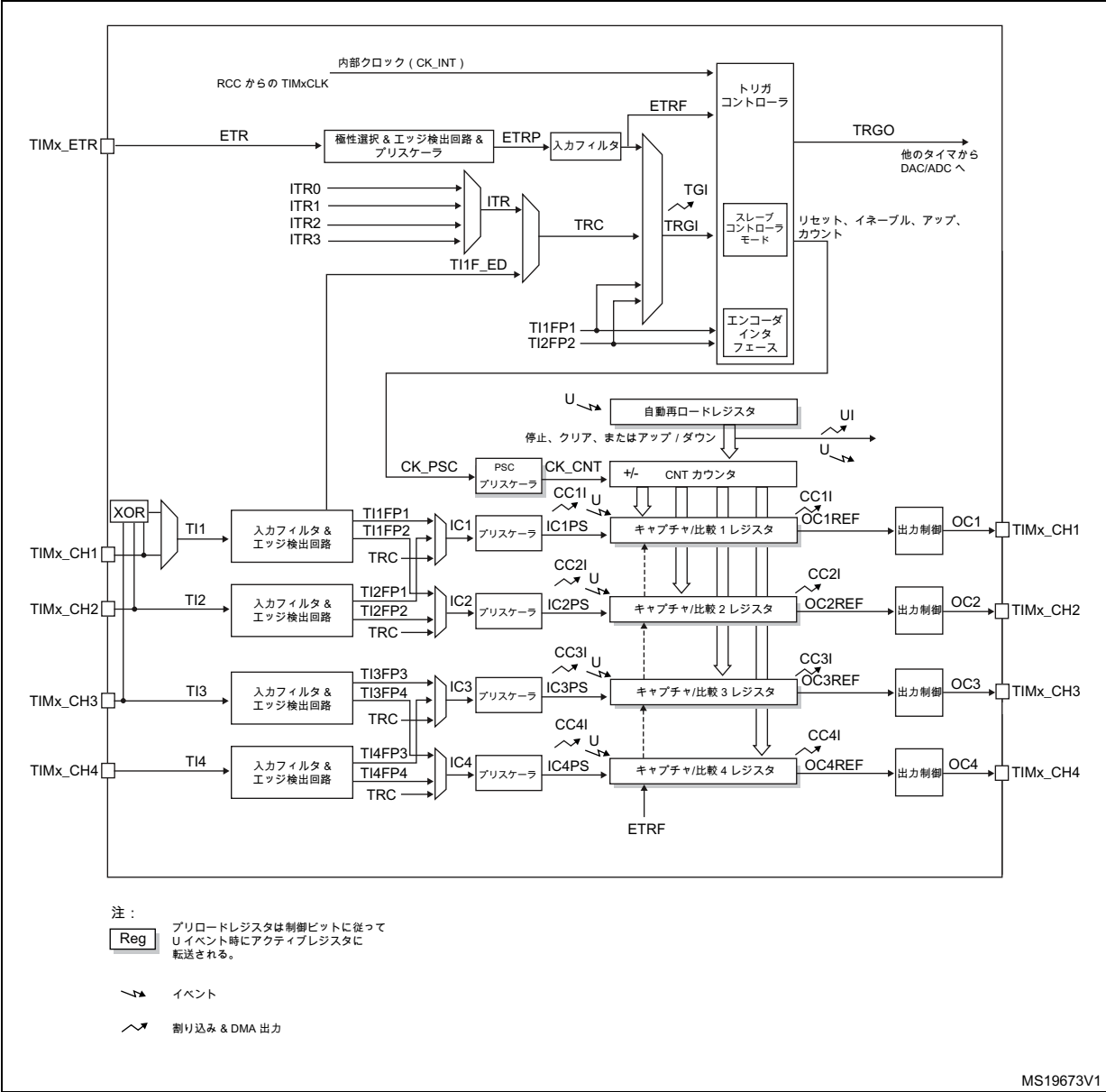
タイマは完全に独立していて、いかなるリソースも共有しません。[セクション 21.3.15](#) で説明するように、同期できます。

21.2 TIM2 の主な機能

汎用 TIMx タイマの主な機能は、次のとおりです。

- 16 ビット (TIM2) のアップカウンタ、ダウンカウンタ、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を、1 から 65535 の間で分周する 16 ビットプログラム可能プリスケラ。
- 次の機能を持つ、最大 4 つの独立チャネル。
 - 入力キャプチャ
 - 出力比較
 - PWM 生成（エッジアラインモードとセンターアラインモード）
 - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割り込み/DMA 生成。
 - 更新：カウンタオーバーフロー/アンダーフロー、カウンタの初期化（ソフトウェアまたは内部/外部トリガによる）
 - トリガイベント（カウンタの開始、停止、初期化、または内部/外部トリガによるカウント）
 - 入力キャプチャ
 - 出力比較
- 位置決め目的のインクリメンタル（直交）エンコーダとホールセンサ回路をサポート
- 外部クロックまたはサイクルごとの電流管理のためのトリガ入力

図 96. 汎用タイマのブロック図



21.3 TIM2 の機能詳細

21.3.1 タイムベースユニット

プログラム可能タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットです。カウンタは、カウントアップ、カウントダウン、またはカウントアップとカウントダウンの両方を行います。カウンタのクロックは、プリスケアラによって分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケアラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。



タイムベースユニットには、次のレジスタで構成されます。

- カウンタレジスタ (TIMx_CNT)
- プリスケールレジスタ (TIMx_PSC) :
- 自動再ロードレジスタ (TIMx_ARR)

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きは、プリロードレジスタへのアクセスになります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット (ARPE) に応じて、常時または更新イベント (UEV) ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 の場合、カウンタがオーバーフロー（またはダウンカウント時はアンダーフロー）に達したときに、更新イベントが送られます。また、ソフトウェアで生成することもできます。更新イベントの生成については、各設定の詳細が説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット (CEN) がセットされているときにのみ、プリスケール出力 CK_CNT から供給されます（カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください）。

実際のカウンタイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケールの説明

プリスケールは、カウンタクロック周波数を 1 から 65536 の間の値で分周することができます。16 ビット/32 ビットレジスタ (TIMx_PSC レジスタ) を使って制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケール比は、次の更新イベントで有効になります。

図 97 と 21.3.2 に、プリスケール比を動作中に変更したときのカウンタの動作の例を示します。

図 97. プリスケール分周比が 1 から 2 に変化したときのカウンタのタイミング図

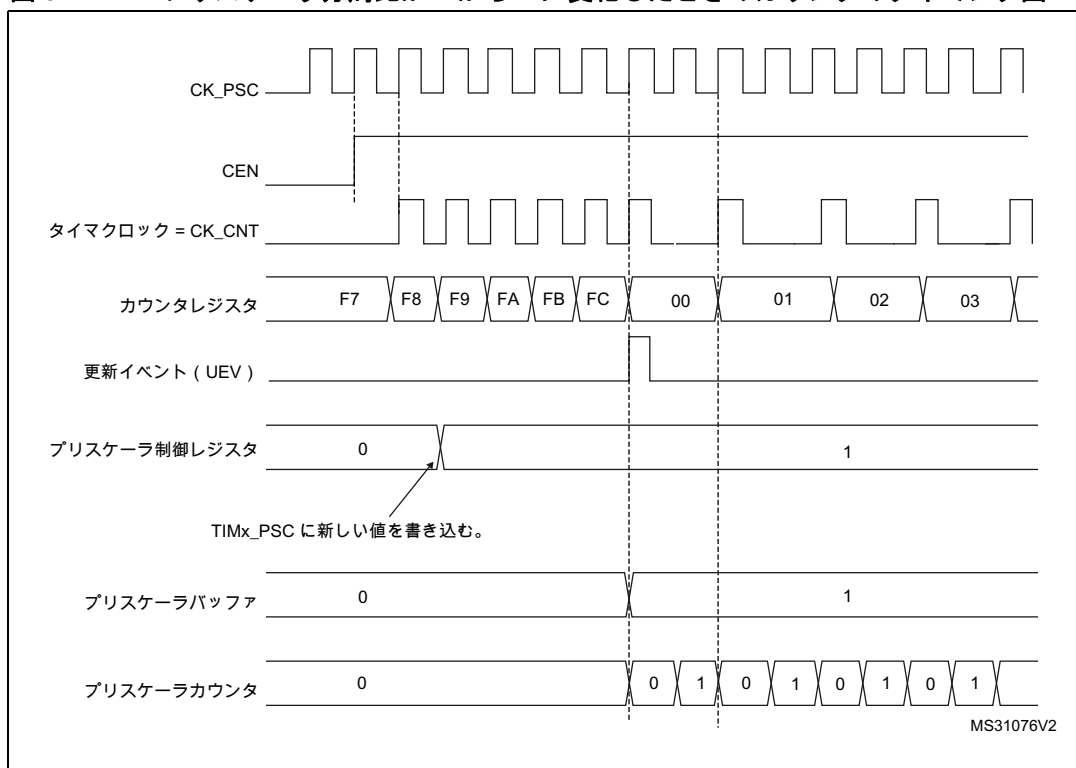
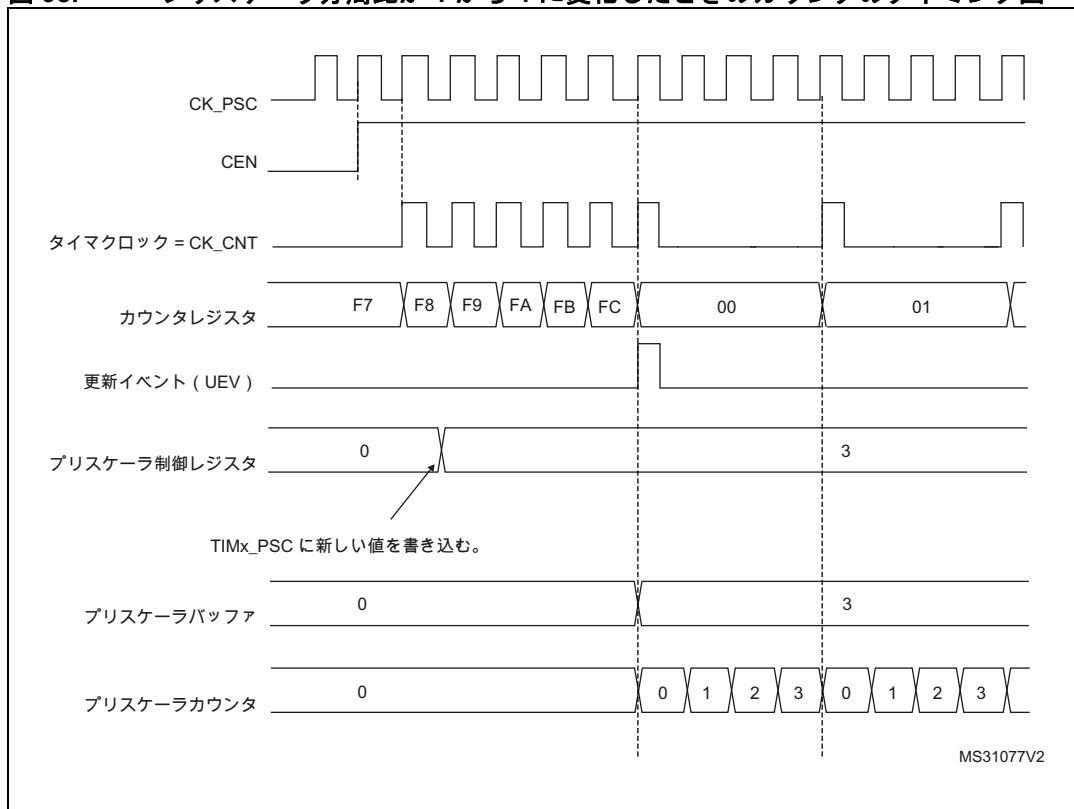


図 98. プリスケアラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



21.3.2 カウンタモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値 (TIMx_ARR レジスタの内容) までカウントし、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、(ソフトウェアで、または、スレーブモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることで生成できます。

UEV イベントは、TIMx_CR1 レジスタの UDIS ビットをセットすることにより、ソフトウェアで無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケアラのカウンタと同じく、カウンタは 0 からカウントをリスタートします (ただし、プリスケアラ比は変化しません)。さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのバッファにはプリロード値 (TIMx_PSC レジスタの内容) が再びロードされます。
- 自動再ロードシャドウレジスタは、プリロード値 (TIMx_ARR) で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 99. 内部クロック分周比が 1 の場合のカウンタのタイミング図

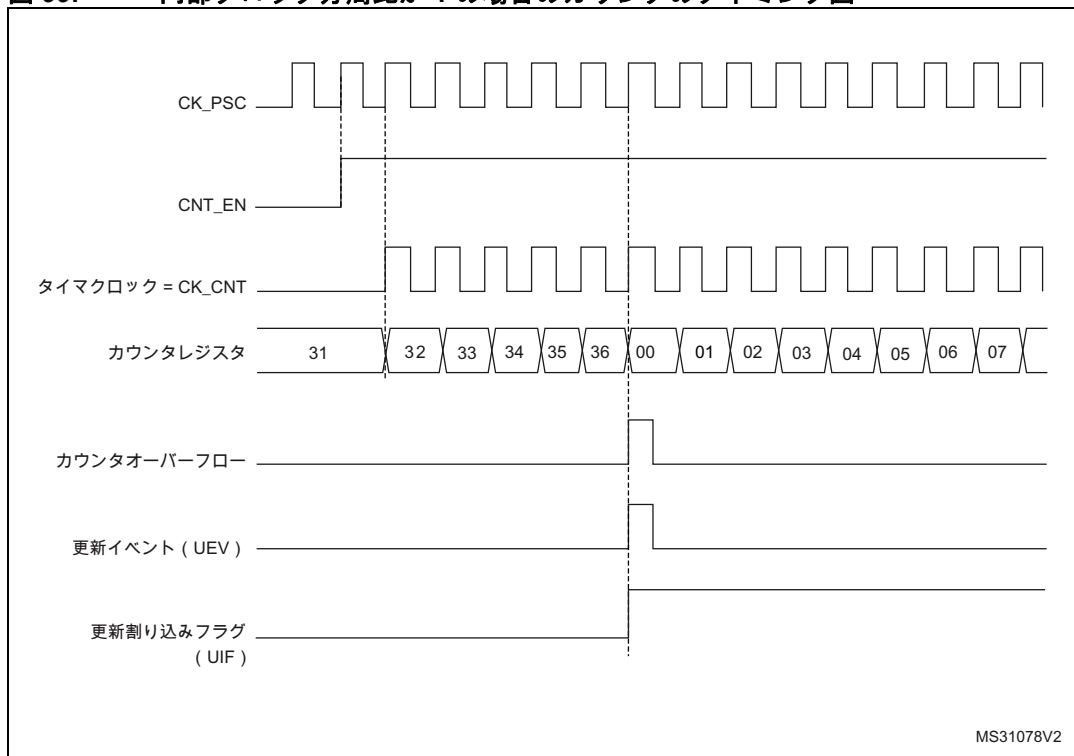


図 100. 内部クロック分周比が 2 の場合のカウンタのタイミング図

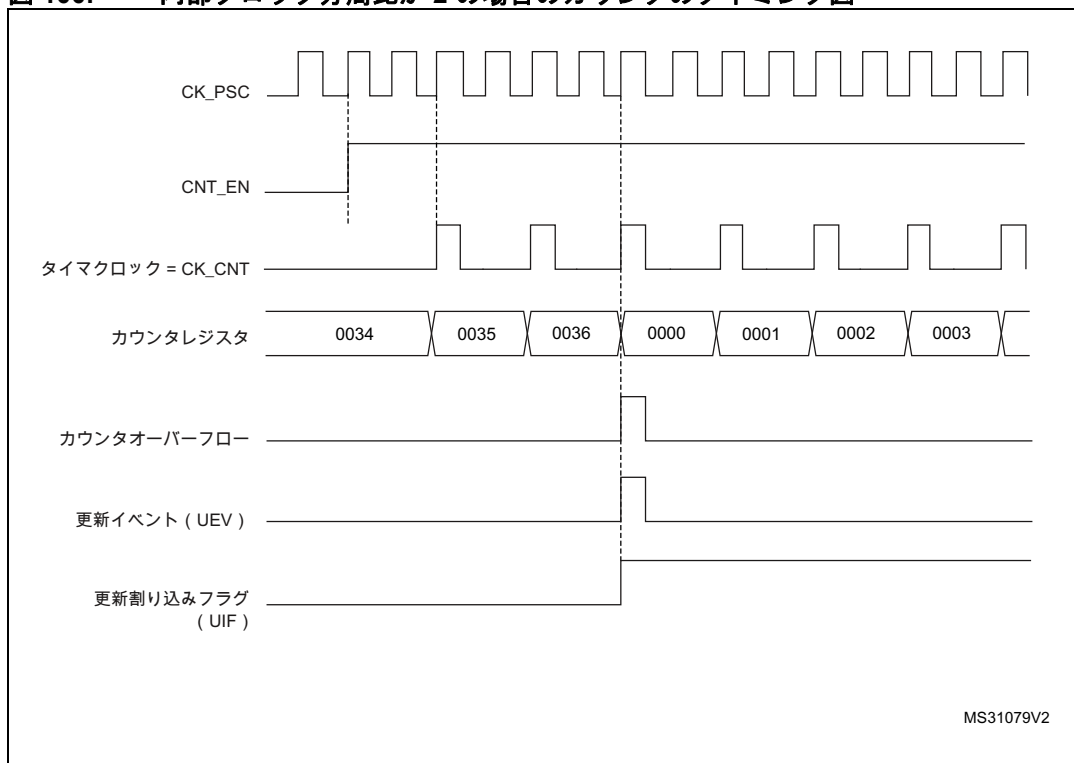


図 101. 内部クロック分周比が 4 の場合のカウンタのタイミング図

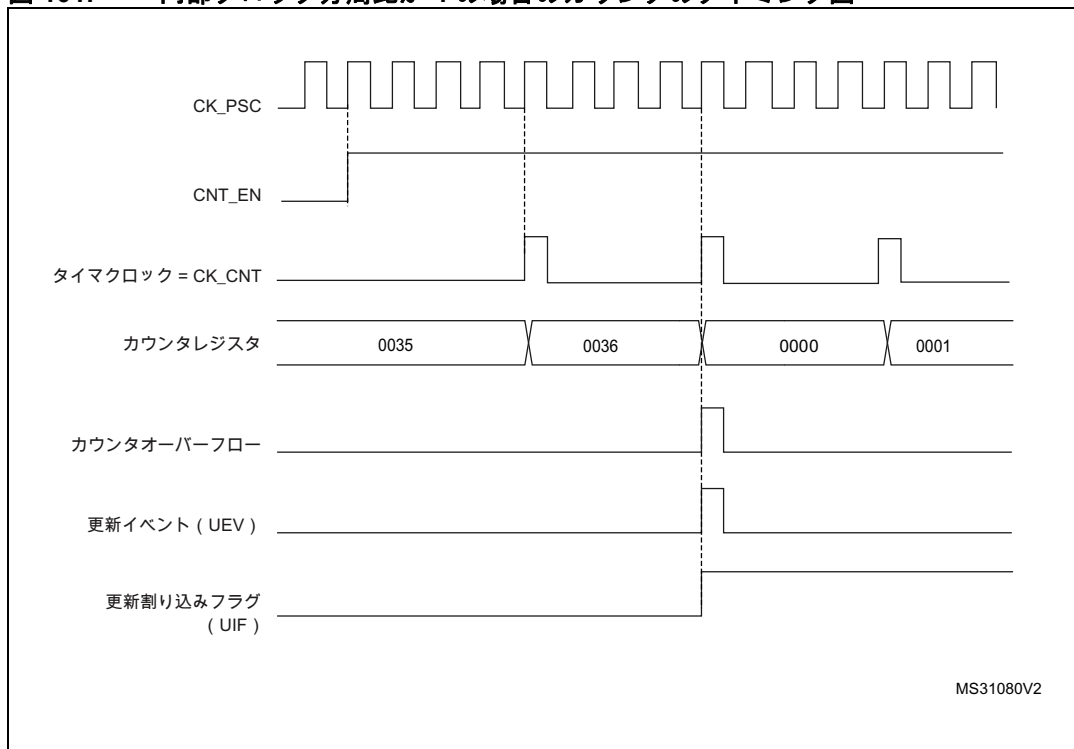


図 102. 内部クロック分周比が N の場合のカウンタのタイミング図

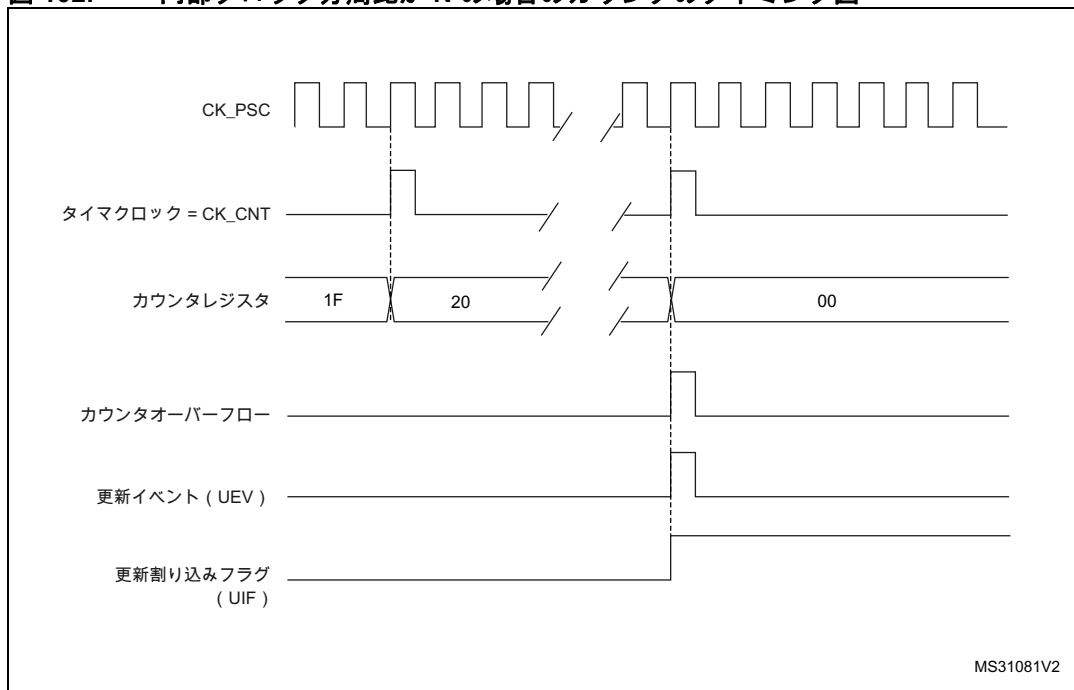


図 103. ARPE=0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタのタイミング図

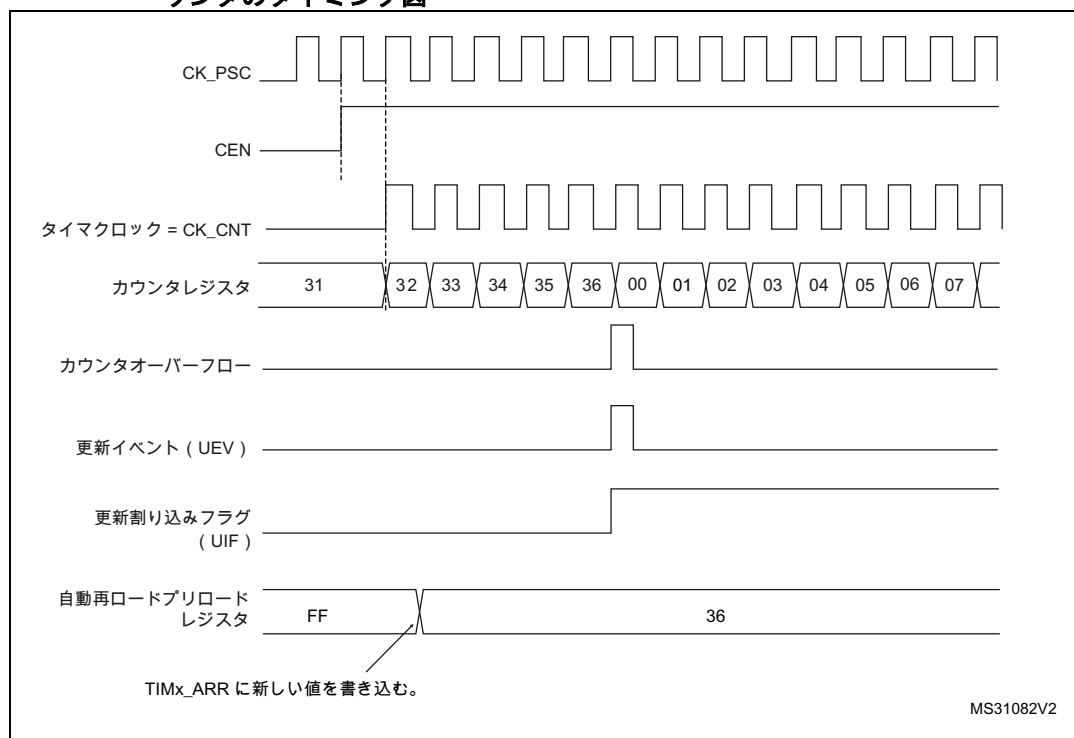
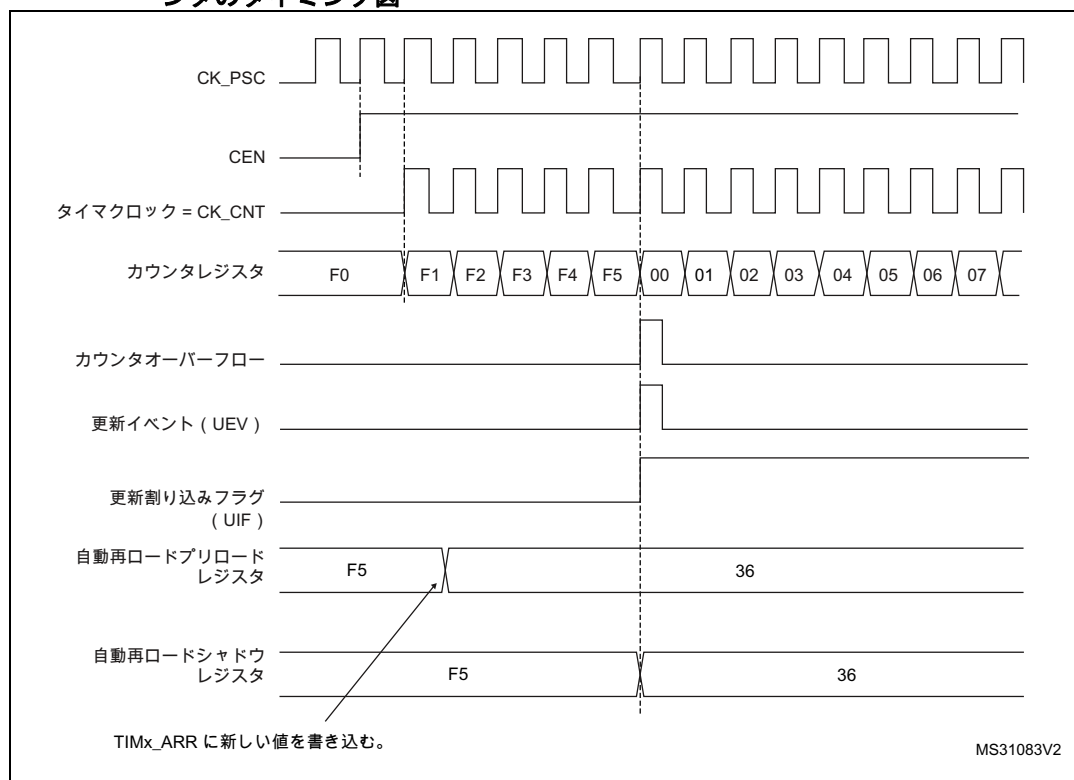


図 104. ARPE=1 (TIMx_ARR はプリロードされる) のときの更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値 (TIMx_ARR レジスタの内容) から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、(ソフトウェアで、または、スレーブモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケアラのカウンタは 0 からリスタートします (しかし、プリスケアラ比は変化しません)。

さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのバッファにはプリロード値 (TIMx_PSC レジスタの内容) が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値 (TIMx_ARR レジスタの内容) で更新されます。カウンタがリロードされる前に自動再ロードが更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 105. 内部クロック分周比が 1 の場合のカウンタのタイミング図

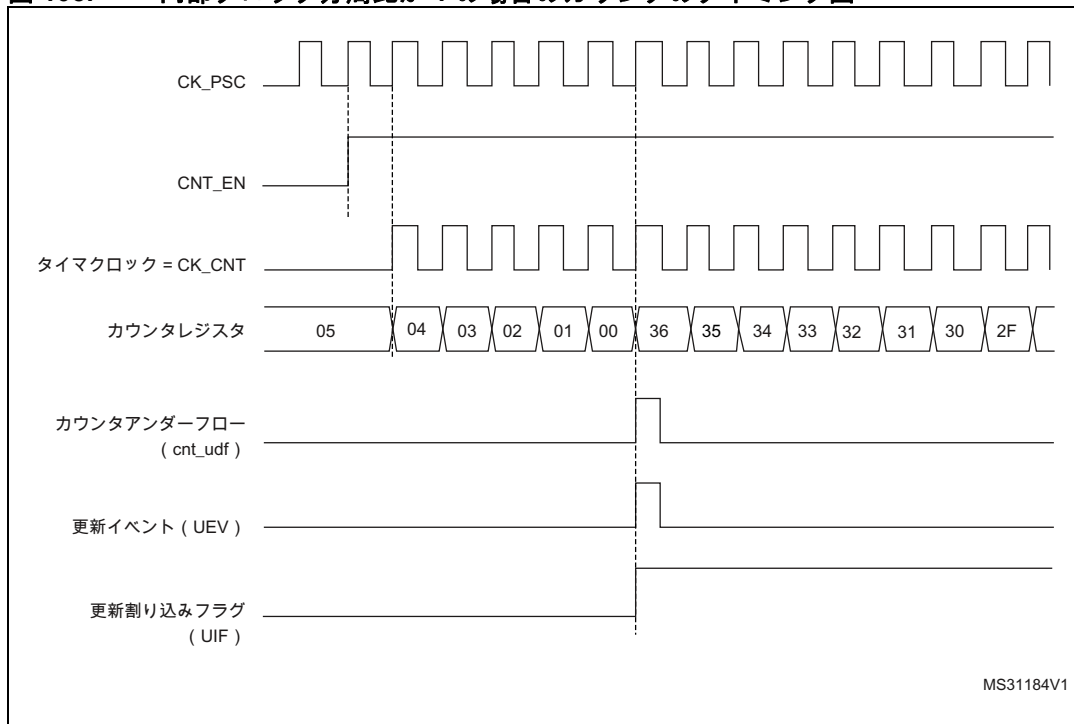


図 106. 内部クロック分周比が 2 の場合のカウンタのタイミング図

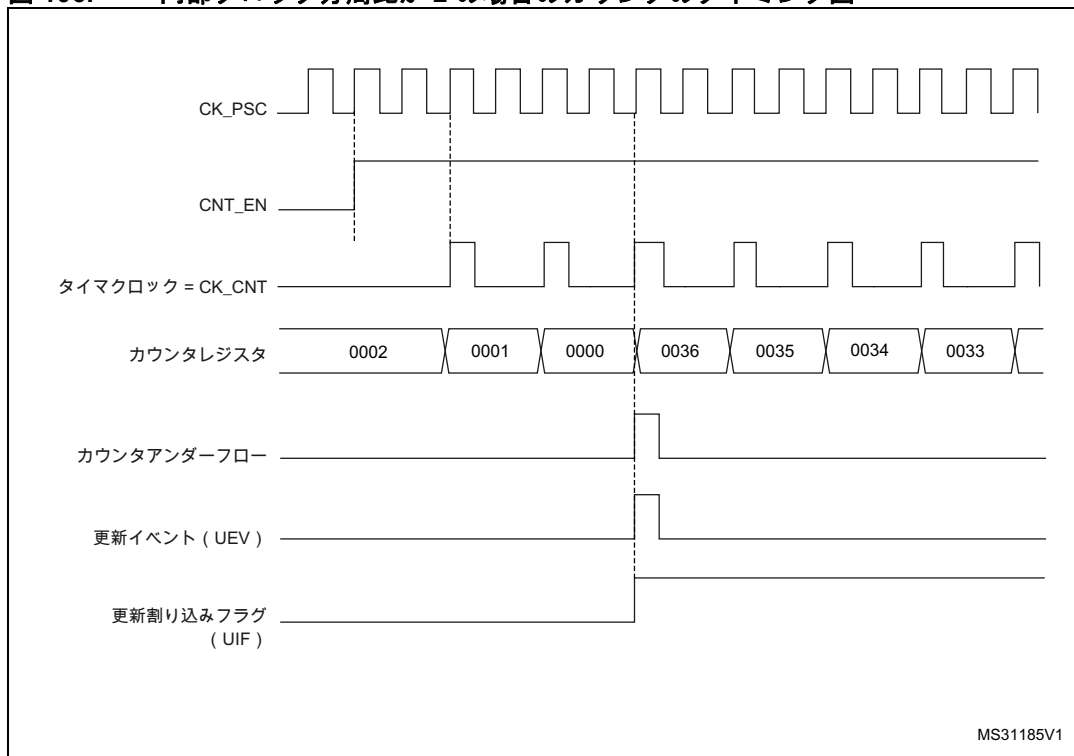


図 107. 内部クロック分周比が 4 の場合のカウンタのタイミング図

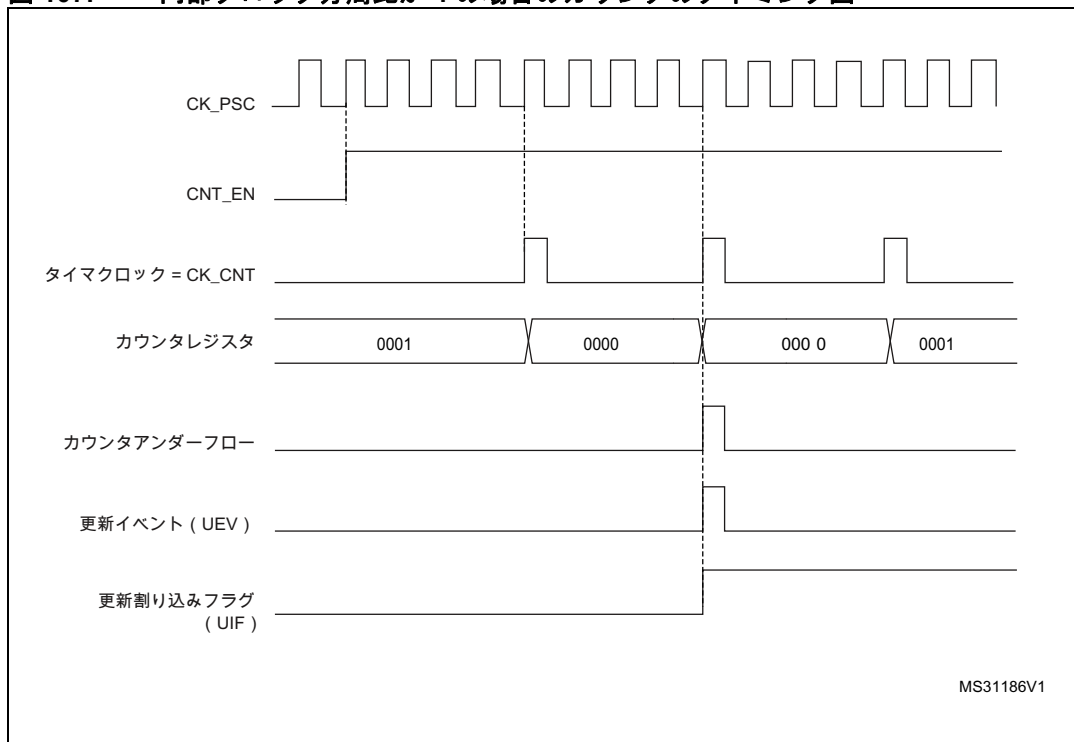


図 108. 内部クロック分周比が N の場合のカウンタのタイミング図

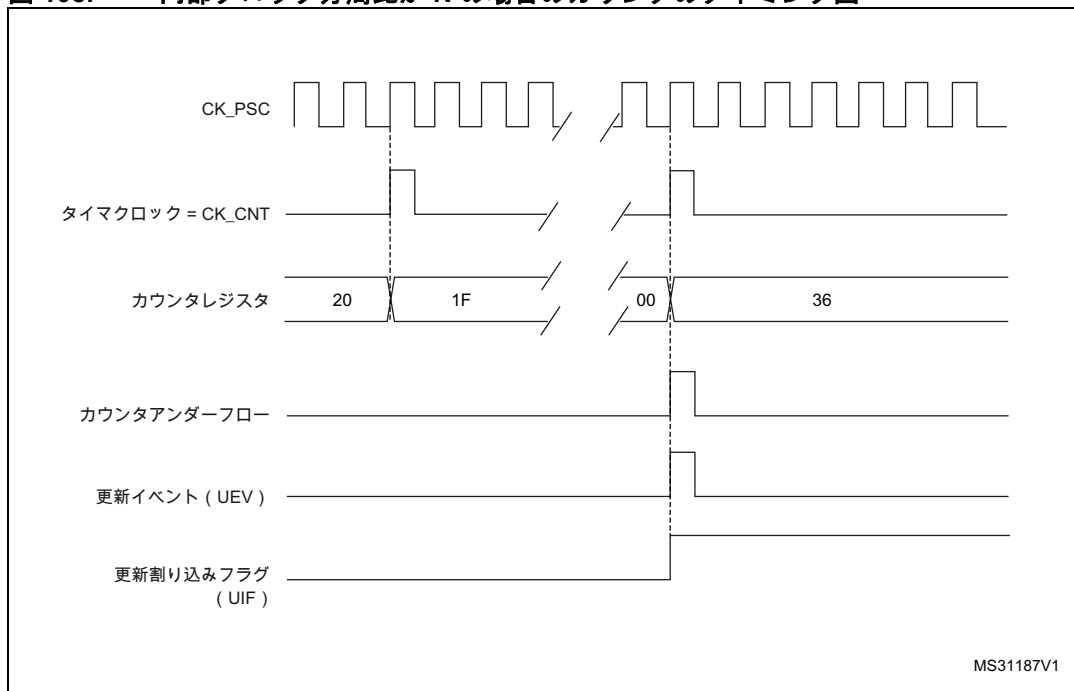
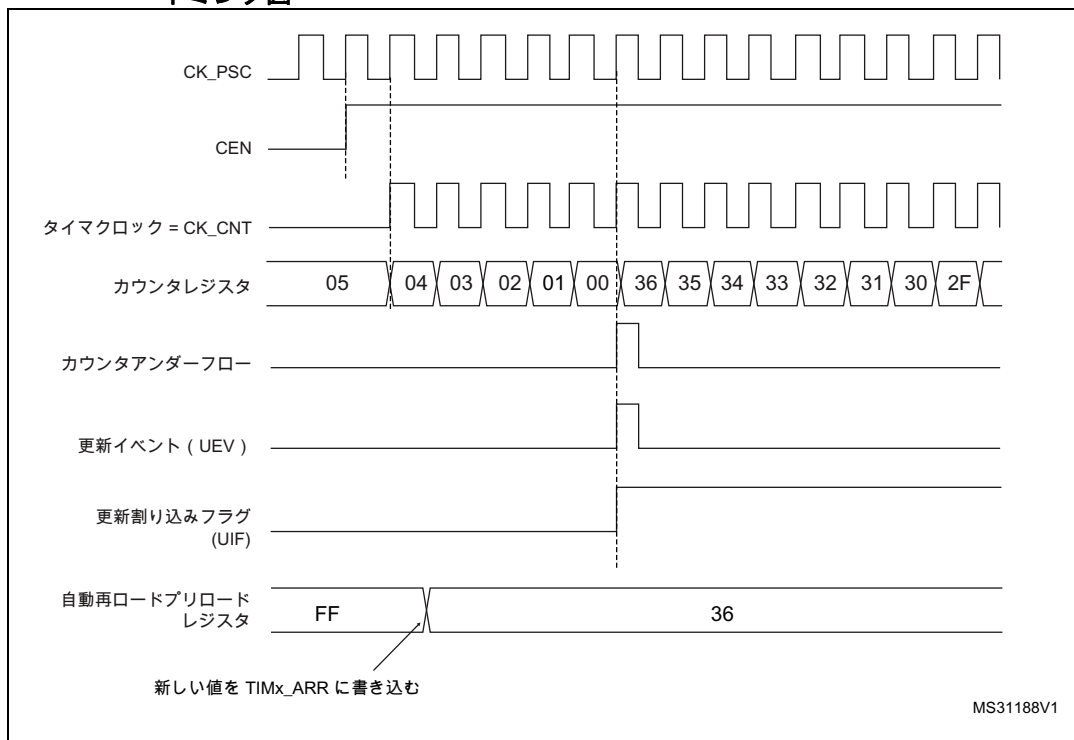


図 109. 繰り返しカウンタが使用されていないときの更新イベント時のカウンタのタイミング図



センターアラインモード (アップ/ダウンカウント)

センターアラインモードでは、カウンタは 0 から自動再ロード値 (TIMx_ARR レジスタの内容) -1 までカウントして、カウンタオーバーフローイベントを生成した後、自動再ロード値から 1 までカウントして、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”に等しくないときにアクティブとなります。出力に設定されたチャンネルの出力比較割り込みフラグは、カウンタがカウントダウンするとき (センターアラインモード 1、CMS=01)、カウンタがカウントアップするとき (センターアラインモード 2、CMS=10)、またはカウンタがカウントアップしてカウントダウンするとき (センターアラインモード 3、CMS=11) にセットされます。

このモードでは、方向ビット (TIMx_CR1 レジスタの DIR) に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローとカウンタアンダーフローごとに生成されます。または、(ソフトウェアで、またはスレーブモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることでも、更新イベントが生成されます。この場合、プリスケアラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

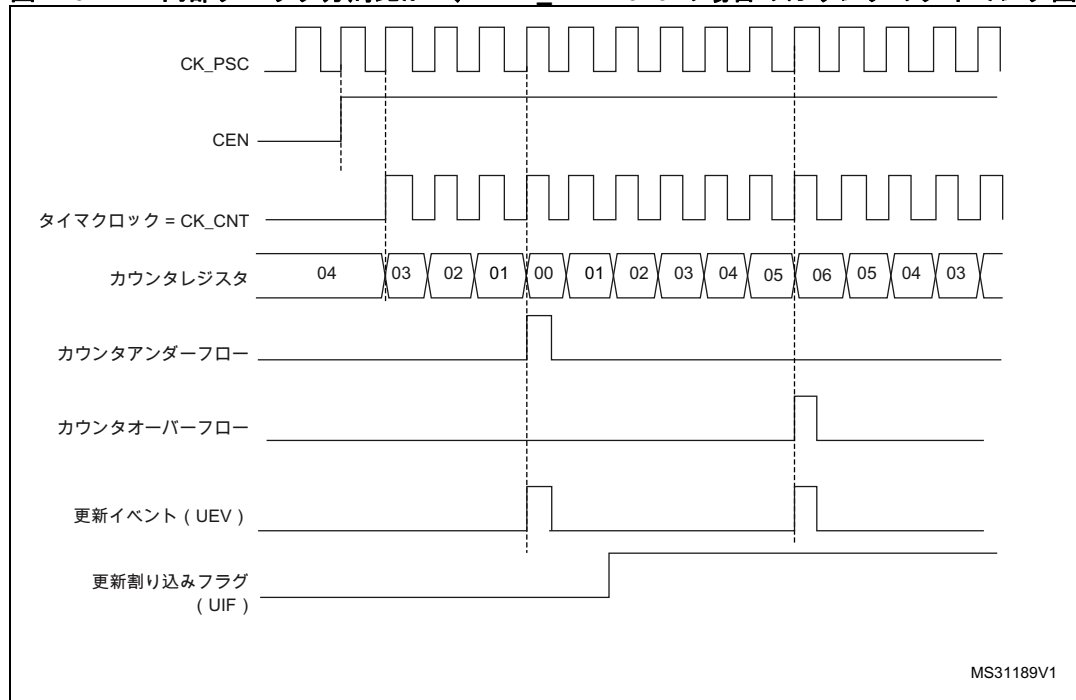
さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのパッファにはプリロード値 (TIMx_PSC レジスタの内容) が再びロードされます。
- 自動再ロードアクティブレジスタは、プリロード値 (TIMx_ARR レジスタの内容) で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は期待通りの周期になります (カウンタに新しい値がロードされます)。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 110. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタのタイミング図



1. ここでは、センタアラインモード 1 が使用されています (詳細については、[セクション 21.4.1 : TIMx 制御レジスタ 1 \(TIMx_CR1\)](#) (450 ページ) を参照してください)。

図 111. 内部クロック分周比が 2 の場合のカウンタのタイミング図

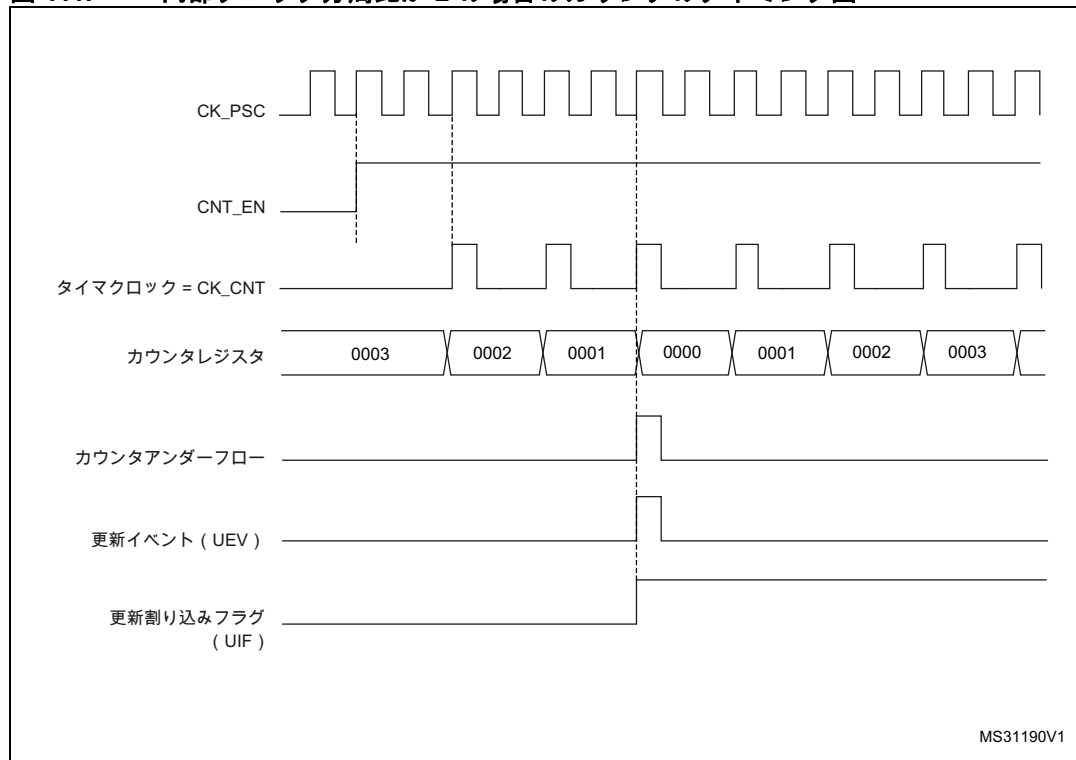
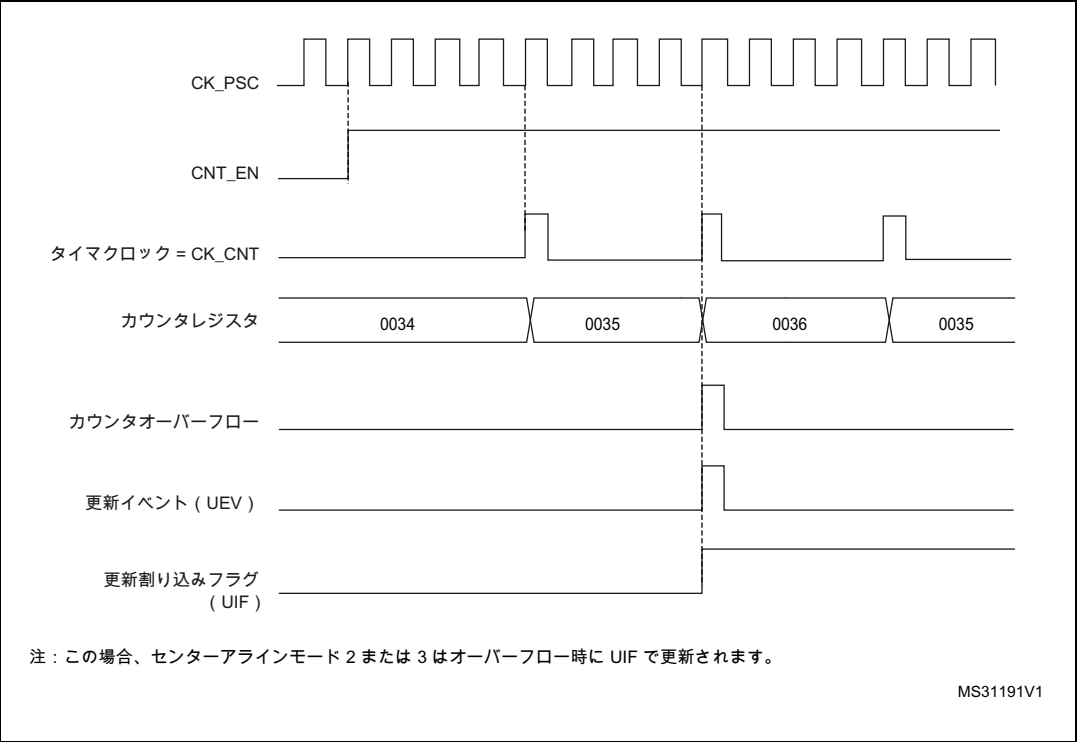


図 112. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 113. 内部クロック分周比が N の場合のカウンタのタイミング図

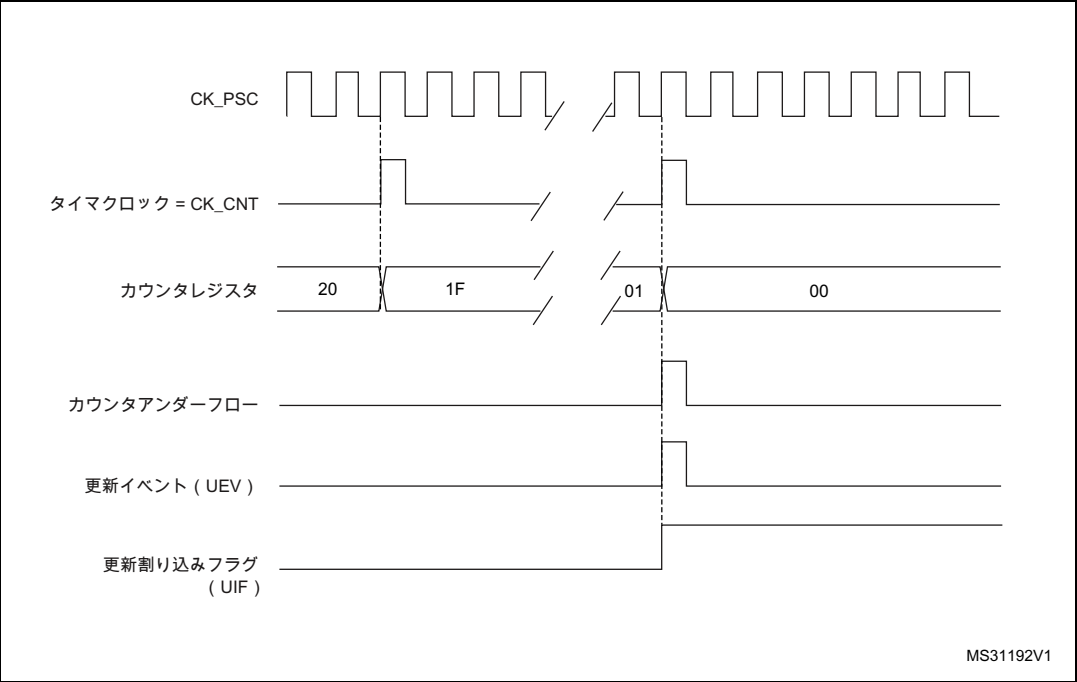


図 114. ARPE=1 (カウンタアンダーフロー) の場合の更新イベント時、カウンタタイミング図

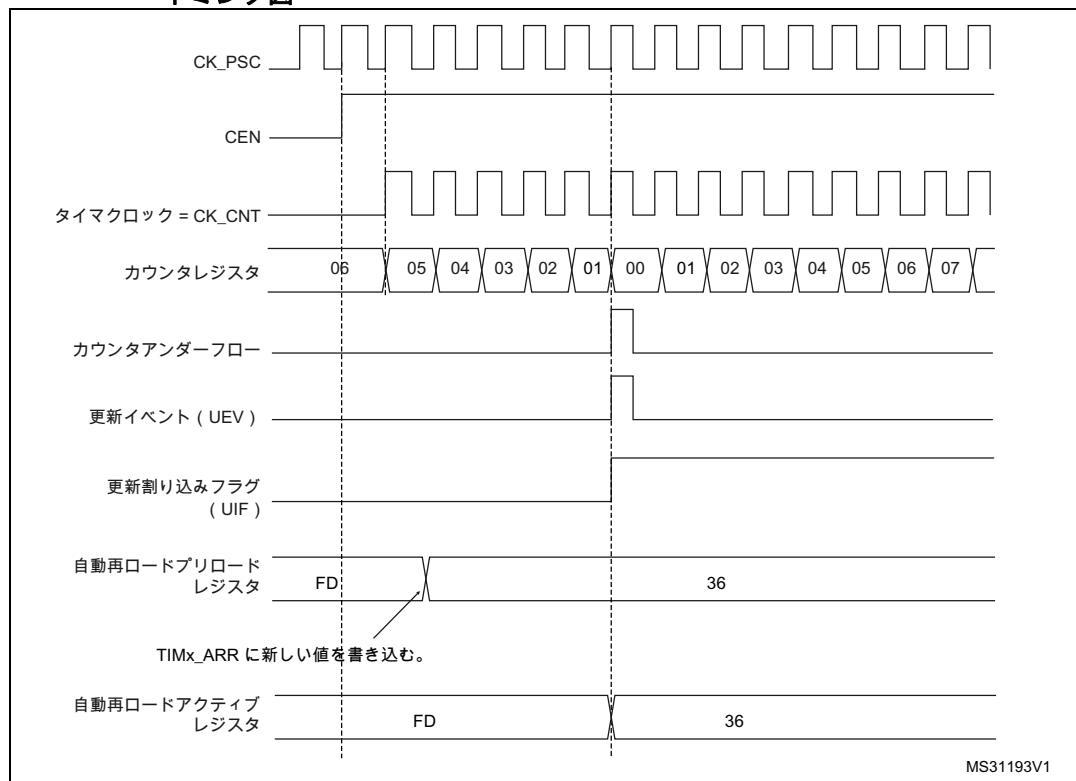
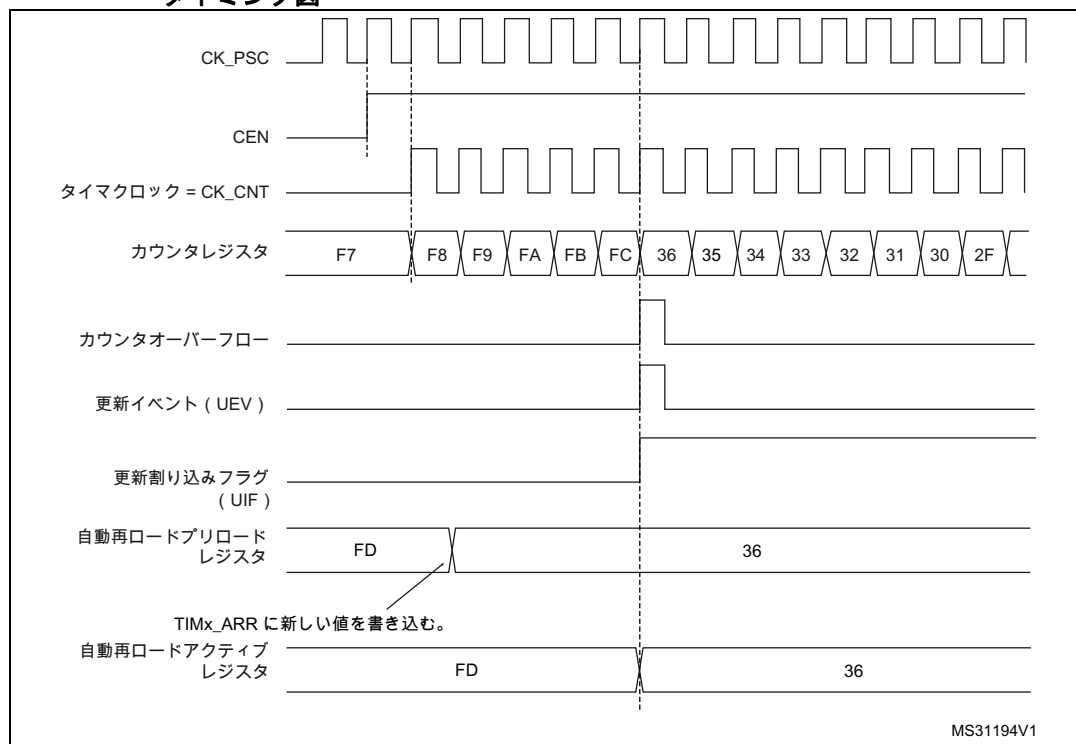


図 115. ARPE=1 (カウンタオーバーフロー) の場合の更新イベント時のカウンタのタイミング図



21.3.3 クロック選択

カウンタクロックは、次のクロックソースによって供給されます。

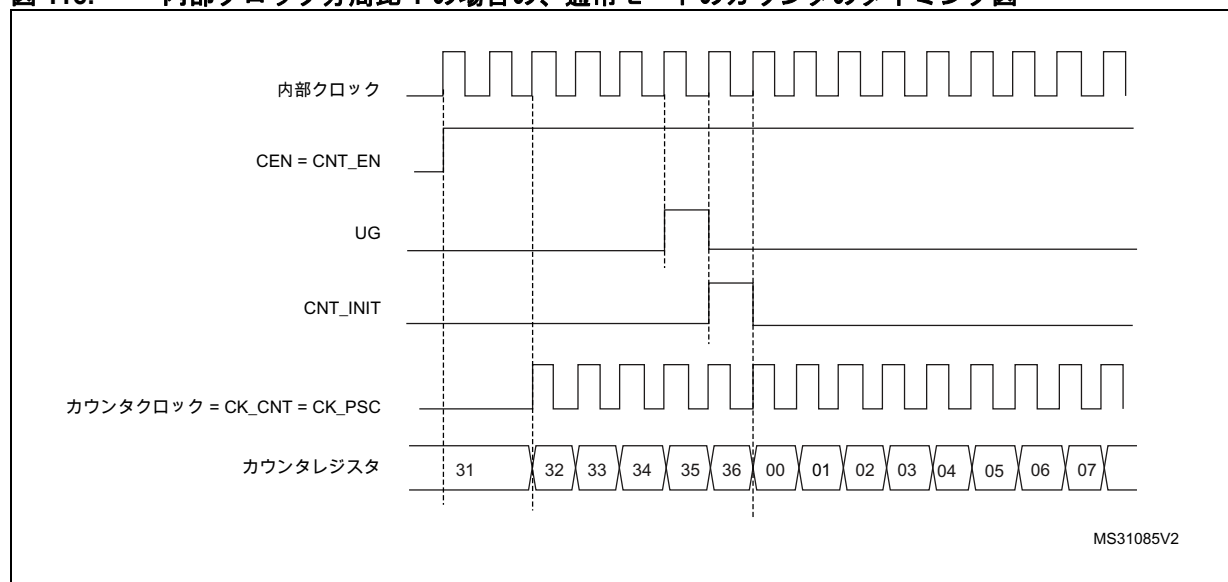
- 内部クロック (CK_INT)
- 外部クロックモード 1: 外部入力ピン (TIX)
- 外部クロックモード 2: 外部トリガ入力 (ETR)
- 内部トリガ入力 (ITRx): タイマを別のタイマのプリスケアラとして使用します。詳細については、[: タイマを別のタイマのプリスケアラとして使用する \(444 ページ\)](#) を参照してください。

内部クロックソース (CK_INT)

スレーブモードコントローラが無効の場合 (TIMx_SMCR レジスタの SMS=000)、CEN、DIR ビット (TIMx_CR1 レジスタ) と UG ビット (TIMx_EGR レジスタ) が実際の制御ビットであり、ソフトウェアでのみ変更できます (自動的にクリアされたままの UG ビットを除きます)。CEN ビットに 1 が書き込まれると、プリスケアラにはクロックとして内部クロック CK_INTが供給されます。

[図 116](#) に、プリスケアラを使用しない場合の制御回路と通常モードのアップカウンタの動作を示します。

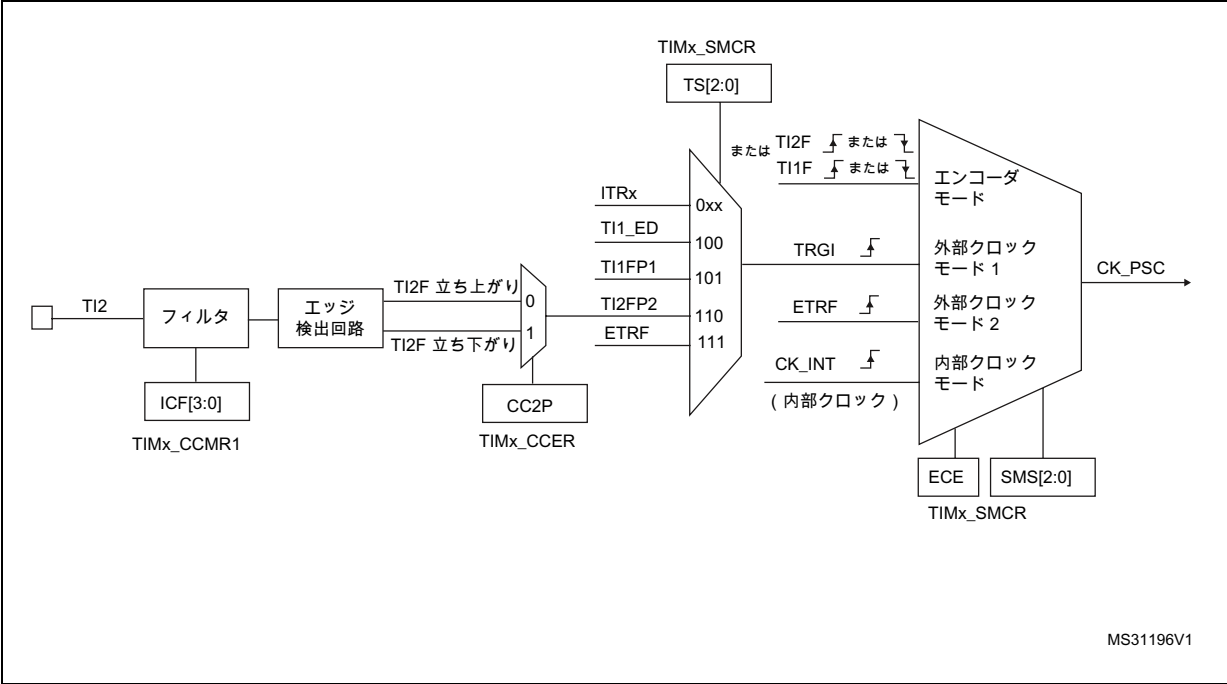
図 116. 内部クロック分周比 1 の場合の、通常モードのカウンタのタイミング図



外部クロックソースモード 1

このモードは TIMx_SMCR レジスタの SMS=111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 117. TIM2 外部クロックの接続例



たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_CCMR1レジスタに CC2S=01 を書き込むことによって、チャンネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F=0000 にしておきます）。

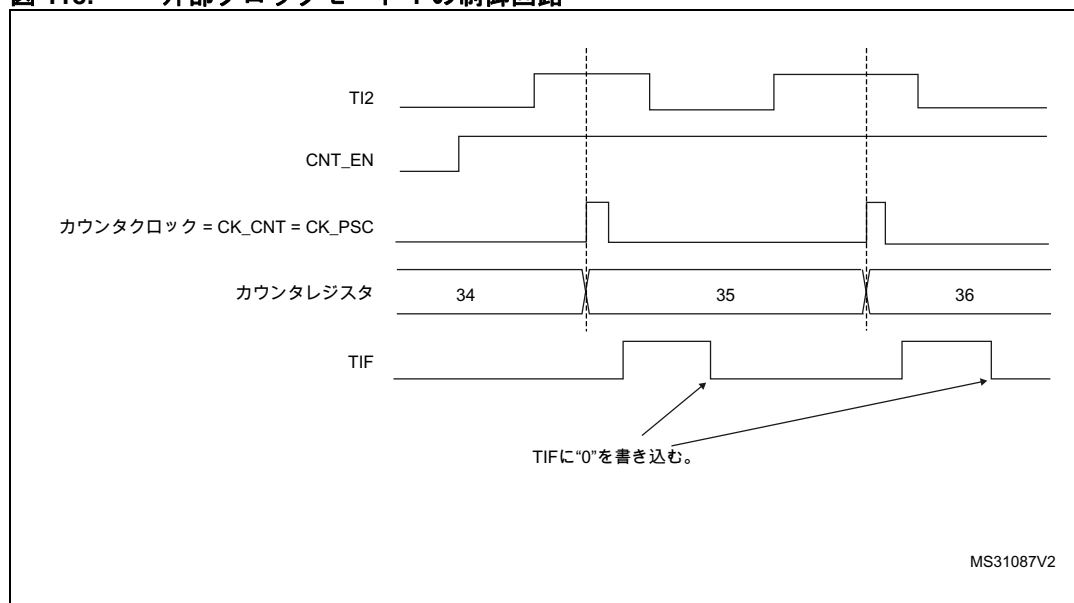
注： キャプチャプリスケアラはトリガには使用されないで、設定は不要です。

3. CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込んで、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタに SMS=111 を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

図 118. 外部クロックモード 1 の制御回路



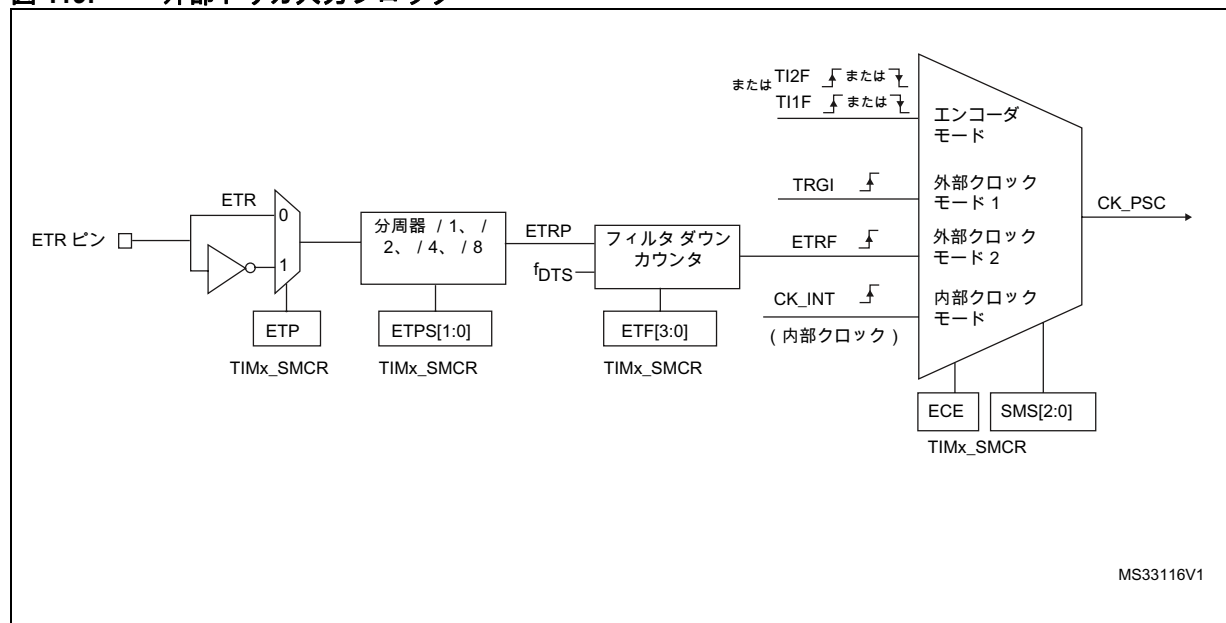
外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE=1 を書き込むことによって選択されます。

カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。

図 119 に、外部トリガ入力ブロックの概要を示します。

図 119. 外部トリガ入力ブロック



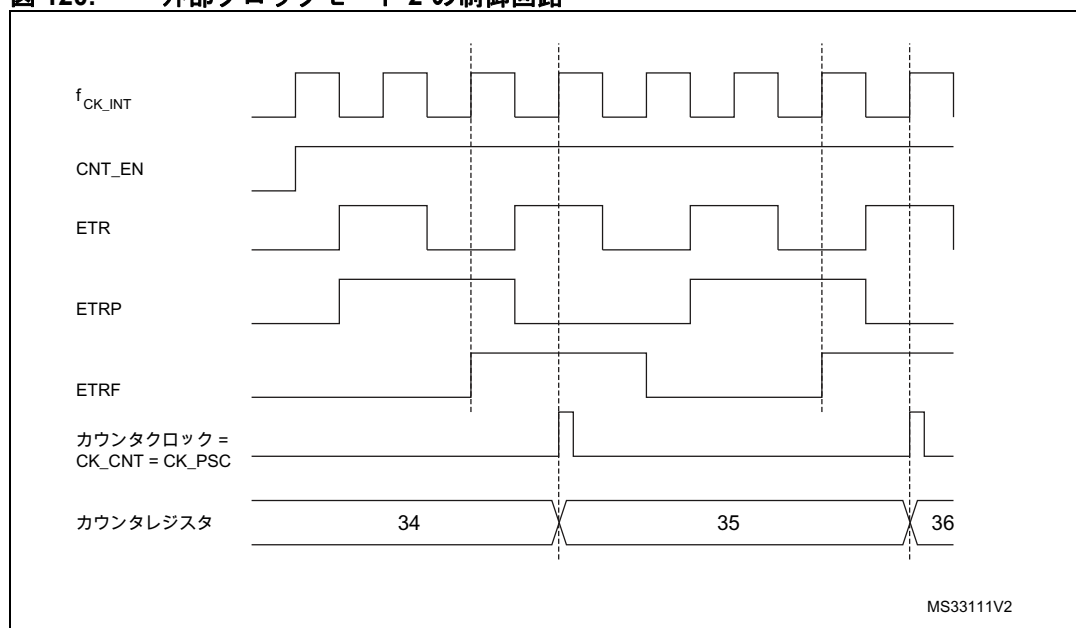
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に 0000 を書き込みます。
2. TIMx_SMCR レジスタに ETPS[1:0]=01 を書き込むことによって、プリスケアラを設定します。
3. TIMx_SMCR レジスタに ETP=0 を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタに ECE=1 を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 120. 外部クロックモード 2 の制御回路



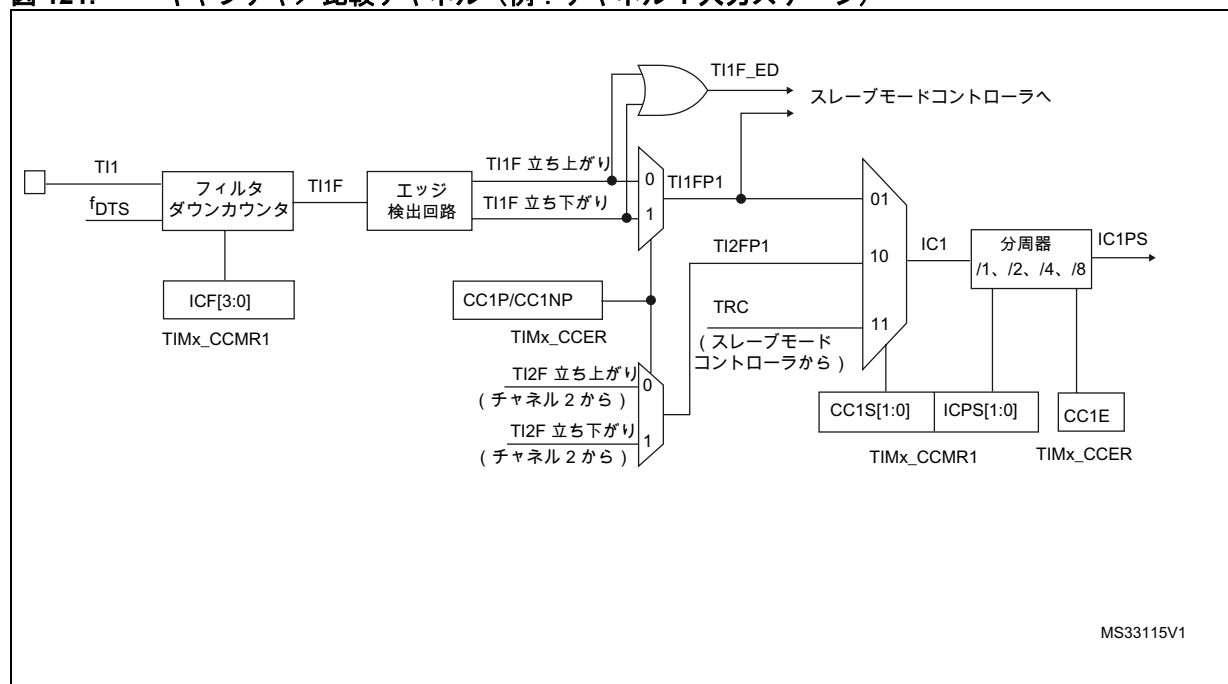
21.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入カステージ（デジタルフィルタ、マルチプレクサ、プリスケアラ）、および出力カステージ（比較回路と出力制御）から構成されています。

次の図に、キャプチャ／比較チャネルの概要を示します。

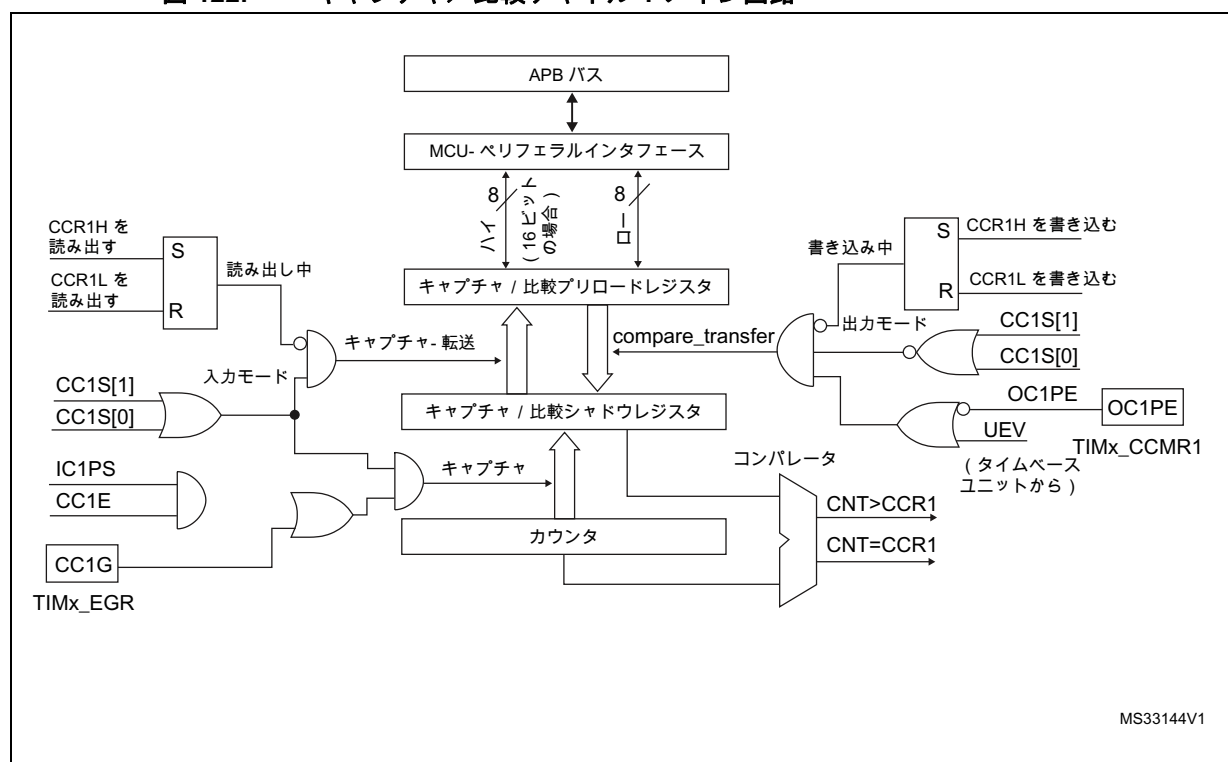
入カステージは、対応する Tl_x 入力をサンプリングして、フィルタリングを行った Tl_xF を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（Tl_xFP_x）を生成します。この信号はプリスケアラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

図 121. キャプチャ／比較チャンネル (例：チャンネル 1 入力ステージ)



出力ステージは、OCxRef（アクティブハイ）として使用される中間波形を生成します。信号の極性は最終出力に影響を与えます。

図 122. キャプチャ／比較チャンネル 1 メイン回路



Block diagram of the TIMx_OC1REF output signal path:

- Inputs:**
 - OCREF_CLR** (input 0 to the first multiplexer)
 - ETRF** (input 1 to the first multiplexer)
 - CNT > CCR1** (input to the output mode controller)
 - CNT = CCR1** (input to the output mode controller)
 - CC1P** (input 1 to the second multiplexer, from TIMx_CCER)
 - CC1E** (input to the output driver, from TIM1_CCER)
- Registers/Blocks:**
 - TIMx_SMCR** (containing **OCCS**)
 - 出力モードコントローラ** (Output Mode Controller)
 - OC1M[2:0]** (Output Mode Register, from TIMx_CCMR1)
 - 出力有効回路** (Output Enable Circuit)
- Signal Flow:**
 - The first multiplexer selects between **OCREF_CLR** and **ETRF** based on the **OCCS** register value. Its output is **ocref_clr_int**.
 - The **出力モードコントローラ** receives **ocref_clr_int** and control signals from **CNT** and **OC1M[2:0]** to generate the **OC1REF** signal.
 - The **OC1REF** signal is also sent to the **マスタモードコントローラ** (Master Mode Controller).
 - The **OC1REF** signal is inverted and then selected by the second multiplexer (along with **CC1P**) to drive the **出力有効回路**.
 - The **出力有効回路** produces the final **OC1** output signal when enabled by **CC1E**.

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

21.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって変化が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ (TIMx_CCRx) が使用されます。キャプチャが発生すると、対応する CCxIF フラグ (TIMx_SR レジスタ) がセットされ、割り込みまたは DMA リクエストを送信できます (有効な場合)。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバーキャプチャフラグ CCxOF (TIMx_SR レジスタ) がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力が立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。このためには、次の手順を使用します。

1. アクティブ入力を選択します。TIMx_CCR1 は TI1 入力とリンクされていなければならない、このためには TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化すると、チャンネルは入力に設定され、TIMx_CCR1 レジスタは読み出し専用になります。
2. タイマに接続する信号に関して、必要な入力フィルタ時間をプログラムします（入力が TIx の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビット）。入力信号の反転時、最大で内部クロックの 5 サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を 5 クロックサイクルより長くプログラミングする必要があります。新しいレベルの連続した 8 個のサンプルが検出されたときに、TI1 で遷移を検証できます（周波数 f_{DTS} でサンプリング）。この場合、TIMx_CCMR1 レジスタの IC1F ビットに 0011 を書き込みます。

3. TI1 チャンネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに “00” を書き込みます（この場合、立ち上がりエッジの選択）。
4. 入力プリスケアラをプログラムします。今回の例では、有効な信号変化ごとにキャプチャを行いたいため、プリスケアラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに 00 を書き込みます）。
5. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
6. 必要な場合は、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にするか、TIMx_DIER レジスタの CC1DE レジスタをセットすることによって、DMA リクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。
- CC1DE ビットに応じて、DMA リクエストが生成されます。

オーバキャプチャを処理するために、オーバキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後、データ読み出し前に発生するオーバキャプチャの見落としを避けることができます。

注： IC 割り込みと DMA リクエストは、TIMx_EGR レジスタの対応する CCxG ビットをセットすることによって、ソフトウェアによって生成することができます。

21.3.6 PWM 入力モード

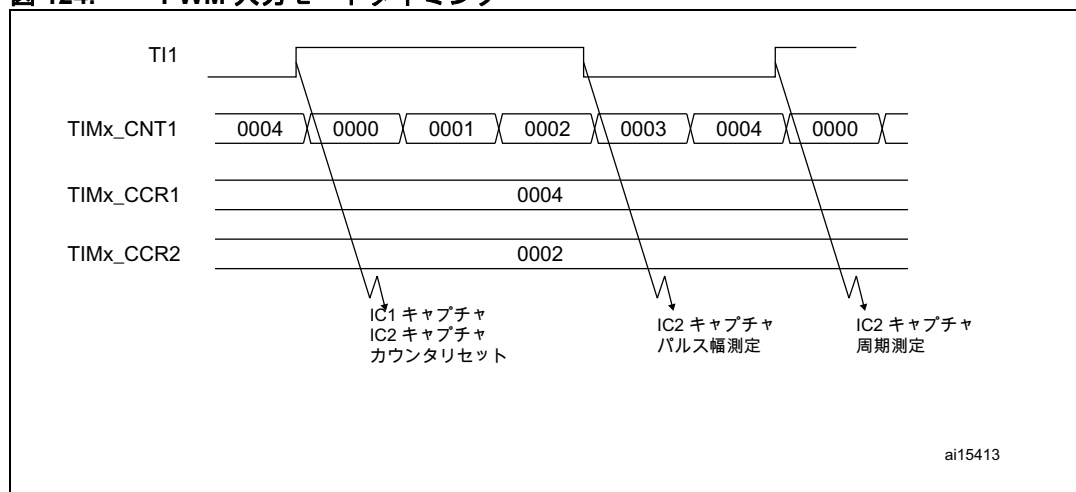
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 2 つの ICx 信号が同じ TIx 入力にマッピングされます。
- この 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期 (TIMx_CCR1 レジスタ) とデューティサイクル (TIMx_CCR2 レジスタ) を測定できます（手順は、CK_INT 周波数とプリスケアラ値によって、若干異なることがあります）。

1. TIMx_CCR1 レジスタの CC1S ビットに“01”を書き込むことによって (TI1 を選択)、TIMx_CCMR1 のアクティブ入力を選択します。
2. CC1P ビットと CC1NP ビットに“0”を書き込むことによって (立ち上がりエッジでアクティブ)、TI1FP1 のアクティブ極性を選択します (TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用します)。
3. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって (TI1 を選択)、TIMx_CCR2 のアクティブ入力を選択します。
4. CC2P ビットと CC2NP ビットに“0”を書き込むことによって (立ち下がりエッジでアクティブ)、TI1FP2 のアクティブ極性を選択します (TIMx_CCR2 のキャプチャに使用されます)。
5. TIMx_SMCR レジスタの TS ビットに 101 を書き込むことによって (TI1FP1 を選択)、有効なトリガ入力を選択します。
6. TIMx_SMCR レジスタの SMS ビットに 100 を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
7. TIMx_CCER レジスタの CC1E と CC2E ビットに 1 を書き込むことによって、キャプチャを有効にします。

図 124. PWM 入力モードタイミング



21.3.7 強制出力モード

このモード (TIMx_CCMRx レジスタの CCxS=00) では、各出力比較信号 (OCxREF、そして OCx) はソフトウェアで強制的にアクティブ、非アクティブのいずれかの状態とされます。これは出力比較レジスタとカウンタの間の比較動作とはかかわりなく行われます。

出力比較信号 (OCxREF/OCx) を強制的にアクティブレベルにするには、対応する TIMx_CCMRx レジスタの OCxM ビットに 101 を書き込みます。これにより、OCxREF は強制的にハイレベルになり (OCxREF は常にアクティブハイ)、OCx は CCxP 極性ビットと逆の値になります。

例：CCxP=0 (OCx アクティブハイ) => OCx は強制的にハイレベルになります。

TIMx_CCMRx レジスタの OCxM ビットに 100 を書き込むことによって、OCxREF 信号を強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットできます。それに応じて、割り込み や DMA リクエストを送信できます。これについては、出力比較モードのセクションで説明します。

21.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ／比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

- 対応する出力ピンに、出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) と出力極性 (TIMx_CCER レジスタの CCxP ビット) によって定義されたプログラム可能値を割り当てます。一致した際、出力ピンは、レベルを維持するか (OCxM=000)、アクティブにセットされるか (OCxM=001)、非アクティブにセットされるか (OCxM=010)、または反転されます (OCxM=011)。
- 割り込みステータスレジスタのフラグをセットします (TIMx_SR レジスタの CCxIF ビット)。
- 対応する割り込みマスク (TIMx_DIER レジスタの CCxIE ビット) がセットされている場合は、割り込みを生成します。
- 対応するイネーブルビット (TIMx_DIER レジスタの CCxDE ビット) がセットされている場合は、DMA リクエストを送信します (DMA リクエスト選択には、TIMx_CR2 レジスタの CCDS ビットが使用されます)。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_CCMRx レジスタの OCxPE ビットを使用してプログラミングできます。

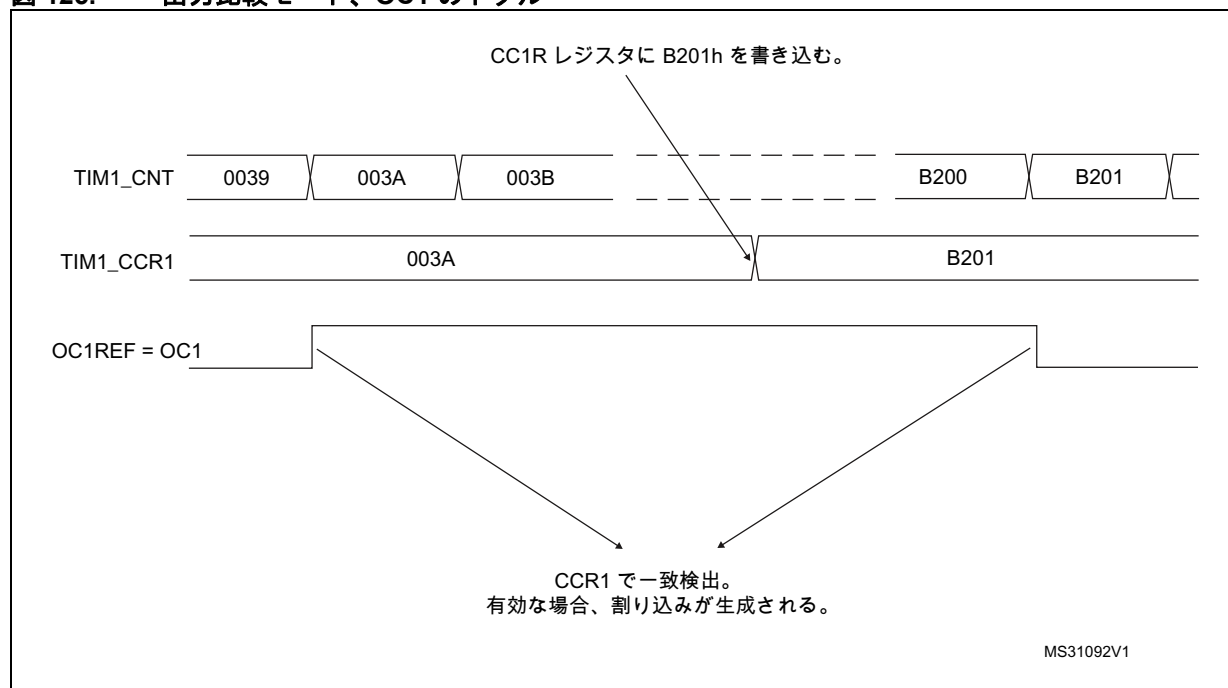
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます (ワンパルスモード)。

手順：

1. カウンタクロックを選択します (内部、外部、プリスケアラ)。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエスト／DMA リクエストを生成する場合は、CCxIE ビット／CCxDE ビットをセットします。
4. 出力モードを選択します。たとえば、CNT が CCRx と一致したときに OCx 出力をトグルし、CCRx プリロードを使用せず、OCx が有効でアクティブハイのときには、OCxM=011、OCxPE=0、CCxP=0、CCxE=1 を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアで TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限り (OCxPE=0)。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を [図 125](#) に示します。

図 125. 出力比較モード、OC1 のトグル



21.3.9 PWM モード

パルス幅変調 (PWM) モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、個々のチャネル (OCx 出力ごとに PWM 1 波形) で、TIMx_CCMRx レジスタの OCxM ビットに "110" (PWM モード 1) や "111" (PWM モード 2) を書き込むことで、独自に選択できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があります。また、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも (アップカウントまたはセンターアラインモードで) 有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx 極性は、TIMx_CCER レジスタの OCxP ビットを使用して、ソフトウェアでプログラム可能です。アクティブハイまたはアクティブローとしてプログラムできます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効になります。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード (1 または 2) では、TIMx_CNT と TIMx_CCRx が常に比較されて、TIMx_CCRx ≤ TIMx_CNT または TIMx_CNT ≤ TIMx_CCRx がどうか判断されます (カウントの方向によります)。ただし、OCREF_CLR 機能 (OCREF は、次の PWM 周期までは ETR 信号を通じて外部イベントによってクリアできる) に従って、OCREF 信号は次の場合にのみアサートされます。

- 比較結果が変化するとき、または
- 出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) が停止構成 (比較なし、OCxM=000) から PWM モードの 1 つ (OCxM=110 または 111) へ切り替えられたとき。

タイマの動作中は、ソフトウェアで強制的に PWM になります。

タイマは、TIMx_CR1 レジスタの CMS ビットに応じて、エッジアラインモードまたはセンターアラインモードで PWM を生成できます。

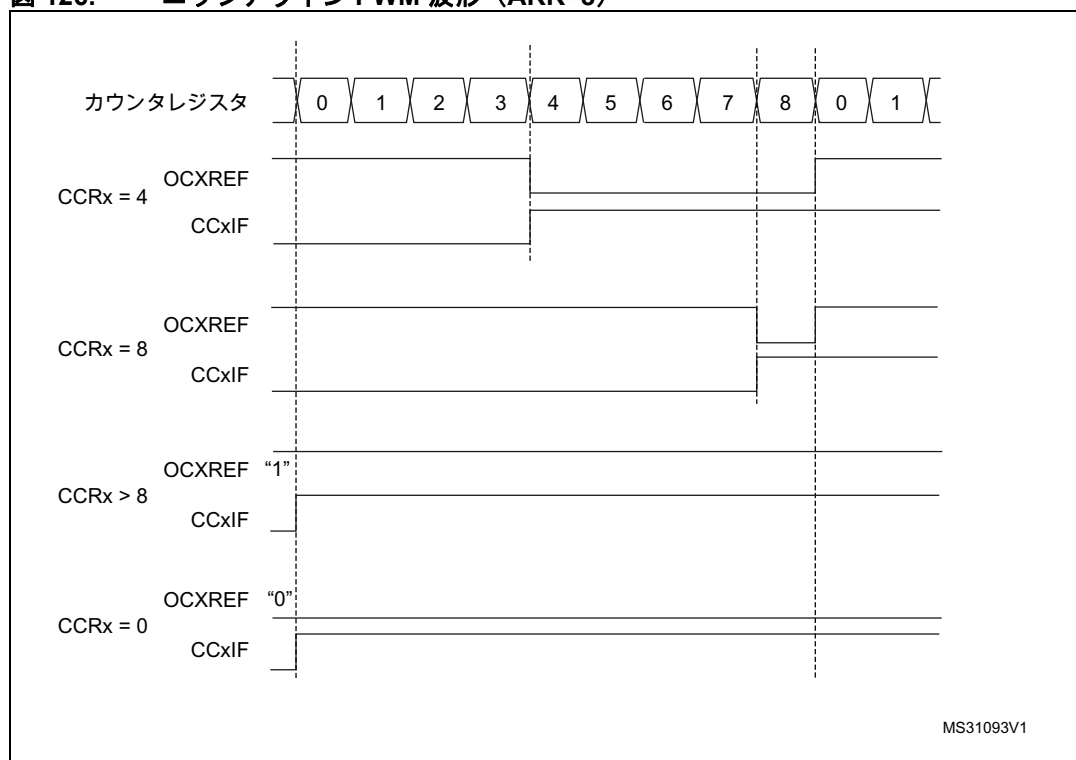
PWM エッジアラインモード

アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。 [セクション：アップカウントモード \(413 ページ\)](#) を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、TIMx_CNT < TIMx_CCRx の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値 (TIMx_ARR レジスタの) より大きい場合、OCxREF は“1”です。比較値が 0 の場合、OCxREF は“0”に保持されます。 [図 126](#) に TIMx_ARR=8 のときのエッジアライン PWM 波形の例を示します。

図 126. エッジアライン PWM 波形 (ARR=8)



ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。 [セクション：ダウンカウントモード \(417 ページ\)](#) を参照してください。

PWM モード 1 では、基準信号 OCxREF は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、OCxREF は“1”です。このモードでは、0 % の PWM 信号を生成することはできません。

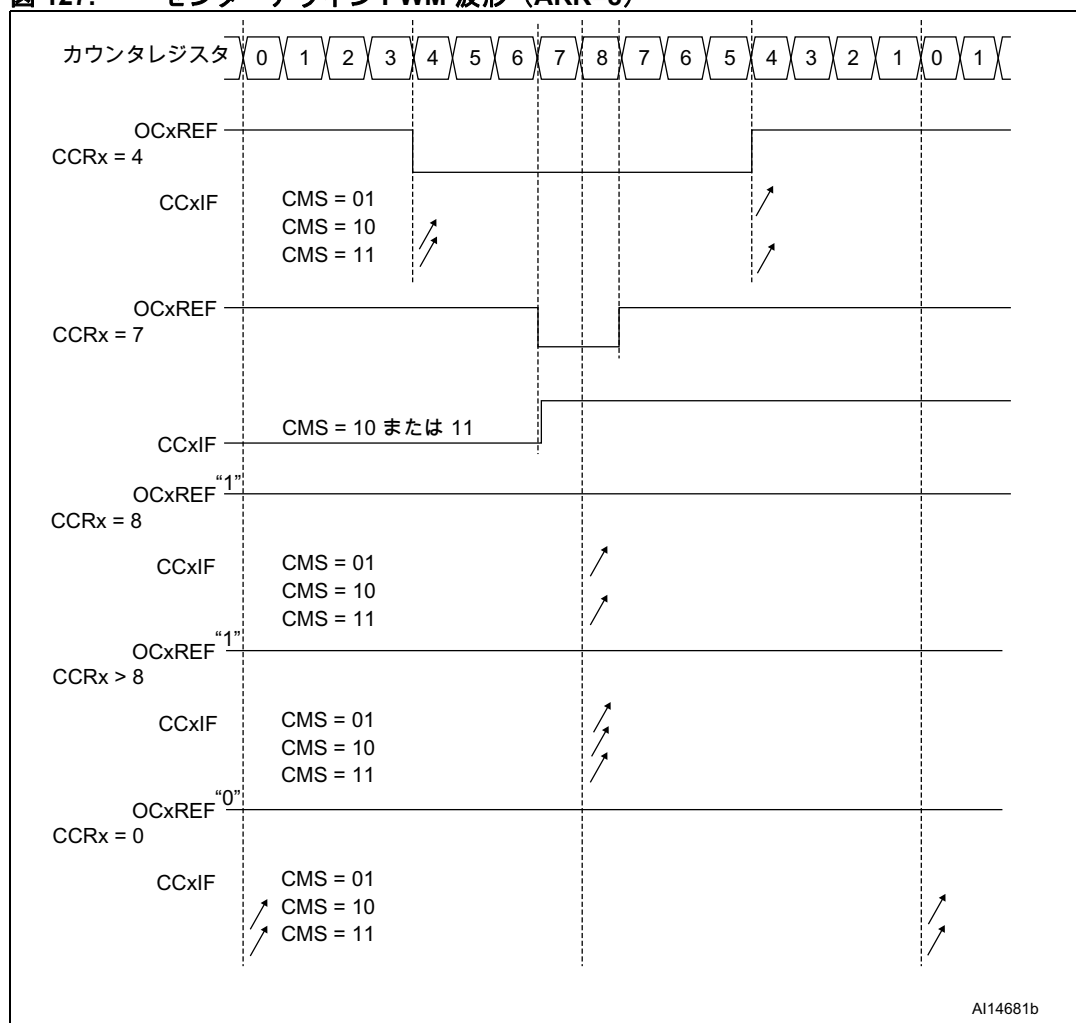
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての構成は、OCxREF/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット（DIR）はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。[セクション：センターアラインモード（アップ/ダウンカウント）（420 ページ）](#)を参照してください。

図 127 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR=8
- PWM モードは PWM モード 1
- フラグは、TIMx_CR1 レジスタの CMS=01 で選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 127. センターアライン PWM 波形 (ARR=8)



センターアラインモードの使用に関するヒント：

- センターアラインモードを開始するときには、現在のアップ/ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、
 - 自動再ロード値より大きい値をカウンタに書き込んだ場合 (TIMx_CNT > TIMx_ARR)、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
 - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成して (TIMx_EGR レジスタの UG ビットをセットする)、動作中はカウンタへの書き込みを行わないことです。

21.3.10 ワンパルスモード

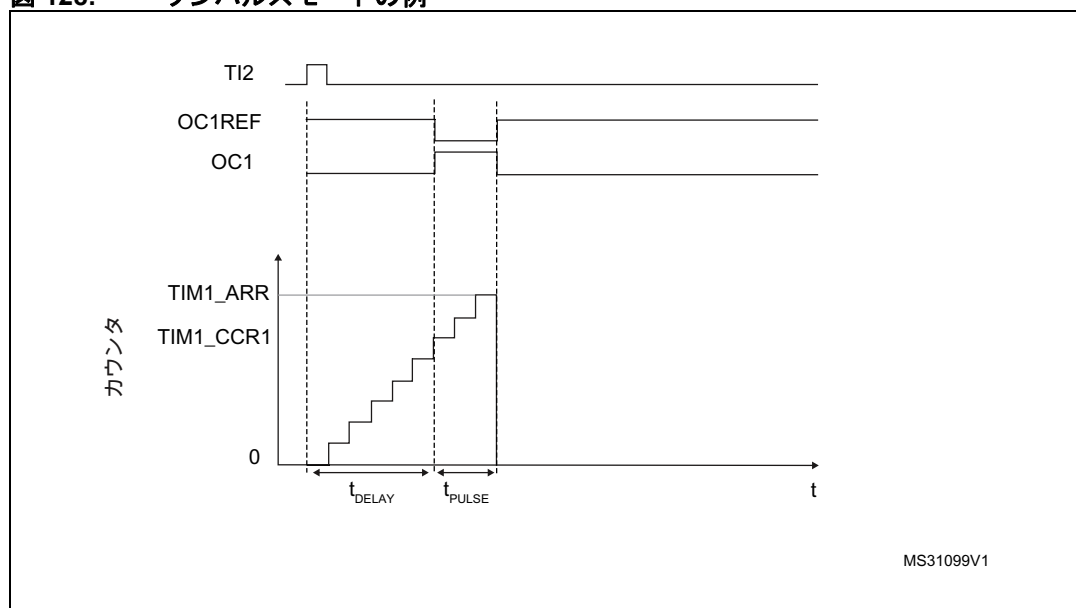
ワンパルスモード (OPM : One Pulse Mode) は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラを通じて制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に (タイマがトリガを待っているときに)、設定が次のようであればなりません。

- アップカウント時 : $CNT < CCRx \neq ARR$ (特に、 $0 < CCRx$)
- ダウンカウント時 : $CNT > CCRx$

図 128. ワンパルスモードの例



たとえば、TI2 入力ピンで立ち上がりエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} 後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタに IC2S=01 を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタで CC2P= 0 と CC2NP = 0 を書き込みます。
- TI2FP2 をスレーブモードコントローラのトリガ (TRGI) として構成するために、TIMx_SMCR レジスタの TS="110"を書き込みます。
- TI2FP2 を使用してカウンタを開始するために、TIMx_SMCR レジスタの SMS ビットに"110" (トリガモード) を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます (クロック周波数とカウンタプリスケアラを考慮に入れて)。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差 (TIMx_ARR - TIMx_CCR1) によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M=111 を書き込むことによって、PWM モード 2 を有効にします。オプションで、TIMx_CCMR1 レジスタの OC1PE=1 と TIMx_CR1 レジスタの ARPE=1 を書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイベントを待ちます。この例では、CC1P に"0"を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので (シングルモード)、TIMx_CR1 レジスタの OPM ビットに"1"を書き込みます。こうすると、カウンタは次の更新イベント時に停止します (カウンタが自動再ロード値に達して、"0"に戻る時点)。TIMx_CR1 レジスタの OPM ビットが"0"にセットされると、繰り返しモードが選択されます。

特殊なケース : OCx 高速イネーブル :

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延 ($t_{DELAY min}$) が制限されます。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF (および OCx) は、比較を考慮せずにトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャンネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

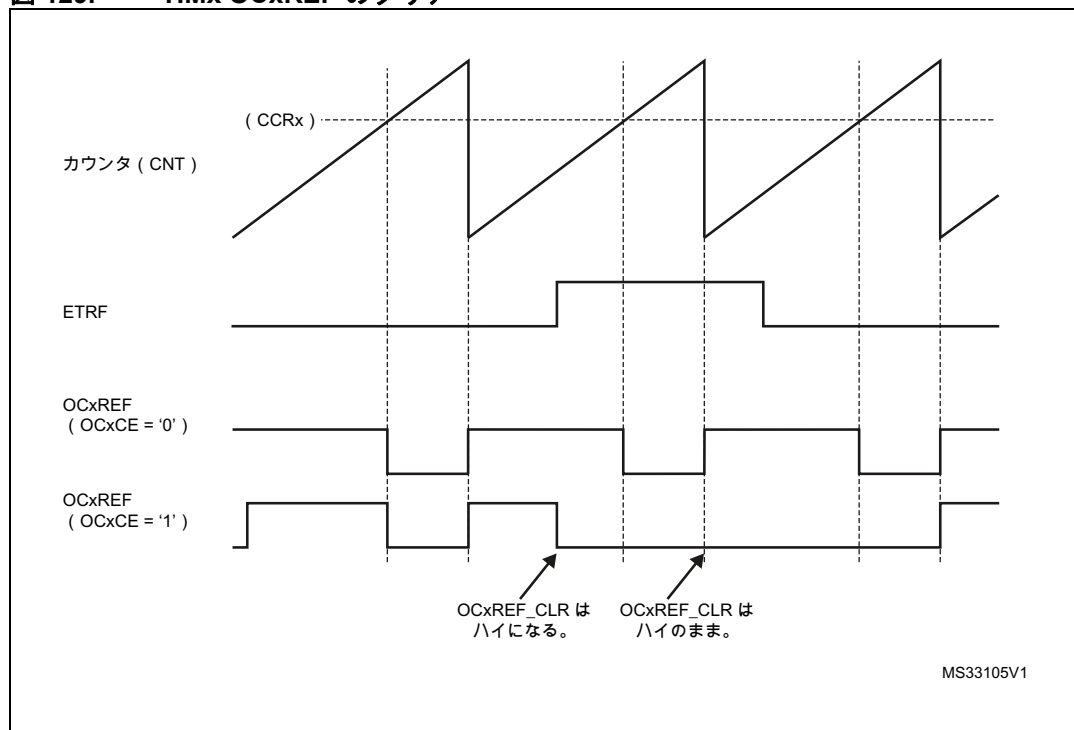
21.3.11 外部イベントによる OCxREF 信号のクリア

1. 外部トリガプリスケアラをオフに保つ必要があります。すなわち、TIMx_SMCR レジスタのビット ETPS[1:0] が 00 にクリアされます。
2. 外部クロックモード 2 を無効にする必要があります。すなわち、TIMx_SMCR レジスタのビット ECE が 0 にクリアされます。
3. 外部トリガ極性 (ETP) と外部トリガフィルタ (ETF) は、アプリケーションのニーズに応じて設定できます。



図 129 に、OCxCE イネーブルビットの両方の値について、ETRF 入力が高くなったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラミングされています。

図 129. TIMx OCxREF のクリア



1. 100% デューティサイクルの PWM の場合 (CCRx>ARR の場合)、次のカウンタオーバーフローで OCxREF が再度有効になります。

21.3.12 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタで、カウンタが TI2 エッジのみをカウントしている場合は SMS="001"を、TI1 エッジのみをカウントしている場合は SMS="010"を、TI1 と TI2 の両方のエッジをカウントしている場合は SMS="011"を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラミングします。CC1NP と CC2NP はクリア状態に維持する必要があります。必要なときには、入力フィルタもプログラミングできます。

2つの入力 TI1 と TI2 は、インクリメンタルエンコーダとのインタフェースに使用されます。表 79 を参照してください。カウンタのクロックは、TI1FP1 または TI2FP2 (入力フィルタおよび極性選択後は TI1 および TI2。フィルタされず、反転されない場合は TI1FP1=TI1、フィルタされず、反転されない場合は TI2FP2=TI2) の有効な変化によって駆動されます。ただし、カウンタ有効なことが前提となります (TIMx_CR1 レジスタの CEN ビットが"1")。2つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIR ビットは、いずれかの入力 (TI1 または TI2) の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。カウンタは、0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします (方向に応じて、0 から

ARR まで、または ARR から 0 まで)。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケアラ、およびトリガ出力機能は、通常動作を続けます。

このモードでは、カウンタはインクリメンタルエンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します (TI1 と TI2 は同時に切り替わらないと想定しています)。

表 79. カウント方向とエンコーダ信号

| アクティブ エッジ | 他方の信号のレベル (TI2 に対する TI1FP1、 TI1 に対する TI2FP2) | TI1FP1 信号 | | TI2FP2 信号 | |
|------------------------|--|-----------|--------|-----------|--------|
| | | 立ち上がり | 立ち下がり | 立ち上がり | 立ち下がり |
| TI1 のみ カウント | ハイ | ダウン | アップ | カウントなし | カウントなし |
| | ロー | アップ | ダウン | カウントなし | カウントなし |
| TI2 のみ カウント | ハイ | カウントなし | カウントなし | アップ | ダウン |
| | ロー | カウントなし | カウントなし | ダウン | アップ |
| TI1 と TI2 の 両方をカウント | ハイ | ダウン | アップ | アップ | ダウン |
| | ロー | アップ | ダウン | ダウン | アップ |

外部インクリメンタルエンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

[図 130](#) に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S=01 (TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置)
- CC2S= 01 (TIMx_CCMR2 レジスタ、TI2FP2 は TI2 に配置)
- CC1P=0、CC1NP="0" (TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1)
- CC2P=0、CC2NP="0" (TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2=TI2)
- SMS=011 (TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下がりの両エッジでアクティブ)
- CEN=1 (TIMx_CR1 レジスタ、カウンタ有効)

図 130. エンコーダインタフェースモードにおけるカウンタの動作例

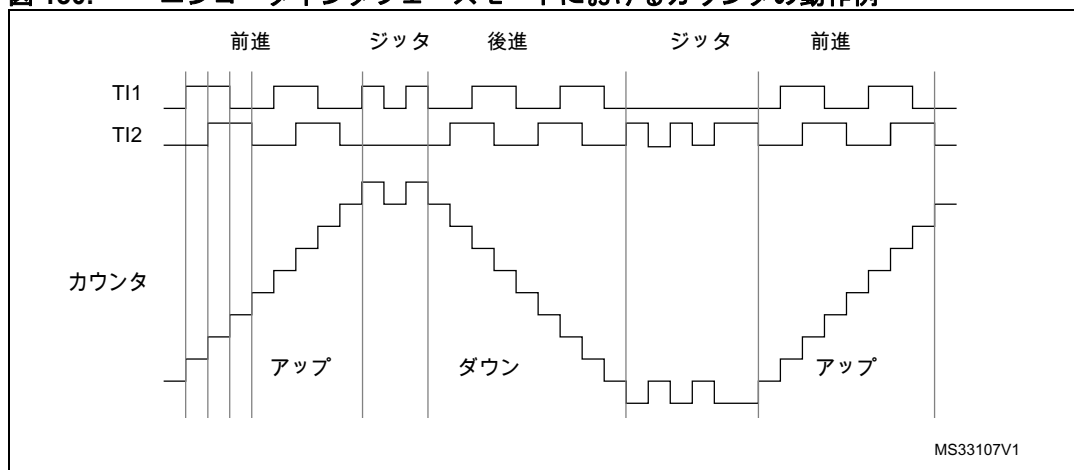
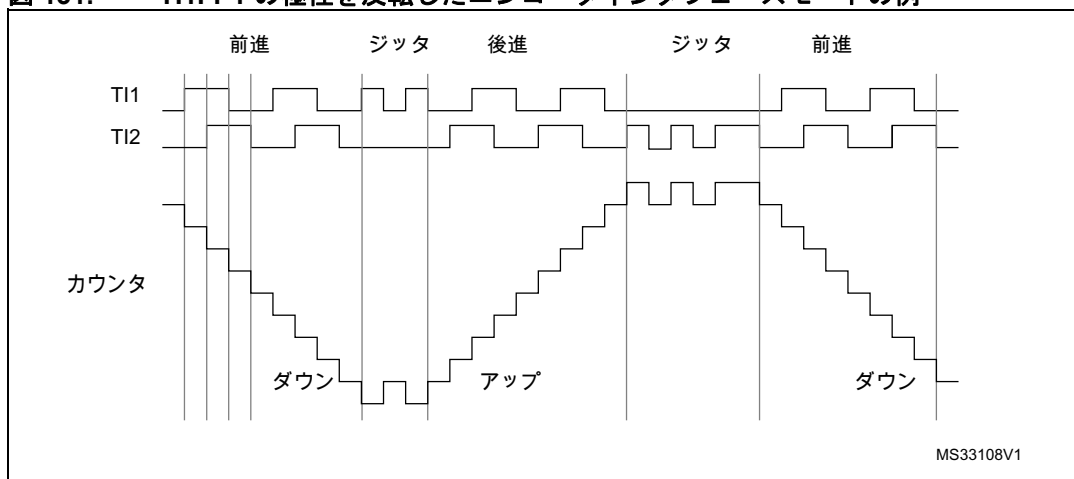


図 131 に、TI1FP1 の極性を反転したときのカウンタの動作を示します（上記と同じ設定ですが、CC1P=1）。

図 131. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに構成した 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします（キャプチャ信号は周期的でなければならない、別のタイマによって生成できます）。使用可能なときには、リアルタイムクロックによって生成される DMA リクエストを通じて値を読み出すことも可能です。

21.3.13 タイマ入力 XOR 機能

TIMx_CR2 レジスタの TI1S ビットを使用して、チャンネル 1 の入力フィルタを、TIMx_CH1 から TIMx_CH3 までの 3 つの入力ピンを組み合わせさせた XOR ゲートの出力に接続できます。

XOR 出力は、トリガや入力キャプチャなど、すべてのタイマ入力機能で使用できます。

21.3.14 タイマと外部トリガの同期

TIMx タイマは、リセットモード、ゲートモード、およびトリガモードで、外部トリガに同期できます。

スレーブモード：リセットモード

カウンタとそのプリスケアラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ (TIMx_ARR、TIMx_CCRx) が更新されます。

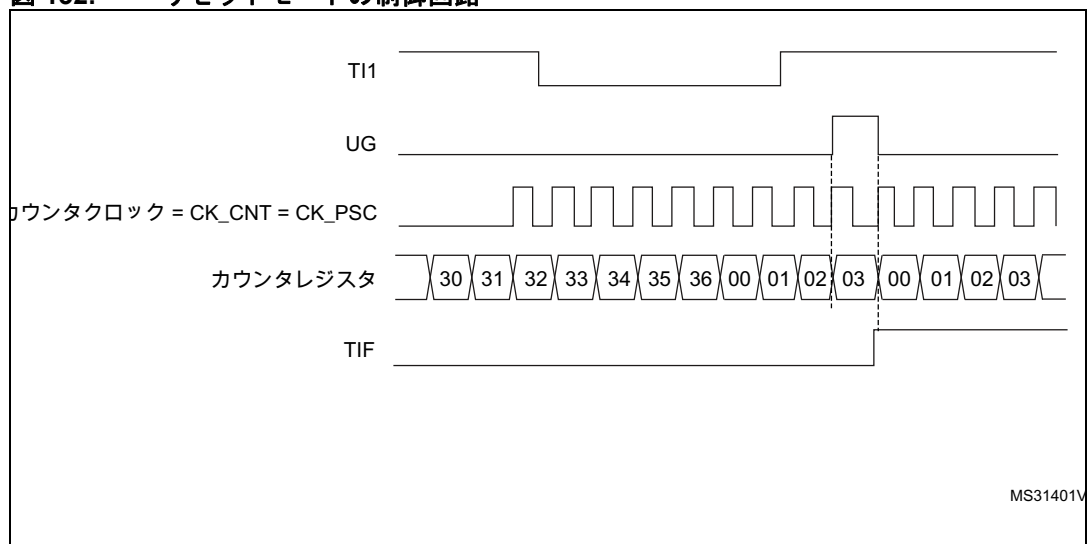
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

- TI1 の立ち上がりエッジを検出するように、チャンネル 1 を設定します。入力フィルタ時間を設定します (この例ではフィルタは不要なので、IC1F=0000 のままにしておきます)。キャプチャプリスケアラはトリガには使用されないため、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します (TIMx_CCMR1 レジスタの CC1S=01)。TIMx_CCER レジスタで CC1P=0 と CC1NP=0 を書き込んで、極性を有効にします (そして、立ち上がりエッジのみを検出します)。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
- TIMx_CR1 レジスタに CEN=1 を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ (TIMx_SR レジスタの TIF ビット)、有効な場合は割り込みリクエストまたは DMA リクエストを送信できます (TIMx_DIER レジスタの TIE および TDE ビット)。

次の図は、自動再ロードレジスタ TIMx_ARR=0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 132. リセットモードの制御回路



スレープモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

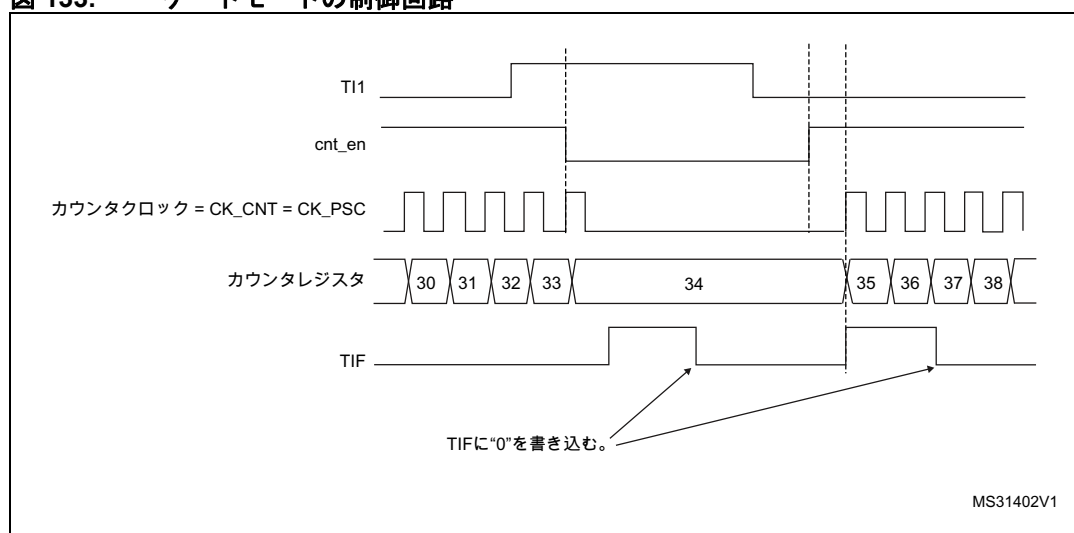
次の例では、アップカウンタは TI1 入力が高レベルのときだけカウントします。

1. TI1 の高レベルを検出するように、チャンネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケールはトリガには使用されないため、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S=01 ビット）。TIMx_CCER レジスタの CC1P=1 と CC1NP=0 を書き込んで、極性を有効にします（そして、高レベルのみを検出します）。
2. TIMx_SMCR レジスタに SMS=101 を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで、カウンタを有効にします（ゲートモードでは、CEN=0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 が高レベルになると内部クロックでカウントを開始して、TI1 が低レベルになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 133. ゲートモードの制御回路



1. ゲートモードはエッジではなくレベルに対して動作するため、CCxP=CCxNP=1 の設定（立ち上がり立ち下がり両エッジの検出）はゲートモードでは意味がありません。

スレープモード：トリガモード

選択された入力のイベントに対応して、カウンタを開始できます。

次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

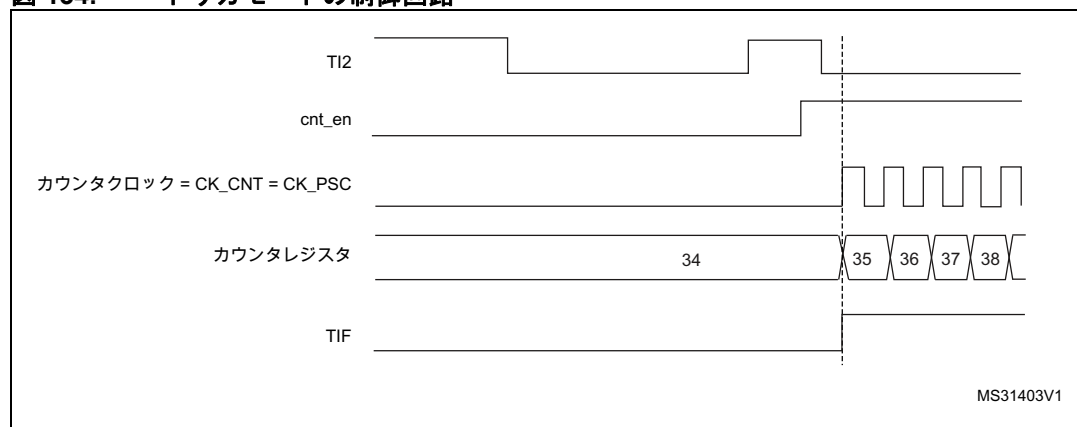
1. TI2 の立ち上がりエッジを検出するように、チャンネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F=0000 のままにしておきます）。キャプチャプリスケールはトリガには使用されないため、設定は不要です。CC2S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC2S=01）。TIMx_CCER レジスタで CC2P=1 と CC2NP=0 を書き込んで、極性を有効にします（そして、高レベルのみを検出します）。

2. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=110 を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 134. トリガモードの制御回路



スレーブモード：外部クロックモード 2 + トリガモード

外部クロックモード 2 は、他のスレーブモードとともに使用できます（ただし、外部クロックモード 1 とエンコーダモードは除きます）。この場合、ETR 信号は外部クロック入力として使用され、別の入力をトリガ入力として選択できます（リセットモード、ゲートモード、およびトリガモードで動作している場合）。TIMx_SMCR レジスタの TS ビットを通じて TRGI として ETR を選択しないようにしてください。

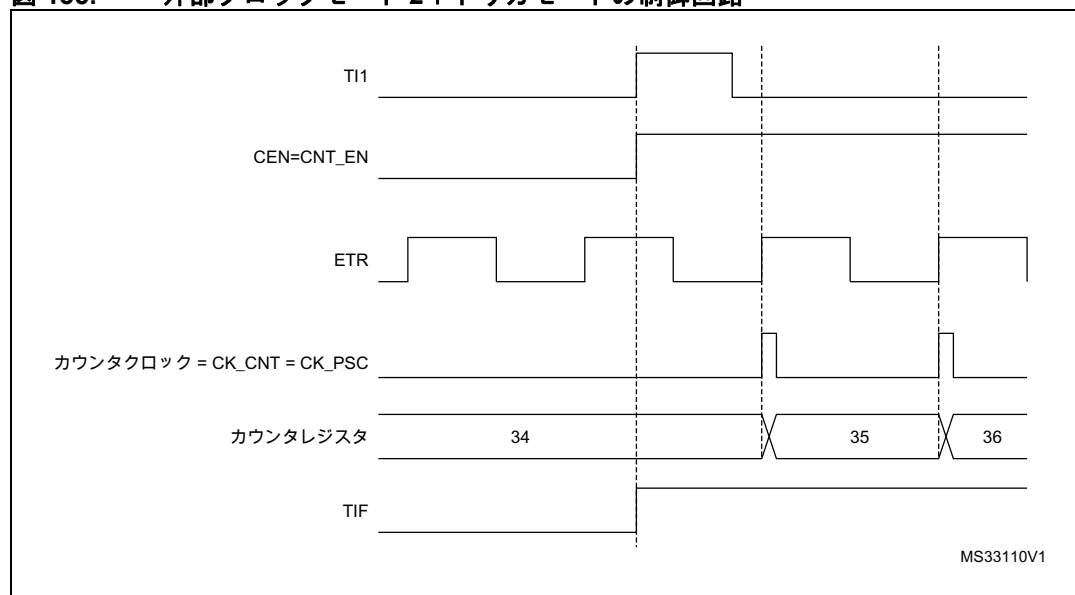
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタで次のようにプログラミングすることによって、外部トリガ入力回路を構成します。
 - ETF = 0000 : フィルタなし
 - ETPS=00 : プリスケール無効
 - ETP=0 : ETR の立ち上がりエッジを検出。ECE=1 で外部クロックモード 2 を有効にします。
2. TI1 の立ち上がりエッジを検出するように、チャンネル 1 を次のように構成します。
 - IC1F=0000 : フィルタなし。
 - キャプチャプリスケールはトリガには使用されないため、設定する必要はありません。
 - TIMx_CCMR1 レジスタの CC1S=01 で、入力キャプチャソースのみを選択します。
 - TIMx_CCER レジスタの CC1P=0 と CC1NP=0 で、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
3. TIMx_SMCR レジスタに SMS=110 を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタに TS=101 を書き込むことによって、入力ソースとして TI1 を選択します。

TI1 の立ち上がりエッジでカウンタが有効になり、TIF フラグがセットされます。カウンタは、ETR の立ち上がりエッジでカウントします。

ETR 信号の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、ETRP 入力の再同期回路によるものです。

図 135. 外部クロックモード 2+トリガモードの制御回路



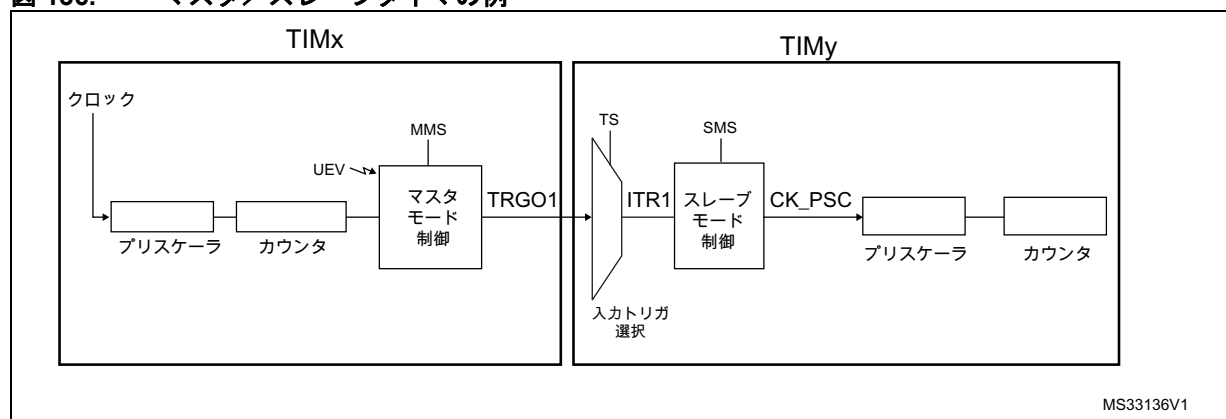
21.3.15 タイマの同期

タイマの同期や連携した動作のために、TIMx タイマを内部で相互リンクすることができます。マスタモードに設定されたタイマは、スレーブモードに設定された別のタイマのカウンタのリセット、開始、停止、またはクロック供給を行うことができます。

図 136: マスタ/スレーブタイマの例に、トリガ選択およびマスタモード選択ブロックの概要を示します。

タイマを別のタイマのプリスケアラとして使用する

図 136. マスタ/スレーブタイマの例



たとえば、タイマ x をタイマ y のプリスケアラとして動作するように設定できます。[図 136](#) を参照してください。このためには、次の操作を行います。

1. タイマ x をマスタモードで設定して、更新イベント UEV ごとに周期的トリガ信号を出力するようにします。TIMx_CR2 レジスタの MMS=010 を書き込んだ場合、更新イベントが生成されるたびに、TRGO1 で立ち上がりエッジが出力されます。
2. タイマ x の TRGO1 出力をタイマ y に接続するには、タイマ y を、ITR1 を内部トリガとして使用するスレーブモードに設定する必要があります。このためには、TIMy_SMCR レジスタの TS ビットを使用します (TS=000 を書き込みます)。
3. 次に、スレーブモードコントローラを外部クロックモード 1 にします (TIMy_SMCR レジスタの SMS=111 を書き込みます)。これによりタイマ y は、タイマ x の周期的なトリガ信号の立ち上がりエッジ (タイマ x カウンタのオーバーフローに対応) をクロックとして動作します。
4. 最後に、それぞれの CEN ビット (TIMx_CR1 レジスタ) をセットすることによって、両方のタイマを有効にする必要があります。

注： タイマ x のトリガ出力として OCx が選択された場合 (MMS=1xx)、その立ち上がりエッジがタイマ y カウンタのクロックとして使用されます。

タイマを使用して別のタイマを有効にする

この例では、タイマ x の出力比較 1 でタイマ y の有効/無効を制御します。接続については、[図 136](#) を参照してください。タイマ y は、タイマ x の OC1REF がハイのときにのみ、分周された内部クロックでカウントします。両方のカウンタクロック周波数は、CK_INT をプリスケアラで 3 分周したものです ($f_{CK_CNT} = f_{CK_INT}/3$)。

1. タイマ x をマスタモードに設定して、その出力比較 1 基準 (OC1REF) 信号をトリガ出力として送信します (TIMx_CR2 レジスタの MMS=100)。
2. タイマ x の OC1REF 波形を設定します (TIMx_CCMR1 レジスタ)。
3. タイマ y を、タイマ x から入力トリガを取得するように設定します (TIMy_SMCR レジスタの TS=000)。
4. タイマ y をゲートモードに設定します (TIMy_SMCR レジスタの SMS=101)。
5. CEN ビット (TIMy_CR1 レジスタ) に 1 を書き込むことによって、タイマ y を有効にします。
6. CEN ビット (TIMx_CR1 レジスタ) に 1 を書き込むことによって、タイマ x を開始します。

注： カウンタ 2 のクロックはカウンタ 1 と同期しないので、このモードはタイマ y カウンタのイネーブル信号のみに影響します。

図 137. タイマ x の OC1REF によるタイマ y のゲート操作

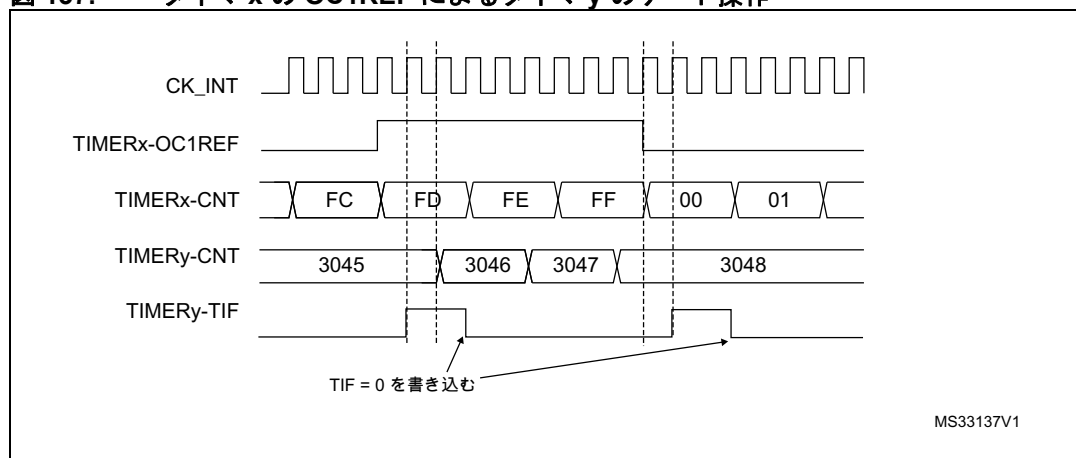
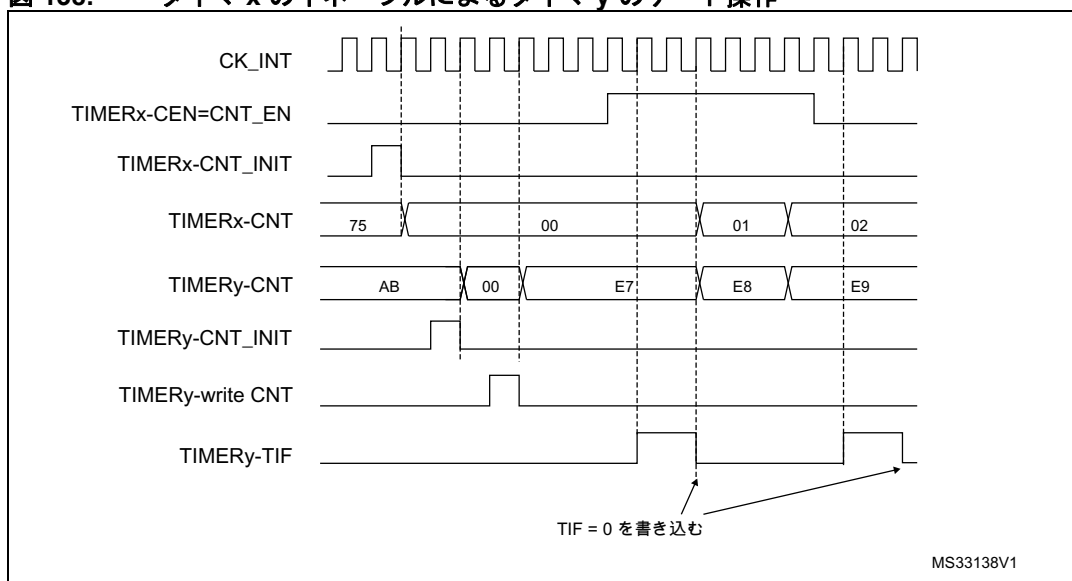


図 137 の例では、タイマ y のカウンタとプリスケアラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。タイマ x を開始する前に両方のタイマをリセットすることによって、特定の値から開始することが可能です。この場合、タイマカウンタに任意の値を書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで容易にタイマをリセットできます。

次の例では、タイマ x とタイマ y を同期させます。タイマ x はマスタであり、0 からカウントを開始します。タイマ y はスレーブであり、0xE7 から開始します。プリスケアラの分周比は両方のタイマで同じです。TIMy_CR1 レジスタの CEN ビットに“0”を書き込むことによってタイマ x を無効にすると、タイマ y は停止します。

1. タイマ x をマスタモードに設定して、その出力比較 1 基準 (OC1REF) 信号をトリガ出力として送信します (TIMx_CR2 レジスタの MMS=100)。
2. タイマ x の OC1REF 波形を設定します (TIMx_CCMR1 レジスタ)。
3. タイマ y を、タイマ x から入力トリガを取得するように設定します (TIMy_SMCR レジスタの TS=000)。
4. タイマ y をゲートモードに設定します (TIMy_SMCR レジスタの SMS=101)。
5. UG ビット (TIMx_EGR レジスタ) に 1 を書き込むことによって、タイマ x をリセットします。
6. UG ビット (TIMy_EGR レジスタ) に 1 を書き込むことによって、タイマ y をリセットします。
7. タイマ y のカウンタ (TIMy_CNT) に 0xE7 を書き込むことによって、タイマ y を 0xE7 に初期化します。
8. CEN ビット (TIMy_CR1 レジスタ) に 1 を書き込むことによって、タイマ y を有効にします。
9. CEN ビット (TIMx_CR1 レジスタ) に 1 を書き込むことによって、タイマ x を開始します。
10. CEN ビット (TIMx_CR1 レジスタ) に 0 を書き込むことによって、タイマ x を停止します。

図 138. タイマ x のイネーブルによるタイマ y のゲート操作



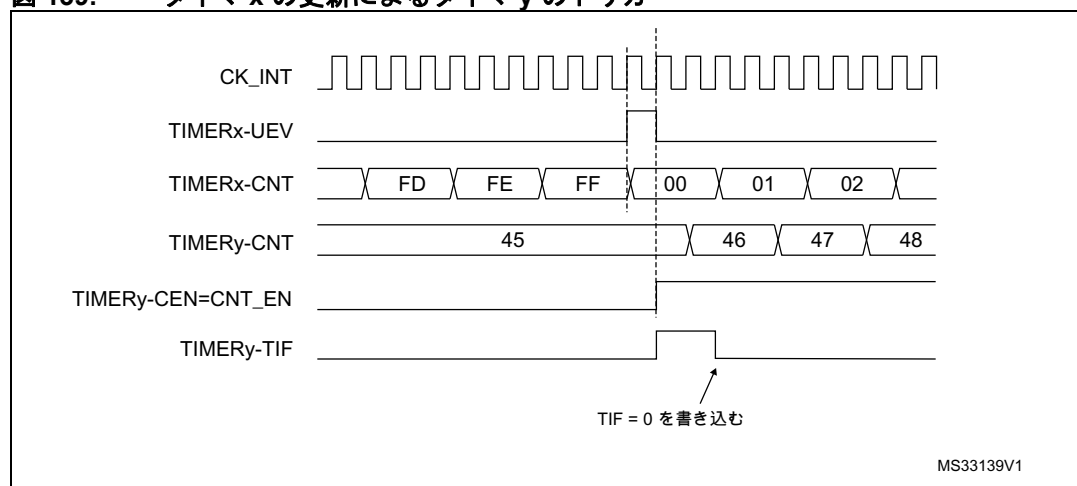
タイマを使用して別のタイマを開始する

この例では、タイマ x の更新イベントによってタイマ y の有効化を設定します。接続については、図 136 を参照してください。タイマ x によって更新イベントが生成されると、タイマ y は、分周された内部クロックで現在値 (ゼロである必要はありません) からカウントを開始します。タイマ y がトリガ信号を受信すると、その CEN ビットが自動的にセットされ、カウンタは TIM2_CR1 レジスタの

CEN ビットに 0 が書き込まれるまでカウントします。両方のカウンタクロック周波数は、CK_INT をプリスケアラで 3 分周したものです ($f_{CK_CNT} = f_{CK_INT}/3$)。

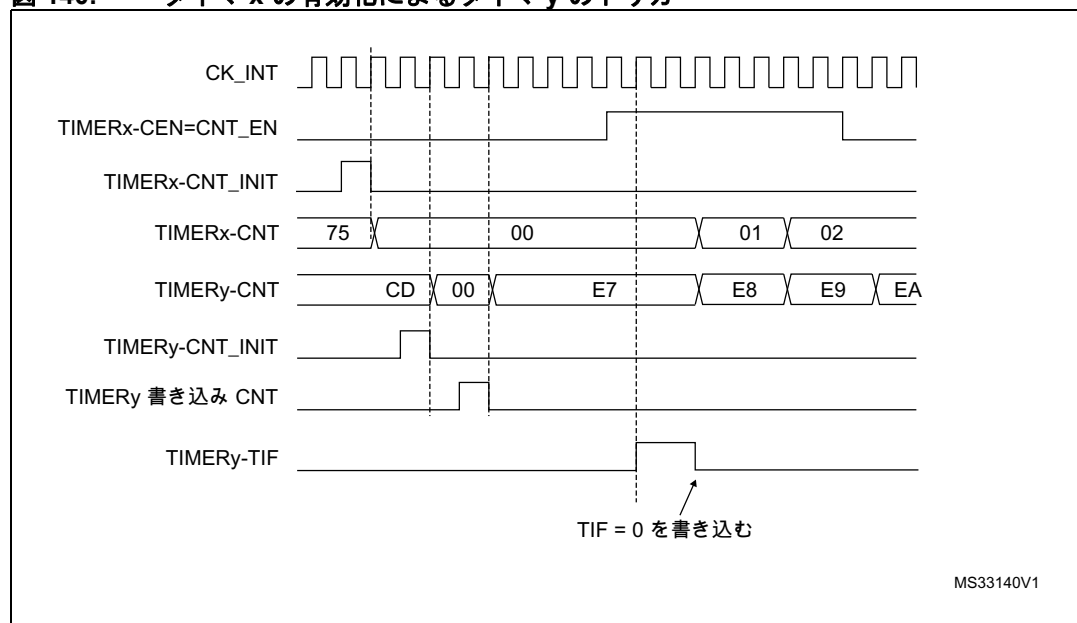
1. タイマ x をマスタモードに設定して、その更新イベント (UEV) をトリガ出力として送信します (TIMx_CR2 レジスタの MMS=010)。
2. タイマ x の周期を設定します (TIMx_ARR レジスタ)。
3. タイマ y を、タイマ x から入力トリガを受け取るように設定します (TIM2_SMCR レジスタの TS=000)。
4. タイマ y をトリガモードに設定します (TIM2_SMCR レジスタの SMS=110)。
5. CEN ビット (TIMx_CR1 レジスタ) に 1 を書き込むことによって、タイマ x を開始します。

図 139. タイマ x の更新によるタイマ y のトリガ



前の例と同じように、カウントを開始する前に両方のカウンタを初期化することができます。図 140 は 図 139 と同じ設定ですが、ゲートモードではなくトリガモードでの動作を示します (TIMy_SMCR レジスタの SMS=110)。

図 140. タイマ x の有効化によるタイマ y のトリガ



タイマを別のタイマのプリスケアラとして使用する

たとえば、タイマ x をタイマ y のプリスケアラとして動作するように設定できます。接続については、[図 136](#) を参照してください。このためには、次の操作を行います。

1. タイマ x をマスタモードに設定して、その更新イベント (UEV) をトリガ出力として送信し (TIMx_CR2 レジスタの MMS=010)、カウンタオーバーフローごとに周期的信号を出力します。
2. タイマ x の周期を設定します (TIMx_ARR レジスタ)。
3. タイマ y を、タイマ x から入力トリガを受け取るように設定します (TIM2_SMCR レジスタの TS=000)。
4. タイマ y を外部クロックモード1に設定します (TIM2_SMCR レジスタの SMS=111)。
5. CEN ビット (TIMy_CR1 レジスタ) に 1 を書き込むことによって、タイマ y を開始します。
6. CEN ビット (TIMx_CR1 レジスタ) に 1 を書き込むことによって、タイマ x を開始します。

外部トリガに対応して 2 つのタイマを同期して開始します。

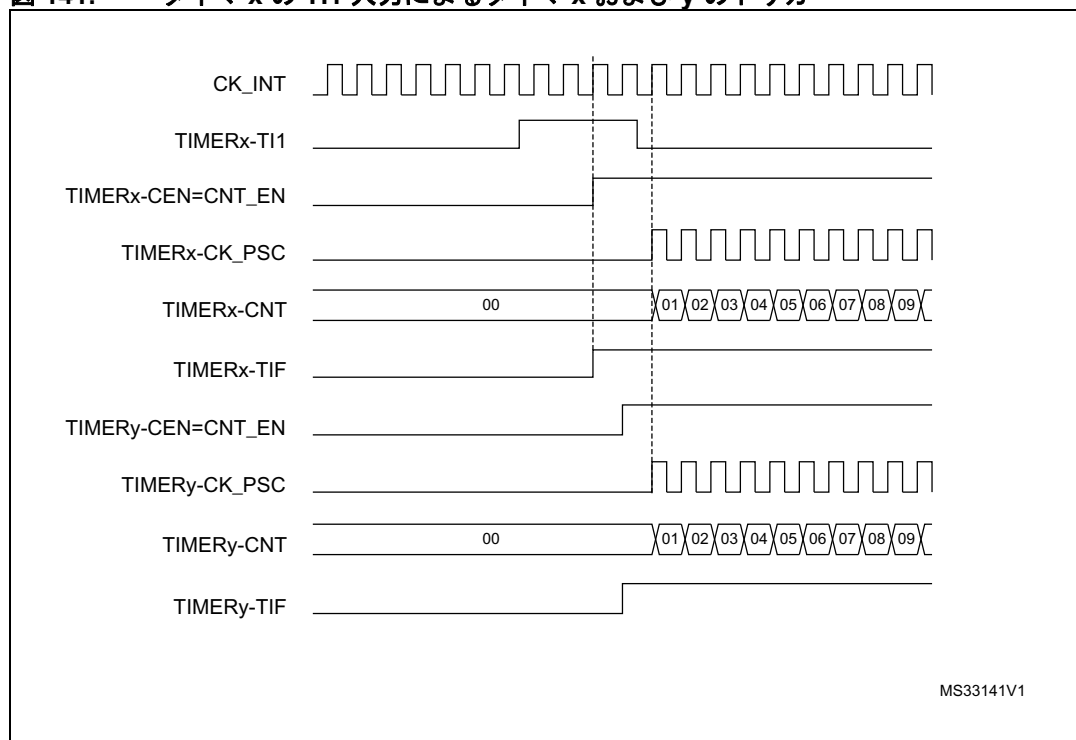
この例では、TI1 入力の立ち上がりエッジでタイマ x を有効にし、タイマ x が有効になると、タイマ y を有効にします。接続については、[図 136](#) を参照してください。カウンタの同時性を確保するため、タイマ x はマスタ/スレーブモードに設定する必要があります (TI1 に対してはスレーブ、タイマ y に対してはマスタ)。

1. タイマ x をマスタモードに設定して、その有効化をトリガ出力として送信します (TIMx_CR2 レジスタの MMS=001)。
2. タイマ x をスレーブモードに設定して、TI1 から入力トリガを受け取るようにします (TIMx_SMCR レジスタの TS=100)。
3. タイマ x をトリガモードに設定します (TIMx_SMCR レジスタの SMS=110)。
4. MSM=1 (TIMx_SMCR レジスタ) を書き込むことによって、タイマ x をマスタ/スレーブモードに設定します。
5. タイマ y を、タイマ x から入力トリガを受け取るように設定します (TIM2_SMCR レジスタの TS=000)。
6. タイマ y をトリガモードに設定します (TIM2_SMCR レジスタの SMS=110)。

TI1 (タイマ x) で立ち上がりエッジが発生すると、両方のカウンタが同時に内部クロックによるカウントを開始し、両方の TIF フラグがセットされます。

注： この例では、両方のタイマが開始前に初期化されます (それぞれの UG ビットをセットすることによって)。両方のカウンタは 0 から開始しますが、カウンタレジスタ (TIMx_CNT) に書き込むことによって、容易にオフセットを挿入できます。マスタ/スレーブモードでは、タイマ x の CNT_EN と CK_PSC の間に遅延が挿入されます。

図 141. タイマ x の TI1 入力によるタイマ x および y のトリガ



21.3.16 デバッグモード

マイクロコントローラがデバッグモードになると (Cortex[®]-M0+ コアは停止状態)、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、停止します。詳細については、[セクション 33.9.2: タイマ、ウォッチドッグ、および I2C のデバッグサポート](#) を参照してください。

21.4 TIM2 レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

32 ビットのペリフェラルレジスタには、ワード (32 ビット) 単位で書き込む必要があります。他のすべてのペリフェラルレジスタには、ハーフワード (16 ビット) またはワード (32 ビット) 単位で書き込む必要があります。読み出しアクセスは、バイト (8 ビット)、ハーフワード (16 ビット)、またはワード (32 ビット) 単位で行うことができます。

21.4.1 TIMx 制御レジスタ 1 (TIMx_CR1)

アドレスオフセット : 0x00

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|----------|----|------|-----|----|-----|-----|-----|------|-----|
| Res. | Res. | Res. | Res. | Res. | Res. | CKD[1:0] | | ARPE | CMS | | DIR | OPM | URS | UDIS | CEN |
| | | | | | | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9:8 **CKD** : クロック分周

このビットフィールドは、タイマクロック (CK_INT) 周波数と、デジタルフィルタ (ETR、Tlx) によって使用されるサンプリングクロックとの間の分周比を示します。

00 : $t_{DTS} = t_{CK_INT}$
01 : $t_{DTS} = 2 \times t_{CK_INT}$
10 : $t_{DTS} = 4 \times t_{CK_INT}$
11 : 予約済み

ビット 7 **ARPE** : 自動再ロードプリロードイネーブル

0 : TIMx_ARR レジスタはバッファされません。
1 : TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS** : センターアラインモード選択

00 : エッジアラインモードカウンタは、方向ビット (DIR) に応じて、カウントアップまたはカウントダウンします。
01 : センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル (TIMx_CCMRx レジスタの CCxS=00) の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。
10 : センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル (TIMx_CCMRx レジスタの CCxS=00) の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。
11 : センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル (TIMx_CCMRx レジスタの CCxS=00) の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注 : カウンタが有効 (CEN=1) なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR** : 方向

0 : カウンタはアップカウンタとして使用されます。
1 : カウンタはダウンカウンタとして使用されます。

注 : このビットは、タイマがセンターアラインモードまたはエンコーダモードに設定されているときには読み出し専用です。

ビット 3 **OPM** : ワンパルスモード

0 : カウンタは更新イベントで停止しません。
1 : カウンタは次の更新イベントでカウントを停止します (CEN ビットをクリア)。

ビット 2 URS : 更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアによってセット／クリアされます。

0 : 次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します（有効な場合）。これらのイベントは、次のとおりです。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1 : カウンタオーバーフロー／アンダーフローのみが更新割り込みまたは DMA リクエストを生成します（有効な場合）。

ビット 1 UDIS : 更新ディセーブル

このビットは、UEV イベント生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : UEV は有効です。更新イベント（UEV）は、次のいずれかのイベントによって生成されます。

- カウンタオーバーフロー／アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

バッファを持つレジスタにはプリロード値がロードされます。

1 : UEV は無効です。更新イベントは生成されず、シャドウレジスタ（ARR、PSC、CCR_x）は値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケラは再初期化されます。

ビット 0 CEN : カウンタイネーブル

0 : カウンタは無効です。

1 : カウンタは有効です。

注： 外部クロック、ゲートモード、およびエンコーダモードは、CEN ビットが事前にソフトウェアによってセットされている場合のみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

21.4.2 TIMx 制御レジスタ 2 (TIMx_CR2)

アドレスオフセット : 0x04

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|----------|-----|-----|------|------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI1S | MMS[2:0] | | | CCDS | Res. | Res. | Res. |
| | | | | | | | | r/w | r/w | r/w | r/w | r/w | | | |

ビット 15:8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 **TI1S** : TI1 選択

- 0 : TIMx_CH1 ピンが TI1 入力に接続されます。
- 1 : TIMx_CH1、CH2、および CH3 ピンが TI1 入力に接続されます (XOR 接続)。

ビット 6:4 **MMS** : マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます (TRGO)。組み合わせは、次のとおりです。

000 : **リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力 (TRGO) として使用されます。トリガ入力によってリセットが生成される場合 (スレーブモードコントローラがリセットモードに設定されているとき)、TRGO 信号は実際のリセットより遅延します。

001 : **イネーブル** - カウンタイネーブル信号 CNT_EN, がトリガ出力 (TRGO) として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和 (OR) によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ/スレーブモードが選択されている場合を除き、TRGO には遅延が存在します (TIMx_SMCR レジスタの MSM ビットの説明を参照してください)。

010 : **更新** - 更新イベントがトリガ出力 (TRGO) として使用されます。たとえば、マスタタイマをスレーブタイマのプリスケアラとして使用できます。

011 : **パルス比較** - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき (すでにハイであった場合も)、トリガ出力は正のパルスを送信します。 (TRGO)

100 : **比較** - OC1REF 信号がトリガ出力 (TRGO) として使用されます。

101 : **比較** - OC2REF 信号がトリガ出力 (TRGO) として使用されます。

110 : **比較** - OC3REF 信号がトリガ出力 (TRGO) として使用されます。

111 : **比較** - OC4REF 信号がトリガ出力 (TRGO) として使用されます。

ビット 3 **CCDS** : キャプチャ/比較 DMA 選択

- 0 : CCx DMA リクエストは、CCx イベントが発生すると送信されます。
- 1 : CCx DMA リクエストは、更新イベントが発生すると送信されます。

ビット 2:0 予約済みであり、リセット値のままにしておかなければなりません。

21.4.3 TIMx スレーブモード制御レジスタ (TIMx_SMCR)

アドレスオフセット : 0x08

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----|-----|-----------|----|----------|----|----|----|-----|---------|----|----|---|------|----------|----|
| ETP | ECE | ETPS[1:0] | | ETF[3:0] | | | | MSM | TS[2:0] | | | | Res. | SMS[2:0] | |
| RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | | | RW | RW |

ビット 15 **ETP** : 外部トリガ極性

このビットは、ETR と $\overline{\text{ETR}}$ のいずれがトリガ動作に使用されるかを選択します。
0 : ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブになります。
1 : ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブになります。

ビット 14 **ECE** : 外部クロックイネーブル

このビットは、外部クロックモード 2 を有効にします。
0 : 外部クロックモード 2 は無効です。
1 : 外部クロックモード 2 は有効です。カウンタは、ETRF 信号のアクティブエッジによってクロック供給されます。
1 : ECE ビットをセットすることは、TRGI が ETRF に接続された状態で外部クロックモード 1 を選択することと同じ効果があります (SMS=111、TS=111)。
2 : 外部クロックモード 2 と次のスレーブモード、すなわち、リセットモード、ゲートモード、またはトリガモードを同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません (TS ビットが 111 でないことが必要)。
3 : 外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力 は ETRF です。

ビット 13:12 **ETPS** : 外部トリガプリスケアラ

外部トリガ信号 ETRP の周波数は、最大でも CK_INT 周波数の 1/4 でなければなりません。プリスケアラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。
00 : プリスケアラオフ
01 : ETRP 周波数は 2 分周されます。
10 : ETRP 周波数は 4 分周されます。
11 : ETRP 周波数は 8 分周されます。

ビット 11:8 **ETF[3:0]** : 外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプルする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。
0000 : フィルタなし、サンプリングは f_{DTS} で行われます。
0001 : $f_{\text{SAMPLING}} = f_{\text{CK_INT}}$, N=2
0010 : $f_{\text{SAMPLING}} = f_{\text{CK_INT}}$, N=4
0011 : $f_{\text{SAMPLING}} = f_{\text{CK_INT}}$, N=8
0100 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/2$, N=6
0101 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/2$, N=8
0110 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/4$, N=6
0111 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/4$, N=8
1000 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/8$, N=6
1001 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/8$, N=8
1010 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/16$, N=5
1011 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/16$, N=6
1100 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/16$, N=8
1101 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/32$, N=5
1110 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/32$, N=6
1111 : $f_{\text{SAMPLING}} = f_{\text{DTS}}/32$, N=8



ビット 7 **MSM** : マスタ/スレーブモード

0 : 影響なし。

1 : トリガ入力（TRGI）に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期（TRGO を通じて）を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期したい場合に役立ちます。

ビット 6:4 **TS** : トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

000 : 内部トリガ 0（ITR0）

001 : 内部トリガ 1（ITR1）

010 : 予約済み。

011 : 予約済み。

100 : TI1 エッジ検出回路（TI1F_ED）

101 : フィルタタイマ入力 1（TI1FP1）

110 : フィルタタイマ入力 2（TI2FP2）

111 : 外部トリガ入力（ETRF）

各タイマでの ITRx の詳細については、[表 80:TIM2 内部トリガ接続\(454 ページ\)](#) を参照してください。

注： **設定変更時の誤ったエッジ検出を避けるために、これらのビットは、使用されていないとき（SMS=000 のときなど）にのみ変更しなければなりません。**

ビット 3 予約済みであり、1 のままにしておかなければなりません。

ビット 2:0 **SMS** : スレーブモード選択

外部信号が選択されると、トリガ信号（TRGI）のアクティブエッジが外部入力で選択された極性にリンクされます（入力制御レジスタおよび制御レジスタの説明を参照してください）。

000 : スレーブモードは無効です。CEN =“1”の場合、プリスケアラは内部クロックによって直接クロック供給されます。

001 : エンコーダモード 1 - カウンタは、TI1FP1 のレベルに応じて、TI2FP2 のエッジでカウントアップ/ダウンします。

010 : エンコーダモード 2 - カウンタは、TI2FP2 のレベルに応じて、TI1FP1 のエッジでカウントアップ/ダウンします。

011 : エンコーダモード 3 - カウンタは、他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでカウントアップ/ダウンします。

100 : リセットモード - 選択されたトリガ入力（TRGI）の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。

101 : ゲートモード - カウンタクロックは、トリガ入力（TRGI）がハイのときに有効になります。トリガがローになると、カウンタは停止します（リセットはされません）。カウンタの開始と停止の両方が制御されます。

110 : トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します（リセットはされません）。カウンタの開始のみが制御されます。

111 : 外部クロックモード 1 - 選択されたトリガ（TRGI）の立ち上がりエッジがカウンタのクロックとして供給されます。

注： **トリガ入力として TI1F_ED が選択されている場合（TS=100）、ゲートモードを使用することはできません。TI1F_ED は TI1F の変化ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。**

表 80. TIM2 内部トリガ接続

| スレーブ TIM | ITR0 (TS = 000) | ITR1 (TS = 001) |
|----------|-----------------|-----------------|
| TIM2 | TIM21 | TIM22 |

21.4.4 TIMx DMA／割り込み有効レジスタ (TIMx_DIER)

アドレスオフセット : 0x0C

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|-----|------|-------|-------|-------|-------|-----|------|-----|------|-------|-------|-------|-------|-----|
| Res. | TDE | Res. | CC4DE | CC3DE | CC2DE | CC1DE | UDE | Res. | TIE | Res. | CC4IE | CC3IE | CC2IE | CC1IE | UIE |
| | rw | | rw | rw | rw | rw | rw | | rw | | rw | rw | rw | rw | rw |

ビット 15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **TDE** : トリガ DMA リクエストイネーブル
0 : トリガ DMA リクエストは無効です。
1 : トリガ DMA リクエストは有効です。

ビット 13 予約済み。常に 0 として読み出されます。

ビット 12 **CC4DE** : キャプチャ／比較 4 DMA リクエストイネーブル
0 : CC4 DMA リクエストは無効です。
1 : CC4 DMA リクエストは有効です。

ビット 11 **CC3DE** : キャプチャ／比較 3 DMA リクエストイネーブル
0 : CC3 DMA リクエストは無効です。
1 : CC3 DMA リクエストは有効です。

ビット 10 **CC2DE** : キャプチャ／比較 2 DMA リクエストイネーブル
0 : CC2 DMA リクエストは無効です。
1 : CC2 DMA リクエストは有効です。

ビット 9 **CC1DE** : キャプチャ／比較 1 DMA リクエストイネーブル
0 : CC1 DMA リクエストは無効です。
1 : CC1 DMA リクエストは有効です。

ビット 8 **UDE** : 更新 DMA リクエストイネーブル
0 : 更新 DMA リクエストは無効です。
1 : 更新 DMA リクエストは有効です。

ビット 7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **TIE** : トリガ割り込みイネーブル
0 : トリガ割り込みは無効です。
1 : トリガ割り込みは有効です。

ビット 5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **CC4IE** : キャプチャ／比較 4 割り込みイネーブル
0 : CC4 割り込みは無効です。
1 : CC4 割り込みは有効です。

ビット 3 **CC3IE** : キャプチャ／比較 3 割り込みイネーブル
0 : CC3 割り込みは無効です。
1 : CC3 割り込みは有効です。

- ビット 2 **CC2IE** : キャプチャ／比較 2 割り込みイネーブル
0 : CC2 割り込みは無効です。
1 : CC2 割り込みは有効です。
- ビット 1 **CC1IE** : キャプチャ／比較 1 割り込みイネーブル
0 : CC1 割り込みは無効です。
1 : CC1 割り込みは有効です。
- ビット 0 **UIE** : 更新割り込みイネーブル
0 : 更新割り込みは無効です。
1 : 更新割り込みは有効です。

21.4.5 TIMx ステータスレジスタ（TIMx_SR）

アドレスオフセット : 0x10
リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|-------|-------|-------|-------|------|------|-------|------|-------|-------|-------|-------|-------|
| Res. | Res. | Res. | CC4OF | CC3OF | CC2OF | CC1OF | Res. | Res. | TIF | Res. | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | | | rc_w0 | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 |

- ビット 15:13 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 12 **CC4OF** : キャプチャ／比較 4 オーバーキャプチャフラグ
CC1OF の説明を参照してください。
- ビット 11 **CC3OF** : キャプチャ／比較 3 オーバーキャプチャフラグ
CC1OF の説明を参照してください。
- ビット 10 **CC2OF** : キャプチャ／比較 2 オーバーキャプチャフラグ
CC1OF の説明を参照してください。
- ビット 9 **CC1OF** : キャプチャ／比較 1 オーバーキャプチャフラグ
このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。ソフトウェアで“0”を書き込むことによってクリアされます。
0 : オーバキャプチャは検出されていません。
1: CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。
- ビット 8:7 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 6 **TIF** : トリガ割り込みフラグ
このフラグは、トリガイイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。
0 : トリガイイベントは発生していません。
1 : トリガ割り込みが保留中です。
- ビット 5 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 4 **CC4IF** : キャプチャ／比較 4 割り込みフラグ
CC1IF の説明を参照してください。
- ビット 3 **CC3IF** : キャプチャ／比較 3 割り込みフラグ
CC1IF の説明を参照してください。



ビット 2 **CC2IF** : キャプチャ/比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 **CC1IF** : キャプチャ/比較 1 割り込みフラグ

チャンネル CC1 が出力として設定されている場合 :

このフラグは、カウンタが比較値と一致したときにハードウェアによってセットされます (センターアラインモードでは、例外もあります。TIMx_CR1 レジスタの CMS ビットの説明を参照してください)。ソフトウェアによってクリアされます。

0 : 一致していません。

1 : カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時 (アップカウントおよびアップ/ダウンカウントモードの場合)、またはアンダーフロー時 (ダウンカウントモードの場合) に CC1IF ビットはハイになります。

チャンネル CC1 が入力として設定されている場合 :

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0 : 入力キャプチャは発生していません。

1 : カウンタの値が TIMx_CCR1 レジスタにキャプチャされました (IC1 で、選択された極性に一致するエッジが検出されました)。

ビット 0 **UIF** : 更新割り込みフラグ

- このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- オーバーフローまたはアンダーフロー時と、TIMx_CR1 レジスタの UDIS=0 の場合。
- TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。

TIMx_CR1 レジスタの URS=0 かつ UDIS=0 であり、トリガイイベントによって CNT が再初期化されたとき (同期制御レジスタの説明を参照)。

21.4.6 TIMx イベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|----|------|------|------|------|------|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TG | Res. | CC4G | CC3G | CC2G | CC1G | UG |
| | | | | | | | | | w | | w | w | w | w | w |

ビット 15:7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **TG**：トリガ生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込みまたは DMA 転送が発生します。

ビット 5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **CC4G**：キャプチャ／比較 4 生成

CC1G の説明を参照してください。

ビット 3 **CC3G**：キャプチャ／比較 3 生成

CC1G の説明を参照してください。

ビット 2 **CC2G**：キャプチャ／比較 2 生成

CC1G の説明を参照してください。

ビット 1 **CC1G**：キャプチャ／比較 1 生成

このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：チャンネル 1 でキャプチャ／比較イベントが生成されます。

チャンネル CC1 が出力として設定されている場合：

CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。

チャンネル CC1 が入力として設定されている場合：

カウンタの現在値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みまたは DMA リクエストが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

ビット 0 **UG**：更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0：影響なし。

1：カウンタを再初期化し、レジスタの更新を生成します。プリスケアラカウンタもクリアされます（プリスケアラ比は変化しません）。センターラインモードが選択されている場合、または、DIR=0（カウントアップ）の場合、カウンタはクリアされます。そうでない場合、DIR=1（カウントダウン）であれば、自動再ロード値（TIMx_ARR）をとりま

21.4.7 TIMx キャプチャ／比較モードレジスタ 1 (TIMx_CCMR1)

アドレスオフセット : 0x18

リセット値 : 0x0000

チャンネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャンネルの方向は対応する CCxS ビットで定まります。このレジスタの他のビットはすべて、入力モードと出力モードで異なる機能を持ちます。特定のビットについて、OCxx は、チャンネルが出力設定のときの機能を示し、ICxx は、チャンネルが入力設定のときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意する必要があります。

| | | | | | | | | | | | | | | | |
|-----------|-----------|-----|-----|-------------|-------|-----------|-----|-----------|-----------|-----|-------------|-------|-------|-----------|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OC2CE | OC2M[2:0] | | | OC2PE | OC2FE | CC2S[1:0] | | OC1CE | OC1M[2:0] | | | OC1PE | OC1FE | CC1S[1:0] | |
| IC2F[3:0] | | | | IC2PSC[1:0] | | | | IC1F[3:0] | | | IC1PSC[1:0] | | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

出力比較モード

ビット 15 **OC2CE** : 出力比較 2 クリアイネーブル

ビット 14:12 **OC2M[2:0]** : 出力比較 2 モード

ビット 11 **OC2PE** : 出力比較 2 プリロードイネーブル

ビット 10 **OC2FE** : 出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]** : キャプチャ／比較 2 選択

このビットフィールドは、チャンネルの方向（入力／出力）と、使用される入力を定義します。

00 : CC2 チャンネルは出力として設定されます。

01 : CC2 チャンネルは入力として設定され、IC2 は TI2 に配置されます。

10 : CC2 チャンネルは入力として設定され、IC2 は TI1 に配置されます。

11 : CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC2S** ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC2E=0) のときにのみ書き込み可能です。

ビット 7 **OC1CE** : 出力比較 1 クリアイネーブル

OC1CE : 出力比較 1 クリアイネーブル

0 : OC1Ref は ETRF 入力の影響を受けません。

1 : OC1Ref は ETRF 入力のハイレベルが検出されるとクリアされます。

ビット 6:4 OC1M : 出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは CC1P および CC1NP ビットに依存します。

000 : 停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT との間の比較結果は出力に影響しません (このモードはタイミングベースを生成するために使用されます)。

001 : 一致時にチャンネル 1 をアクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ/比較レジスタ 1 (TIMx_CCR1) と一致したときに、強制的にハイになります。

010 : 一致時にチャンネル 1 を非アクティブレベルに設定します。OC1REF 信号は、カウンタ TIMx_CNT がキャプチャ/比較レジスタ 1 (TIMx_CCR1) と一致したときに、強制的にローになります。

011 : 反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

100 : 強制非アクティブレベル - OC1REF は強制的にローになります。

101 : 強制アクティブレベル - OC1REF は強制的にハイになります。

110 : PWM モード 1 - カウントアップ時、チャンネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。カウントダウン時、チャンネル 1 は、TIMx_CNT > TIMx_CCR1 の場合は非アクティブ (OC1REF="0") に、そうでない場合はアクティブ (OC1REF="1") になります。

111 : PWM モード 2 - カウントアップ時、チャンネル 1 は、TIMx_CNT < TIMx_CCR1 の場合は非アクティブに、そうでない場合はアクティブになります。カウントダウン時、チャンネル 1 は、TIMx_CNT > TIMx_CCR1 の場合はアクティブに、そうでない場合は非アクティブになります。

注 : 1: これらのビットは、LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、CC1S=00 (チャンネルは出力に設定) のときには、変更できません。

2: PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときのみ、OCREF のレベルが変化します。

ビット 3 OC1PE : 出力比較 1 プリロードイネーブル

0 : TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1 : TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注 : 1: これらのビットは、LOCK レベル 3 がプログラムされていて (TIMx_BDTR レジスタの LOCK ビット)、CC1S=00 (チャンネルは出力に設定) のときには、変更できません。

2: PWM モードは、ワンパルスモード (TIMx_CR1 レジスタの OPM ビットがセットされている) のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 OC1FE : 出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガがイベントの効果を加速するために使用されます。

0 : CC1 の動作は、トリガがオンのときでも、通常、カウンタと CCR1 の値に依存します。トリガ入力のエッジ発生から CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1 : トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このような場合、OC は、比較結果に関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OCFE は、チャンネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 CC1S : キャプチャ/比較 1 選択

このビットフィールドは、チャンネルの方向 (入力/出力) と、使用される入力を定義します。

00 : CC1 チャンネルは出力として設定されます。

01 : CC1 チャンネルは入力として設定され、IC1 は TI1 に配置されます。

10 : CC1 チャンネルは入力として設定され、IC1 は TI2 に配置されます。

11 : CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : CC1S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC1E=0) のときのみ書き込み可能です。

入力キャプチャモード

ビット 15:12 **IC2F** : 入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]** : 入力キャプチャ 2 プリスケアラ

ビット 9:8 **CC2S** : キャプチャ/比較 2 選択

このビットフィールドは、チャンネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC2 チャンネルは出力として設定されます。

01 : CC2 チャンネルは入力として設定され、IC2 は TI2 に配置されます。

10 : CC2 チャンネルは入力として設定され、IC2 は TI1 に配置されます。

11 : CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC2S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC2E=0) のときにのみ書き込み可能です。**

ビット 7:4 **IC1F** : 入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な変化をもたらすには N 個のイベント発生が必要です。

0000 : $f_{DTS}1000 : f_{SAMPLING}=f_{DTS}/8, N=6$ ではフィルタサンプリングは行われません。

0001 : $f_{SAMPLING}=f_{CK_INT}, N=2$ 1001 : $f_{SAMPLING}=f_{DTS}/8, N=8$

0010 : $f_{SAMPLING}=f_{CK_INT}, N=4$ 1010 : $f_{SAMPLING}=f_{DTS}/16, N=5$

0011 : $f_{SAMPLING}=f_{CK_INT}, N=8$ 1011 : $f_{SAMPLING}=f_{DTS}/16, N=6$

0100 : $f_{SAMPLING}=f_{DTS}/2, N=6$ 1100 : $f_{SAMPLING}=f_{DTS}/16, N=8$

0101 : $f_{SAMPLING}=f_{DTS}/2, N=8$ 1101 : $f_{SAMPLING}=f_{DTS}/32, N=5$

0110 : $f_{SAMPLING}=f_{DTS}/4, N=6$ 1110 : $f_{SAMPLING}=f_{DTS}/32, N=6$

0111 : $f_{SAMPLING}=f_{DTS}/4, N=8$ 1111 : $f_{SAMPLING}=f_{DTS}/32, N=8$

ビット 3:2 **IC1PSC** : 入力キャプチャ 1 プリスケアラ

このビットフィールドは、CC1 入力 (IC1) に作用するプリスケアラの分周比を定義します。

プリスケアラは、CC1E=0 (TIMx_CCER レジスタ) になるとリセットされます。

00 : プリスケアラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01 : キャプチャは、2 イベントごとに行われます。

10 : キャプチャは、4 イベントごとに行われます。

11 : キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S** : キャプチャ/比較 1 選択

このビットフィールドは、チャンネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC1 チャンネルは出力として設定されます。

01 : CC1 チャンネルは入力として設定され、IC1 は TI1 に配置されます。

10 : CC1 チャンネルは入力として設定され、IC1 は TI2 に配置されます。

11 : CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC1S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC1E=0) のときにのみ書き込み可能です。**

21.4.8 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCMR2）

アドレスオフセット：0x1C

リセット値：0x0000

上記の CCMR1 レジスタの説明を参照してください。

| | | | | | | | | | | | | | | | |
|-----------|-----------|-----|-----|-------------|-------|-----------|-----|-----------|-----------|-----|-------------|-------|-------|-----------|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OC4CE | OC4M[2:0] | | | OC4PE | OC4FE | CC4S[1:0] | | OC3CE | OC3M[2:0] | | | OC3PE | OC3FE | CC3S[1:0] | |
| IC4F[3:0] | | | | IC4PSC[1:0] | | | | IC3F[3:0] | | | IC3PSC[1:0] | | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

出力比較モード

ビット 15 OC4CE：出力比較 4 クリアイネーブル

ビット 14:12 OC4M：出力比較 4 モード

ビット 11 OC4PE：出力比較 4 プリロードイネーブル

ビット 10 OC4FE：出力比較 4 高速イネーブル

ビット 9:8 CC4S：キャプチャ／比較 4 選択

このビットフィールドは、チャンネルの方向（入力／出力）と、使用される入力を定義します。
00：CC4 チャンネルは出力として設定されます。
01：CC4 チャンネルは入力として設定され、IC4 は TI4 に配置されます。
10：CC4 チャンネルは入力として設定され、IC4 は TI3 に配置されます。
11：CC4 チャンネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力を選択されている場合のみ機能します。

注： CC4S ビットは、チャンネルがオフ（TIMx_CCER レジスタの CC4E=0）のときにのみ書き込み可能です。

ビット 7 OC3CE：出力比較 3 クリアイネーブル

ビット 6:4 OC3M：出力比較 3 モード

ビット 3 OC3PE：出力比較 3 プリロードイネーブル

ビット 2 OC3FE：出力比較 3 高速イネーブル

ビット 1:0 CC3S：キャプチャ／比較 3 選択

このビットフィールドは、チャンネルの方向（入力／出力）と、使用される入力を定義します。
00：CC3 チャンネルは出力として設定されます。
01：CC3 チャンネルは入力として設定され、IC3 は TI3 に配置されます。
10：CC3 チャンネルは入力として設定され、IC3 は TI4 に配置されます。
11：CC3 チャンネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット（TIMx_SMCR レジスタ）で内部トリガ入力を選択されている場合のみ機能します。

注： CC3S ビットは、チャンネルがオフ（TIMx_CCER レジスタの CC3E=0）のときにのみ書き込み可能です。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

入力キャプチャモード

ビット 15:12 **IC4F** : 入力キャプチャ 4 フィルタ

ビット 11:10 **IC4PSC** : 入力キャプチャ 4 プリスケアラ

ビット 9:8 **CC4S** : キャプチャ/比較 4 選択

このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC4 チャネルは出力として設定されます。

01 : CC4 チャネルは入力として設定され、IC4 は TI4 に配置されます。

10 : CC4 チャネルは入力として設定され、IC4 は TI3 に配置されます。

11 : CC4 チャネルは入力として設定され、IC4 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC4S ビットは、チャネルがオフ (TIMx_CCER レジスタの CC4E=0) のときにのみ書き込み可能です。**

ビット 7:4 **IC3F** : 入力キャプチャ 3 フィルタ

ビット 3:2 **IC3PSC** : 入力キャプチャ 3 プリスケアラ

ビット 1:0 **CC3S** : キャプチャ/比較 3 選択

このビットフィールドは、チャネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC3 チャネルは出力として設定されます。

01 : CC3 チャネルは入力として設定され、IC3 は TI3 に配置されます。

10 : CC3 チャネルは入力として設定され、IC3 は TI4 に配置されます。

11 : CC3 チャネルは入力として設定され、IC3 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC3S ビットは、チャネルがオフ (TIMx_CCER レジスタの CC3E=0) のときにのみ書き込み可能です。**

21.4.9 TIMx キャプチャ/比較有効レジスタ (TIMx_CCER)

アドレスオフセット : 0x20

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|------|------|------|-------|------|------|------|-------|------|------|------|-------|------|------|------|
| CC4NP | Res. | CC4P | CC4E | CC3NP | Res. | CC3P | CC3E | CC2NP | Res. | CC2P | CC2E | CC1NP | Res. | CC1P | CC1E |
| r/w | | r/w | r/w | r/w | | r/w | r/w | r/w | | r/w | r/w | r/w | | r/w | r/w |

ビット 15 **CC4NP** : キャプチャ/比較 4 出力極性

CC1NP の説明を参照してください。

ビット 14 予約済みであり、リセット値のままにしておかなければなりません。

ビット 13 **CC4P** : キャプチャ/比較 4 出力極性

CC1P の説明を参照してください。

ビット 12 **CC4E** : キャプチャ/比較 4 出力イネーブル。

CC1E の説明を参照してください。

ビット 11 **CC3NP** : キャプチャ/比較 3 出力極性

CC1NP の説明を参照してください。

ビット 10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **CC3P** : キャプチャ/比較 3 出力極性

CC1P の説明を参照してください。

ビット 8 **CC3E** : キャプチャ/比較 3 出力イネーブル。

CC1E の説明を参照してください。



- ビット 7 **CC2NP** : キャプチャ／比較 2 出力極性
CC1NP の説明を参照してください。
- ビット 6 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 5 **CC2P** : キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。
- ビット 4 **CC2E** : キャプチャ／比較 2 出力イネーブル。
CC1E の説明を参照してください。
- ビット 3 **CC1NP** : キャプチャ／比較 1 出力極性
CC1 チャンネルが出力として設定されている場合 :
この場合、CC1NP はクリアされたままでなければなりません。
CC1 チャンネルが入力として設定されている場合 :
このビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます (CC1P の説明を参照してください)。
- ビット 2 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 1 **CC1P** : キャプチャ／比較 1 出力極性
CC1 チャンネルが出力として設定されている場合 :
0 : OC1 はアクティブハイです。
1 : OC1 はアクティブローです。
CC1 チャンネルが入力として設定されている場合 :
CC1NP/CC1P ビットは、トリガまたはキャプチャ操作の TI1FP1 および TI2FP1 の極性を選択します。
00 : 非反転／立ち上がりエッジ
回路は TIxFP1 の立ち上がりエッジに反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されません (ゲートモードでのトリガ、エンコーダモード)。
01 : 反転／立ち下がりエッジ
回路は TIxFP1 の立ち下がりエッジに反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されます (ゲートモードでのトリガ、エンコーダモード)。
10 : 予約済み。この設定は使用しないでください。
11 : 非反転／両エッジ
回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されません (ゲートモードでのトリガ) 。この設定をエンコーダモードに使用することはできません。
- ビット 0 **CC1E** : キャプチャ／比較 1 出力イネーブル。
CC1 チャンネルが出力として設定されている場合 :
0 : オフ - OC1 はアクティブではありません。
1 : オン - OC1 信号は、対応する出力ピンに出力されます。
CC1 チャンネルが入力として設定されている場合 :
このビットによって、カウンタ値のキャプチャ／比較レジスタ 1 (TIMx_CCR1) へのキャプチャが実際に行われるかどうかが決まります。
0 : キャプチャは無効です。
1 : キャプチャは有効です。

表 81. 標準 OCx チャンネルの出力制御ビット

| CCxE ビット | OCx 出力状態 |
|----------|--------------------------|
| 0 | 出力無効 (OCx=0、OCx_EN=0) |
| 1 | OCx=OCxREF + 極性、OCx_EN=1 |

注：標準 OCx チャンネルに接続されている外部入出力ピンの状態は、OCx チャンネルの状態と、GPIO レジスタに依存します。

21.4.10 TIMx カウンタ (TIMx_CNT)

アドレスオフセット：0x24

リセット値：0x0000

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

ビット 15:0 CNT[15:0]：カウンタ値下位ビット

21.4.11 TIMx プリスケアラ (TIMx_PSC)

アドレスオフセット：0x28

リセット値：0x0000

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

ビット 15:0 PSC[15:0]：プリスケアラ値

カウンタクロック周波数 CK_CNT は $f_{CK_PSC} / (PSC[15:0] + 1)$ に等しいです。

PSC は、更新イベントごとにアクティブプリスケアラレジスタにロードされる値を含みます。

21.4.12 TIMx 自動再ロードレジスタ (TIMx_ARR)

アドレスオフセット：0x2C

リセット値：0x0000

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

ビット 15:0 ARR[15:0]:自動再ロード値下位ビット

ARR は、実際の自動再ロードレジスタにロードされる値です。

ARP の更新と動作の詳細については、[セクション 21.3.1：タイムベースユニット \(411 ページ\)](#) を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。

21.4.13 TIMx キャプチャ／比較モードレジスタ 1（TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR1[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15:0 CCR1[15:0]：キャプチャ／比較 1 値下位ビット

チャンネル CC1 が出力として設定されている場合：
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。

チャンネル CC1 が入力として設定されている場合：
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

21.4.14 TIMx キャプチャ／比較モードレジスタ 2（TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR2[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15:0 CCR2[15:0]：キャプチャ／比較 2 値下位ビット

CC2 チャンネルが出力として設定されている場合：
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC2 出力に送信される値を保持しています。

CC2 チャンネルが入力として設定されている場合：
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

21.4.15 TIMx キャプチャ／比較モードレジスタ 3 (TIMx_CCR3)

アドレスオフセット : 0x3C

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR3[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15:0 **CCR3[15:0]** : キャプチャ／比較値下位ビット

CC3 チャンネルが出力として設定されている場合 :

CCR3 は、実際のキャプチャ／比較 3 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR3 レジスタの OC3PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 3 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC3 出力に送信される値を含みます。

チャンネル CC3 が入力として設定されている場合 :

CCR3 は、最後の入力キャプチャ 3 イベント（IC3）によって転送されたカウンタ値です。

21.4.16 TIMx キャプチャ／比較モードレジスタ 4 (TIMx_CCR4)

アドレスオフセット : 0x40

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR4[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15:0 **CCR4[15:0]** : キャプチャ／比較値下位ビット

1. **CC4 チャンネルが出力として設定されている場合（CC4Sビット）:**

CCR4 は、実際のキャプチャ／比較 4 レジスタにロードされる値（プリロード値）です。

TIMx_CCMR4 レジスタの OC4PE ビットでプリロード機能が選択されていない場合、この値は不変にロードされます。そうでない場合、プリロード値は、更新イベントが発生したときに、アクティブキャプチャ／比較 4 レジスタにコピーされます。

アクティブキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較され、OC4 出力に送信される値を含みます。

2. **CC4 チャンネルが入力として設定されている場合（TIMx_CCMR4 レジスタの CC4S ビット）:**

CCR4 は、最後の入力キャプチャ 4 イベント（IC4）によって転送されたカウンタ値です。

21.4.17 TIMx DMA 制御レジスタ（TIMx_DCR）

アドレスオフセット：0x48

リセット値：0x0000

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

ビット 15:13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12:8 **DBL[4:0]**：DMA バースト長

この 5 ビットのベクタは、DMA 転送回数（タイマは、TIMx_DMAR アドレスに対して読み出したまたは書き込みアクセスが行われるときにバースト転送を認識します）を指定します。

00000：1 回転送
00001：2 回転送、
00010：3 回転送、
...
10001：18 回転送。

ビット 7:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4:0 **DBA[4:0]**：DMA ベースアドレス

この 5 ビットのベクタは、DMA 転送のベースアドレスを指定します（TIMx_DMAR アドレスを通じて読み出し／書き込みアクセスが行われるとき）。DBA は、TIMx_CR1 レジスタのアドレスから始まるオフセットとして定義されます。

例：
00000：TIMx_CR1
00001：TIMx_CR2
00010：TIMx_SMCR
...

例：次の転送を考えます：DBL = 7 回転送 かつ DBA = TIMx_CR1。この場合、転送は、TIMx_CR1 アドレスから始めて、7 つのレジスタに対して行われます。

21.4.18 完全転送の TIMx DMA アドレス（TIMx_DMAR）

アドレスオフセット：0x4C

リセット値：0x0000

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

ビット 15:0 **DMAB[15:0]**：DMA バーストアクセスレジスタ

DMAR レジスタへの読み出しまたは書き込み動作は、次のアドレスにあるレジスタへのアクセスとなります：

$$(\text{TIMx_CR1 アドレス}) + (\text{DBA} + \text{DMA インデックス}) \times 4$$

ここで、TIMx_CR1 アドレスは制御レジスタ 1 のアドレスであり、DBA は TIMx_DCR レジスタで設定された DMA ベースアドレスであり、DMA インデックスは DMA 転送によって自動的に制御され、範囲は 0 から DBL です（DBL は TIMx_DCR 内で設定）。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

DMA バースト機能の使用例

この例では、タイマ DMA バースト機能を使って CCRx レジスタ値を更新します (x = 2、3、4)。このとき、DMA は CCRx レジスタへハーフワードを転送します。

これは次のステップに従って行います。

1. 対応する DMA チャンネルを次のように設定します。
 - DMA チャンネルペリフェラルアドレスを、DMAR レジスタアドレスとします。
 - DMA チャンネルメモリアドレスを、DMA によって CCRx レジスタに転送されるデータを格納する RAM 内のバッファアドレスとします。
 - 転送データ数 = 3 とします (下の注を参照)。
 - サーキュラモードは無効です。
2. DBA と DBL のビットフィールドを次のように設定することによって、DCR レジスタを設定します。
DBL = 3 転送、DBA = 0xE。
3. TIMx 更新 DMA リクエストを有効にします (DIER レジスタのUDE ビットをセット)。
4. TIMx を有効化
5. DMA チャンネルを有効化注 :

注 : この例は、各 CCRx レジスタが 1 回更新される場合です。たとえば、各 CCRx レジスタが 2 回更新される場合は、転送データ数は 6 になります。データ 1、データ 2、データ 3、データ 4、データ 5、データ 6 を格納する RAM のバッファを例にします。データは、CCRx レジスタに次のように転送されます。最初の更新 DMA リクエストでデータ 1 が CCR2 に転送され、データ 2 は CCR3 に、データ 3 は CCR4 にそれぞれ転送され、2 番目の更新 DMA リクエストでデータ 4 が CCR2 に、データ 5 が CCR3 に、データ 6 が CCR4 にそれぞれ転送されます。

21.4.19 TIM2 オプションレジスタ (TIM2_OR)

アドレスオフセット : 0x50

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|---------|----|---------|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI4_RMP | | ETR_RMP | | |
| | | | | | | | | | | | rw | rw | rw | rw | rw |

ビット 15:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4:3 **TI4_RMP** : 内部トリガ (TIM2_CH4 に接続された TI4) 再割当て
このビットは、ソフトウェアによってセット／クリアされます。
01 : COMP2_OUT に接続された TIM2 TI4 入力
10 : COMP1_OUT に接続された TIM2 TI4 入力
その他 : 論理和 GPIO に接続された TIM2 TI4 入力。デバイスのデータシートの代替機能配置表を参照

ビット 2:0 **ETR_RMP** : Timer2 ETR 再割当て
このビットは、ソフトウェアによってセット／クリアされます。
111 : COMP1_OUT に接続された TIM2 ETR 入力
110 : COMP2_OUT に接続された TIM2 ETR 入力
101 : LSE に接続された TIM2 ETR 入力
100 : HSI48 に接続された TIM2 ETR 入力 (下の注を参照)
その他 : 論理和 GPIO に接続された TIM2 ETR 入力。デバイスのデータシートの代替機能配置表を参照
注 : *TIM2 ETR に HSI48 を供給した場合、最大システム周波数が 32 MHz になるため、この ETR は TIMER2 に内部的にプリスケールする必要があります。*

21.4.20 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマップされます。

表 82. TIM2 レジスタマップとリセット値

| オフセット | レジスタ | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|-----------|------|------|------------|-------|----------|-------|-----------|------|------|-----------|------|-------|-------|----------|-------|------|
| 0x00 | TIMx_CR1 | Res. | Res. | Res. | Res. | Res. | Res. | CKD [1:0] | | ARPE | CMS [1:0] | | DIR | OPM | URS | UDIS | CEN |
| | リセット値 | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | TIMx_CR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TIS | MMS[2:0] | | | CCDS | Res. | Res. | Res. |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | | |
| 0x08 | TIMx_SMCR | ETP | ECE | ETPS [1:0] | | ETF[3:0] | | | | MSM | TS[2:0] | | | Res. | SMS[2:0] | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 0x0C | TIMx_DIER | Res. | TDE | COMDE | CC4DE | CC3DE | CC2DE | CC1DE | UDE | Res. | TIE | Res. | CC4IE | | CC2IE | CC1IE | UIE |
| | リセット値 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | | 0 | 0 | 0 | 0 | 0 |
| 0x10 | TIMx_SR | Res. | Res. | Res. | CC4OF | CC3OF | CC2OF | CC1OF | Res. | Res. | TIF | Res. | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | リセット値 | | | | 0 | 0 | 0 | 0 | | | 0 | | 0 | 0 | 0 | 0 | 0 |

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

表 82. TIM2 レジスタマップとリセット値 (続き)

| オフセット | レジスタ | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|--------------------------|------------|---------------|------|------|---------------------|-------|---------------|------|-----------|---------------|------|------|---------------------|-------|---------------|------|
| 0x14 | TIMx_EGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TG | Res. | CC4G | CC3G | CC2G | CC1G | UG |
| | リセット値 | | | | | | | | | | 0 | | 0 | 0 | 0 | 0 | 0 |
| 0x18 | TIMx_CCMR1 出力比較モード | OC2CE | OC2M [2:0] | | | OC2PE | OC2FE | CC2S [1:0] | | OC1CE | OC1M [2:0] | | | OC1PE | OC1FE | CC1S [1:0] | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | TIMx_CCMR1 入力キャプチャモード | IC2F[3:0] | | | | IC2 PSC [1:0] | | CC2S [1:0] | | IC1F[3:0] | | | | IC1 PSC [1:0] | | CC1S [1:0] | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x1C | TIMx_CCMR2 出力比較モード | OC4CE | OC4M [2:0] | | | OC4PE | OC4FE | CC4S [1:0] | | OC3CE | OC3M [2:0] | | | OC3PE | OC3FE | CC3S [1:0] | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | TIMx_CCMR2 入力キャプチャモード | IC4F[3:0] | | | | IC4 PSC [1:0] | | CC4S [1:0] | | IC3F[3:0] | | | | IC3 PSC [1:0] | | CC3S [1:0] | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x20 | TIMx_CCER | CC4NP | Res. | CC4P | CC4E | CC3NP | Res. | CC3P | CC3E | CC2NP | Res. | CC2P | CC2E | CC1NP | Res. | CC1P | CC1E |
| | リセット値 | 0 | | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 | 0 | 0 | | 0 | 0 |
| 0x24 | TIMx_CNT | CNT[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x28 | TIMx_PSC | PSC[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x2C | TIMx_ARR | ARR[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x30 | Res. | | | | | | | | | | | | | | | | |
| 0x34 | TIMx_CCR1 | CCR1[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x38 | TIMx_CCR2 | CCR2[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x3C | TIMx_CCR3 | CCR3[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x40 | TIMx_CCR4 | CCR4[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x44 | Res. | | | | | | | | | | | | | | | | |

表 82. TIM2 レジスタマップとリセット値（続き）

| オフセット | レジスタ | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-------|-----------|------------|------|------|----------|------|------|------|------|------|------|------|----------|---------|---|---------|---|
| 0x48 | TIMx_DCR | Res. | Res. | Res. | DBL[4:0] | | | | | Res. | Res. | Res. | DBA[4:0] | | | | |
| | リセット値 | | | | 0 | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 |
| 0x4C | TIMx_DMAR | DMAB[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x50 | TIM2_OR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | T14_RMP | | ETR_RMP | |
| | リセット値 | | | | | | | | | | | | | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

22 汎用タイマ (TIM21/22)

22.1 概要

TIM21 および TIM22 汎用タイマは、プログラマブルなプリスケラによって駆動される 16 ビット自動再ロードカウンタで構成されています。

これらのカウンタは、入力信号パルス長の測定（入力キャプチャ）や出力波形の生成（出力比較、PWM）など様々な目的に使用できます。

パルス長と波形の周期は、タイマプリスケラと RCC クロックコントローラプリスケラを使用して、数マイクロ秒から数ミリ秒までの範囲で変化させることができます。

タイマ TIM21 および TIM22 は完全に独立していて、いかなるリソースも共用しません。これらのタイマは、[セクション 22.3.14](#)に示すように、相互に同期させることができます。

22.2 TIM21/22 の主な機能

22.2.1 TIM21/22 の主な機能

汎用タイマ TIM21 および TIM22 の機能は以下のとおりです。

- 16 ビットの アップ、ダウン、アップ/ダウン自動再ロードカウンタ。
- カウンタクロック周波数を 1 から 65535 の間の値で分周するために使用される 16 ビットのプログラム可能なプリスケラ（動作中に変更可能）。
- 次の機能を持つ、最大 2 つの独立チャネル。
 - 入力キャプチャ
 - 出力比較
 - PWM 生成（エッジアラインモードとセンターアラインモード）
 - ワンパルスモード出力
- 外部信号でタイマを制御し、複数のタイマを相互接続する同期回路。
- 以下のイベント時の割り込み生成。
 - 更新：カウンタオーバーフロー/アンダーフロー、カウンタ初期化（ソフトウェアまたは内部トリガによる）
 - トリガイベント（内部トリガによるカウンタの開始、停止、初期化、またはカウント）
 - 入力キャプチャ
 - 出力比較

図 142. 汎用タイマのブロック図 (TIM21/22)

内部クロック (CK_INT)

外部トリガ (ETRx) → ETR → 極性選択 & エッジ → ETRP → 入力フィルタ → ETRF

内部トリガ (ITR0, ITR1) → ITR → TRC → TGI/TRGI

トリガコントローラ (トリガコントローラ, スリープコントローラモード, エンコーダインタフェース)

リセット、イネーブル、アップ、カウント

自動再ロードレジスタ (自動再ロードレジスタ, 停止、クリア)

カウンタ (CNTカウンタ)

キャプチャ/比較レジスタ (キャプチャ/比較1レジスタ, キャプチャ/比較2レジスタ)

出力制御 (出力制御, 出力制御)

出力 (OC1, OC2) → TIMx_CH1, TIMx_CH2

注:

レジスタ

イベント

割り込み

MSv33704V2

22.3 TIM21/22 の機能説明

22.3.1 タイムベースユニット

タイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットカウンタです。カウンタは、カウントアップ、カウントダウン、またはカウントアップとダウンの両方をしますが、カウントダウン、またはカウントアップとダウンの両方の場合もあります。カウンタのクロックは、プリスケアラで分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケアラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットには次の要素が含まれます。

- カウンタレジスタ (TIMx_CNT)
- プリスケアラレジスタ (TIMx_PSC)
- 自動再ロードレジスタ (TIMx_ARR)

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きアクセスは、プリロードレジスタに対して行われます。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット (ARPE) に応じて、常時または更新イベント (UEV) ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、更新イベントはカウンタがオーバーフローしたときに送信されます。また、ソフトウェアによって生成することもできます。更新イベントの生成については、各設定の中で詳しく説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタイネーブルビット (CEN) がセットされているときにのみ、プリスケアラ出力 CK_CNT から供給されます (カウンタの有効化の詳細については、スレーブモードコントローラの説明も参照してください)。

TIMx_CR1 レジスタの CEN ビットがセットされてから、カウンタがカウントを開始するまでに 1 クロックサイクルの遅延があることに注意してください。

プリスケアラの説明

プリスケアラは、カウンタクロック周波数を 1 から 65536 までの間の値で分周することができます。16 ビットレジスタ (TIMx_PSC レジスタ) で制御される 16 ビットカウンタをベースとしています。この制御レジスタはバッファされているので、動作中でも変更できます。新しいプリスケアラ比は、次の更新イベントから有効になります。

[図 144](#) と [図 145](#) に、プリスケアラ比を動作中に変更したときのカウンタの動作の例を示します。

図 143. プリスケアラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

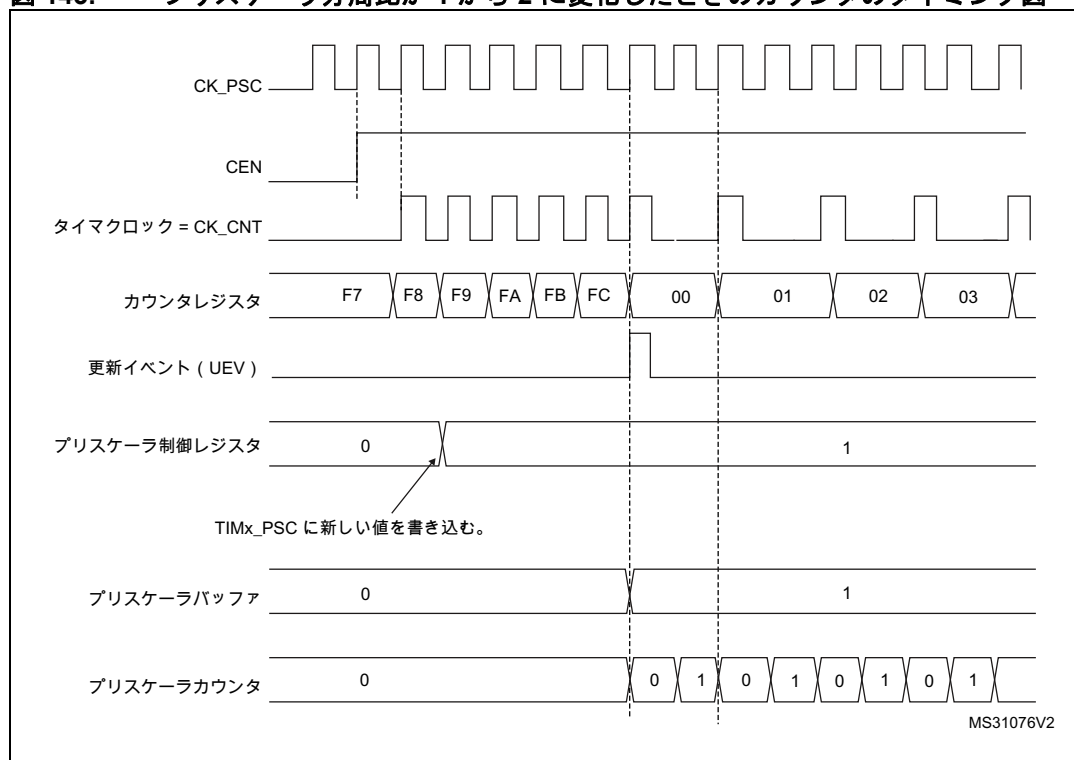
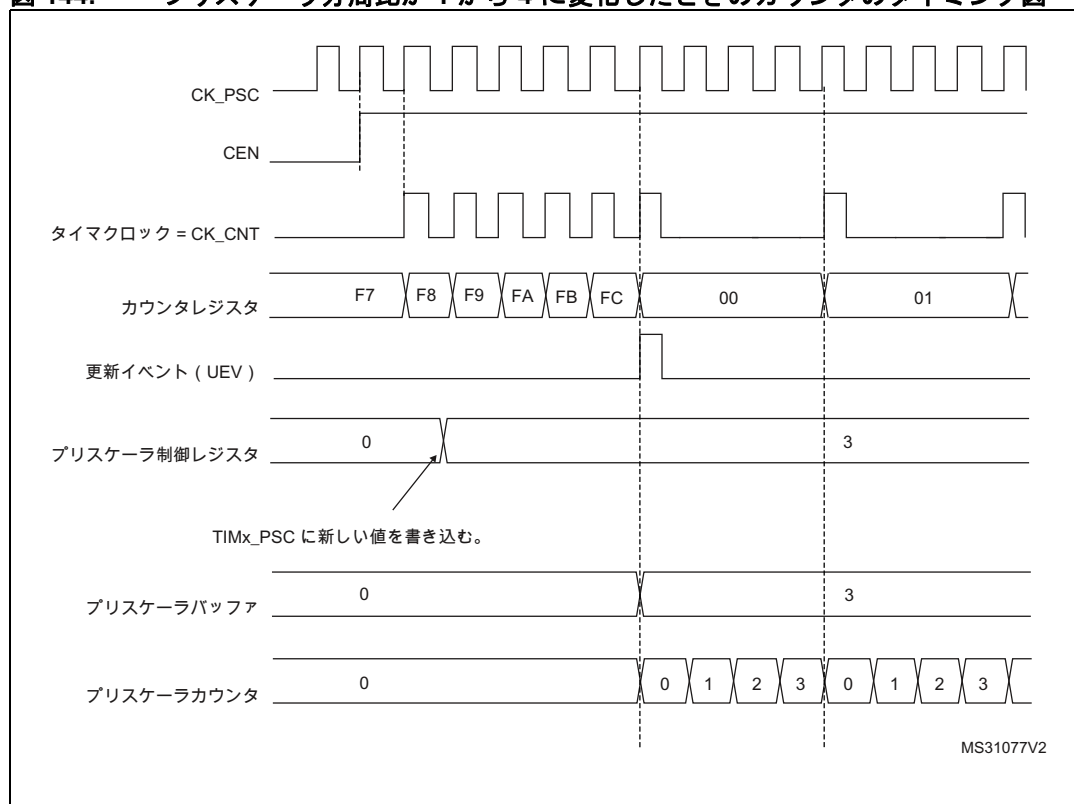


図 144. プリスケアラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



22.3.2 カウントモード

アップカウントモード

アップカウントモードでは、カウンタは 0 から自動再ロード値 (TIMx_ARR レジスタの内容) までカウントしたあと、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

(ソフトウェアによって、または TIM21/TIM22 のスレーブモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることでも更新イベントが生成されます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、プリスケアラのカウンタと同じく、カウンタは 0 からカウントをリスタートします (ただし、プリスケアラ比は変化しません)。さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みが生成されるのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- 自動再ロードシャドウレジスタは、プリロード値 (TIMx_ARR) で更新されます。
- プリスケアラのバッファにはプリロード値 (TIMx_PSC レジスタの内容) が再ロードされます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 145. 内部クロック分周比が 1 の場合のカウンタのタイミング図

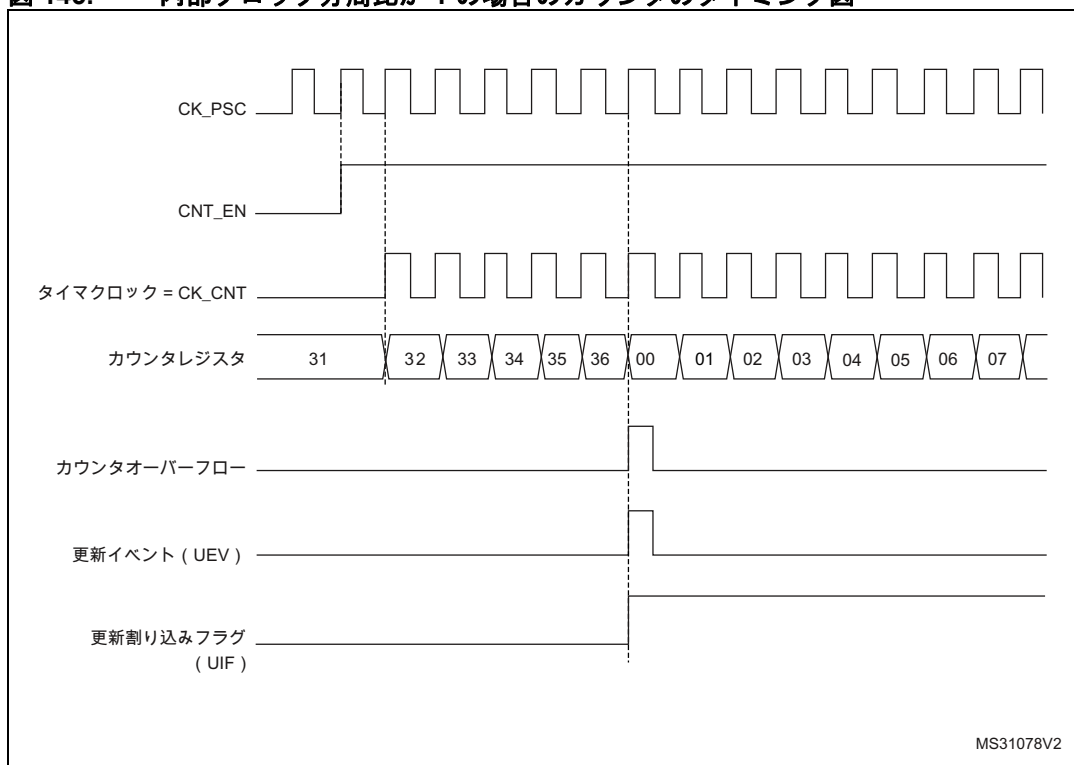


図 146. 内部クロック分周比が 2 の場合のカウンタのタイミング図

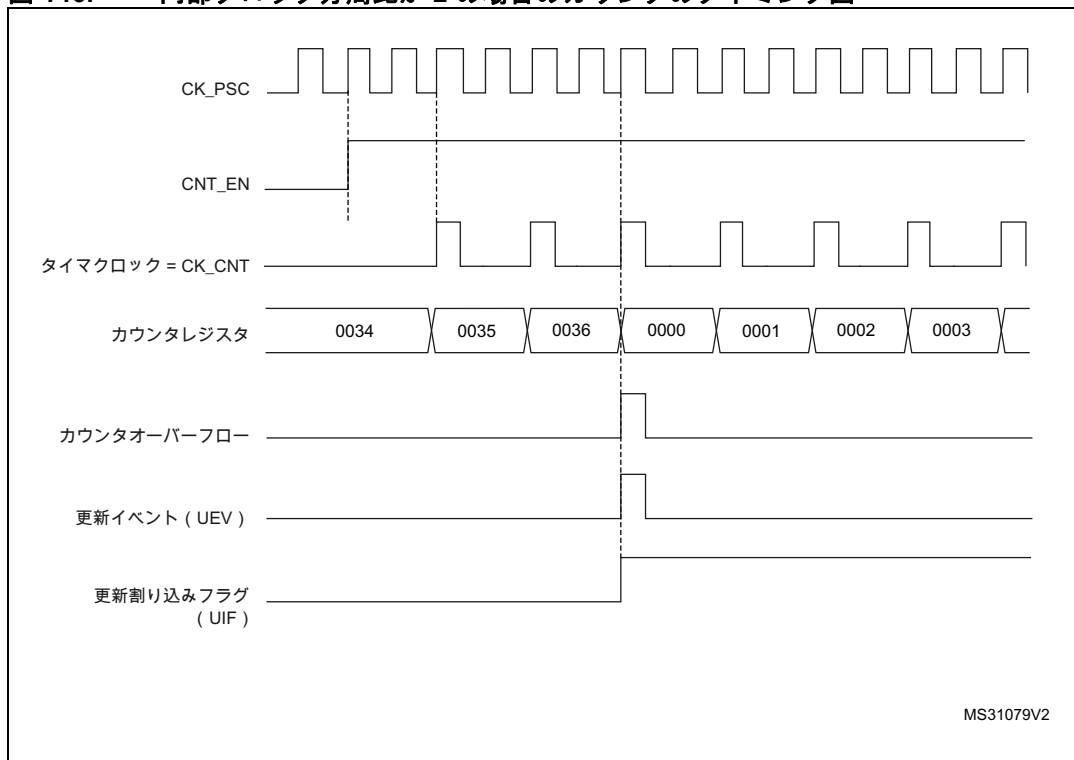


図 147. 内部クロック分周比が 4 の場合のカウンタのタイミング図

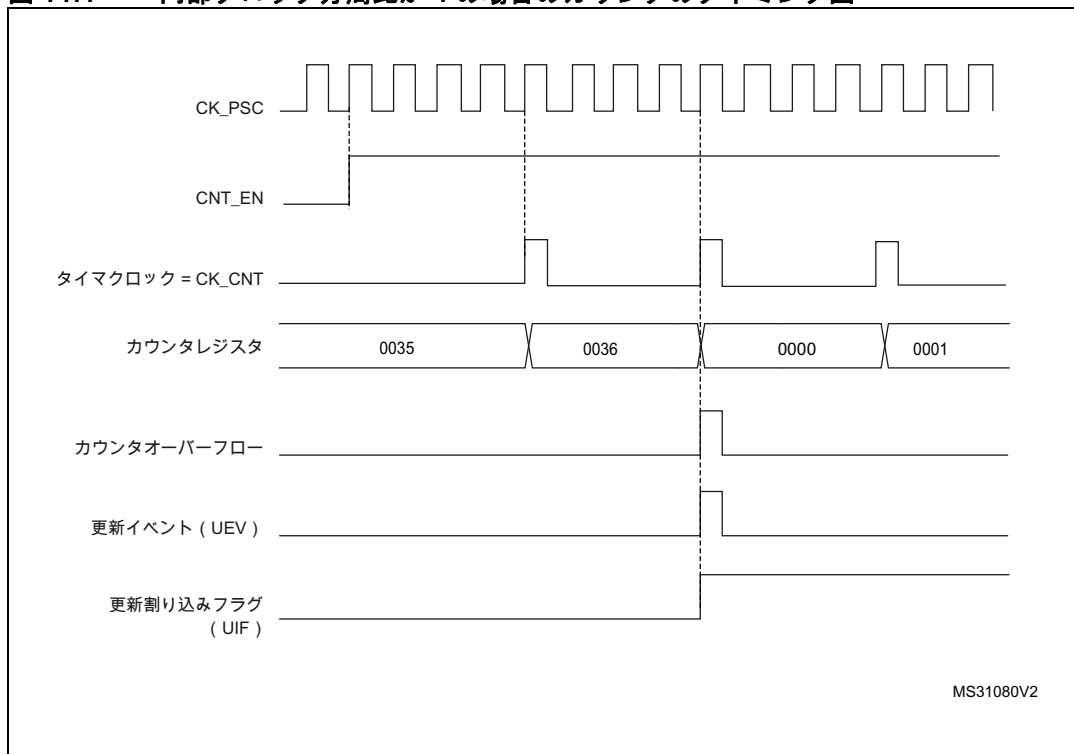


図 148. 内部クロック分周比が N の場合のカウンタのタイミング図

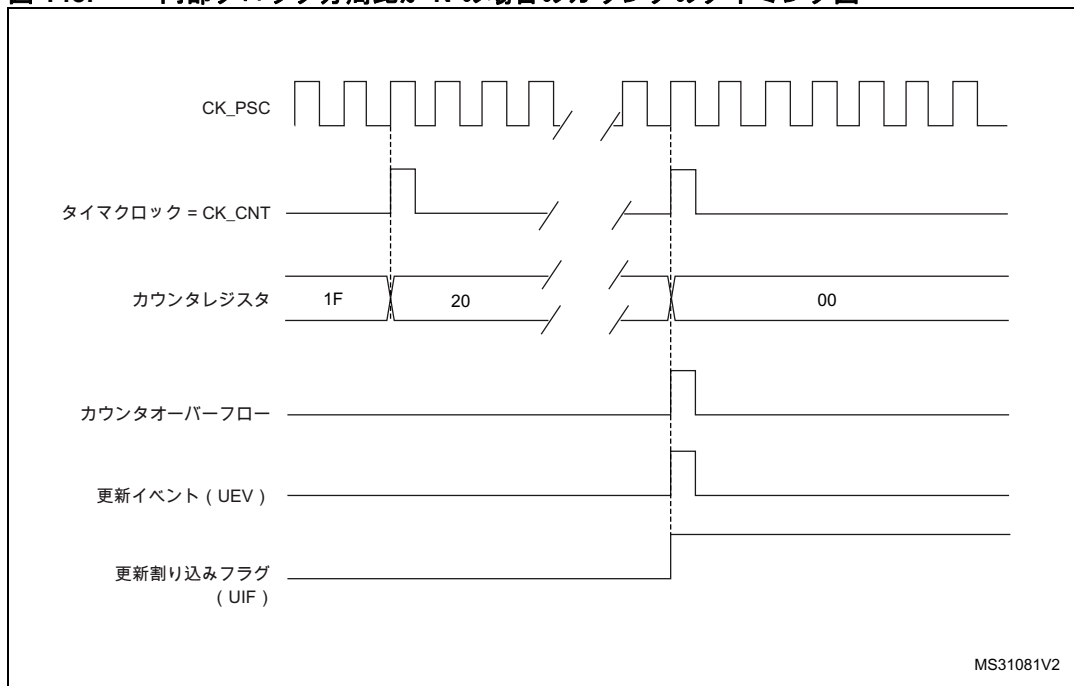


図 149. ARPE = 0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタのタイミング図

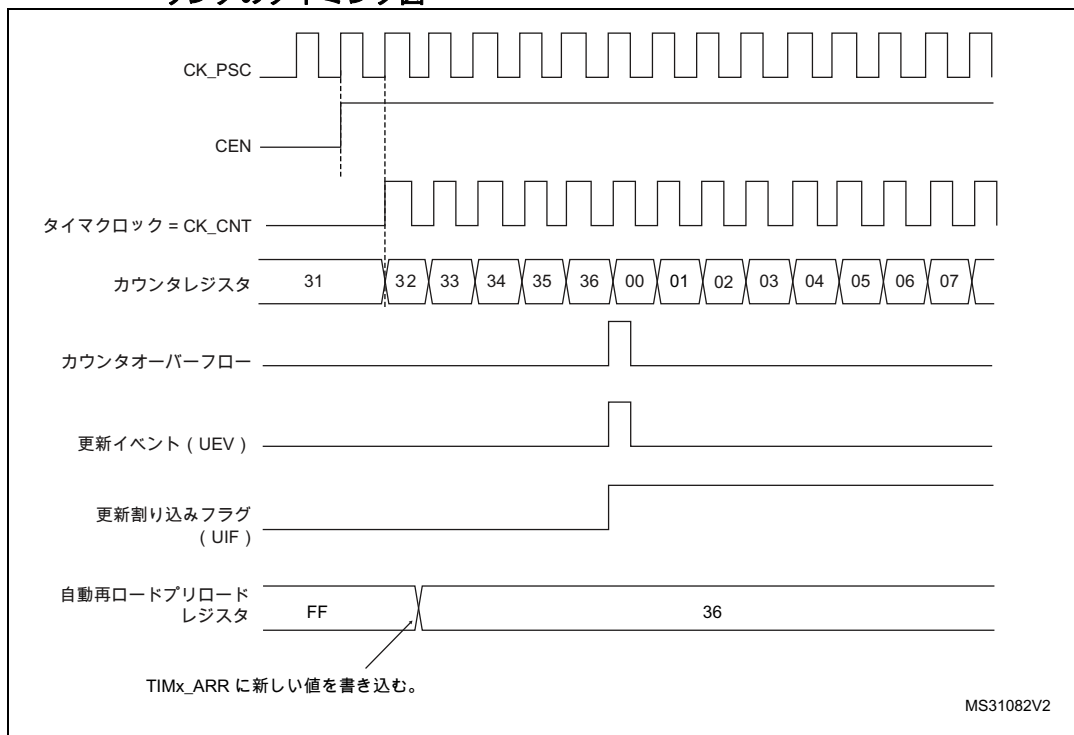
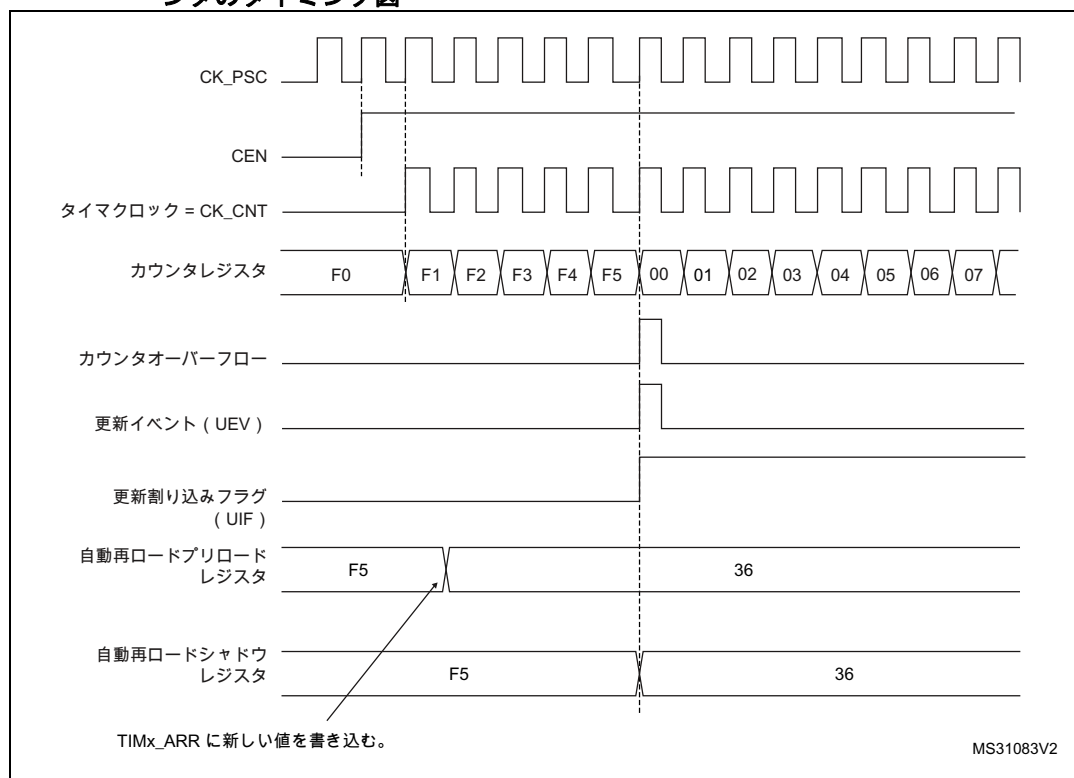


図 150. ARPE = 1 (TIMx_ARR がプリロードされる) の場合の更新イベント時のカウンタのタイミング図



ダウンカウントモード

ダウンカウントモードでは、カウンタは自動再ロード値 (TIMx_ARR レジスタの内容) から 0 までカウントした後、自動再ロード値からカウントダウンをリスタートし、カウンタアンダーフローイベントを生成します。

更新イベントは、カウンタアンダーフローごとに、または、(ソフトウェアで、または、スレーブモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることにより生成できます。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この後 UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値からリスタートしますが、プリスケアラのカウンタは 0 からリスタートします (しかし、プリスケアラ比は変化しません)。

さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みが生成されるのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのバッファにはプリロード値 (TIMx_PSC レジスタの内容) が再ロードされます。
- 自動再アクティブロードレジスタは、プリロード値 (TIMx_ARR の内容) で更新されます。自動再ロードレジスタはカウンタが再ロードされる前に更新されるので、次の周期は期待通りの周期になります。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 151. 内部クロック分周比が 1 の場合のカウンタのタイミング図

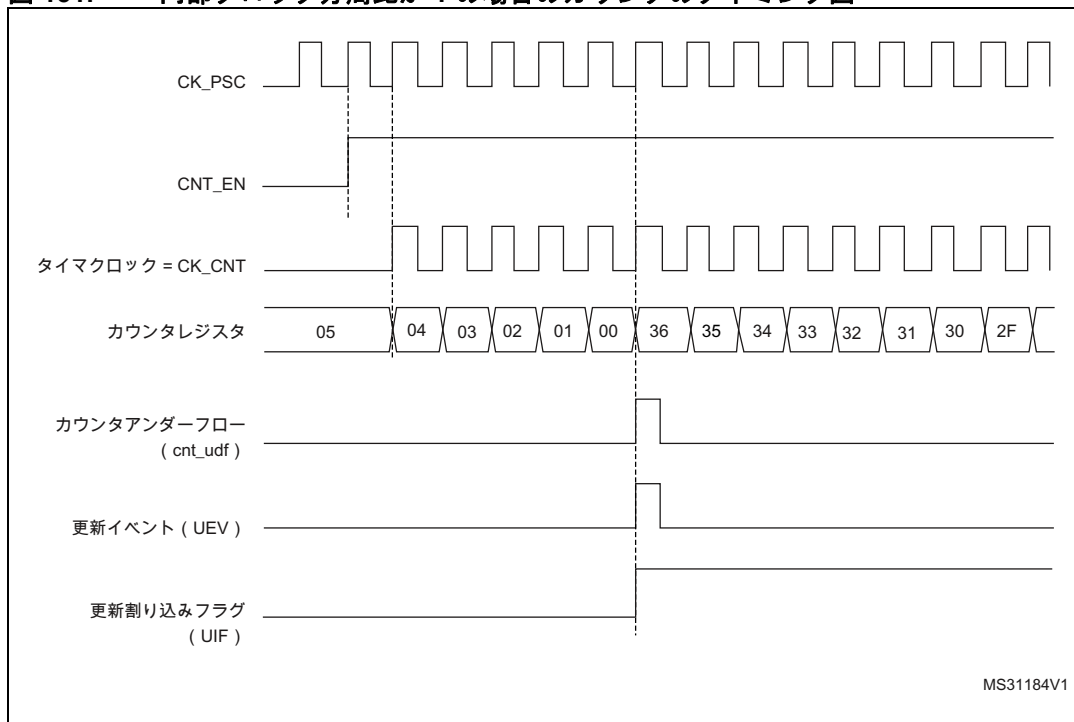


図 152. 内部クロック分周比が 2 の場合のカウンタのタイミング図

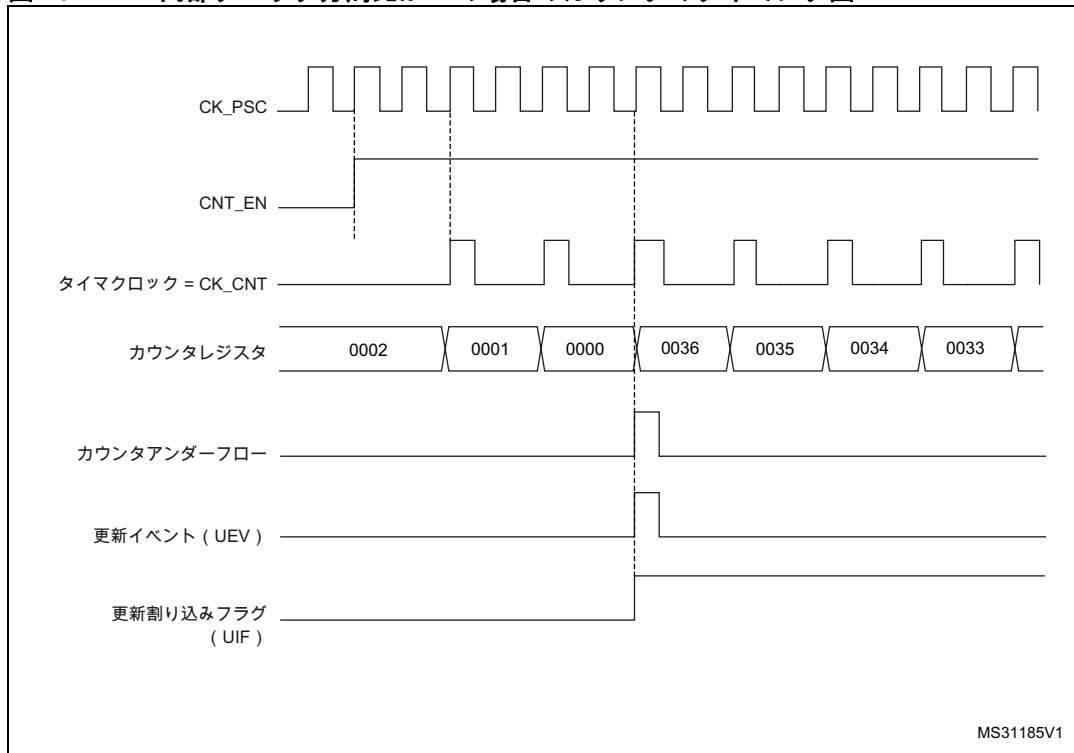


図 153. 内部クロック分周比が 4 の場合のカウンタのタイミング図

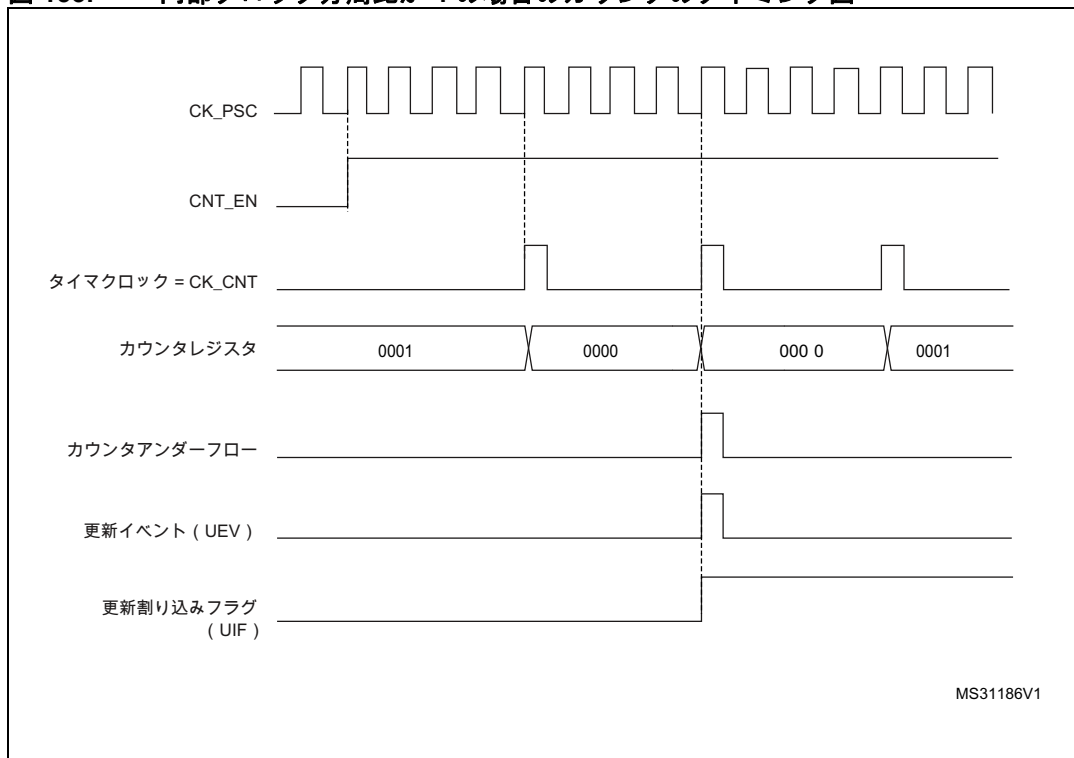
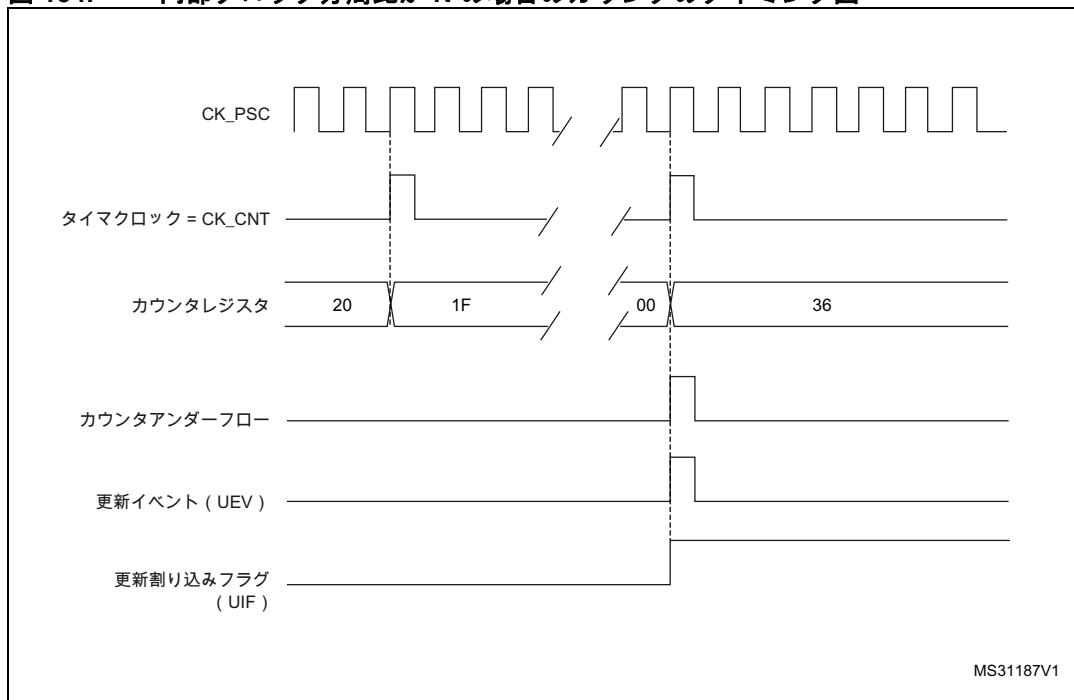


図 154. 内部クロック分周比が N の場合のカウンタのタイミング図



センターアラインモード (アップ/ダウンカウント)

センターアラインモードでは、カウンタは 0 から自動再ロード値 (TIMx_ARR レジスタの内容) -1 までカウントを行い、カウンタオーバーフローイベントを生成します。次に自動再ロード値から 1 までのカウントダウンを行い、カウンタアンダーフローイベントを生成します。その後、0 からカウントをリスタートします。

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブとなります。出力に設定されたチャンネルの出力比較割り込みフラグがセットされるタイミングは、カウンタがカウントダウンしたとき (センターアラインモード 1、CMS = “01”)、カウンタがカウントアップしたとき (センターアラインモード 2、CMS = “10”)、またはカウンタがカウントアップしてカウントダウンしたとき (センターアラインモード 3、CMS = “11”) です。

このモードでは、方向ビット (TIMx_CR1 レジスタの DIR) に書き込むことはできません。このビットは、ハードウェアによって更新されて、カウンタの現在の方向を示します。

更新イベントは、カウンタオーバーフローおよびカウンタアンダーフローごとに、または (ソフトウェアによって、またはスレープモードコントローラを使用して) TIMx_EGR レジスタの UG ビットをセットすることによっても、生成されます。この場合、プリスケアラのカウンタと同じく、カウンタは 0 からカウントをリスタートします。

UEV 更新イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることにより無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。この場合、UDIS ビットに 0 が書き込まれるまで、更新イベントは発生しません。ただし、カウンタは現在の自動再ロード値に基づいて、カウントアップとカウントダウンを続けます。

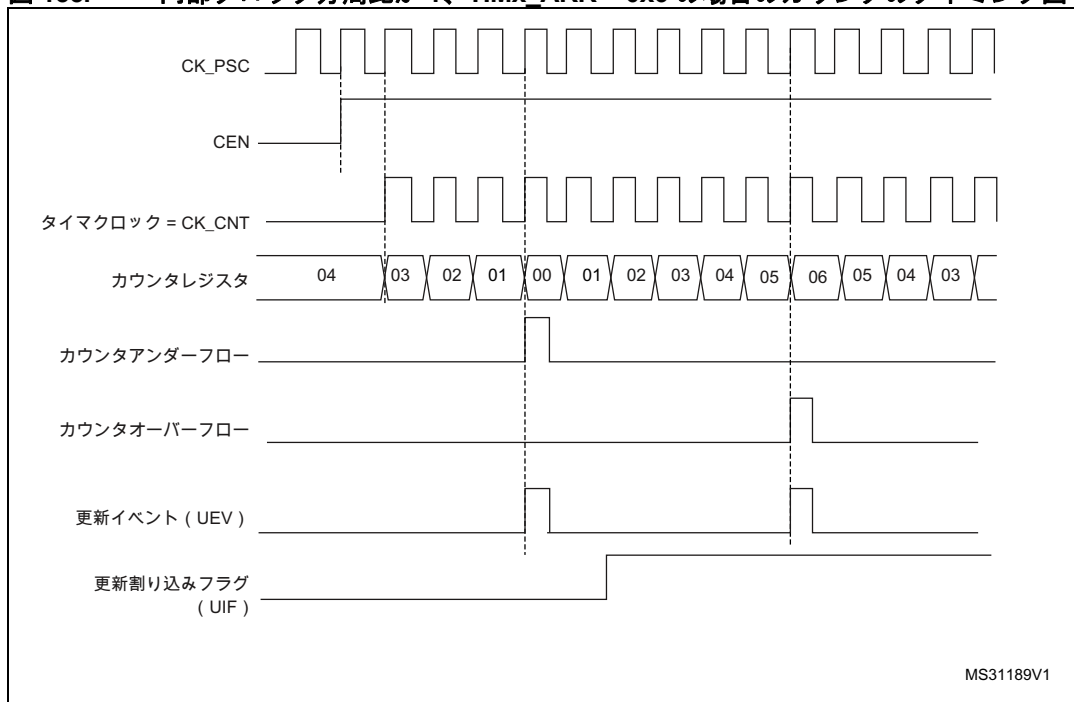
さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みが生成されるのを防ぐためです。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットに応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのパッファにはプリロード値 (TIMx_PSC レジスタの内容) が再ロードされます。
- 自動再ロードアクティブレジスタは、プリロード値 (TIMx_ARR の内容) で更新されます。更新の原因がカウンタオーバーフローである場合には、自動再ロードが更新されてからカウンタが再ロードされるので、次の周期は予想通りの周期になります (カウンタに新しい値がロードされます)。

以下の図は、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 155. 内部クロック分周比が 1、TIMx_ARR = 0x6 の場合のカウンタのタイミング図



1. ここでは、センタアラインモード 1 が使用されています（詳細については、[セクション 22.4.1 : TIM21/22 制御レジスタ 1 \(TIMx_CR1\)](#) (508 ページ) を参照してください）。

図 156. 内部クロック分周比が 2 の場合のカウンタのタイミング図

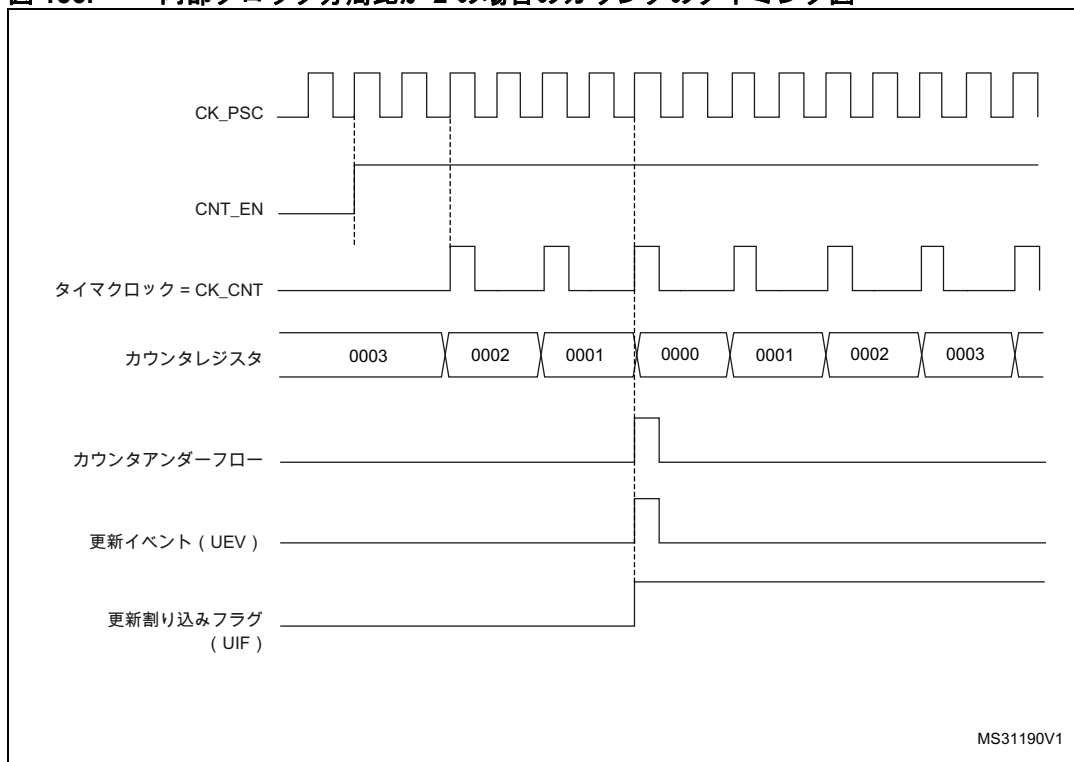
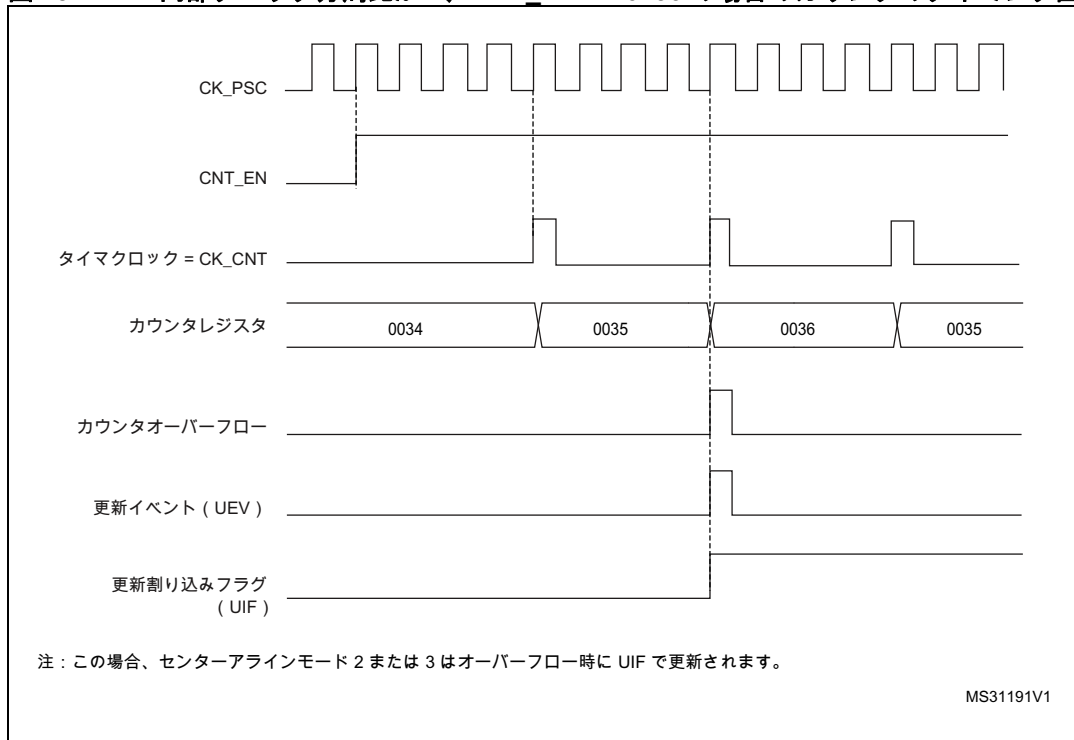


図 157. 内部クロック分周比が 4、TIMx_ARR = 0x36 の場合のカウンタのタイミング図



1. センターアラインモード 2 または 3 が使用され、オーバーフロー時に UIF がセットされます。

図 158. 内部クロック分周比が N の場合のカウンタのタイミング図

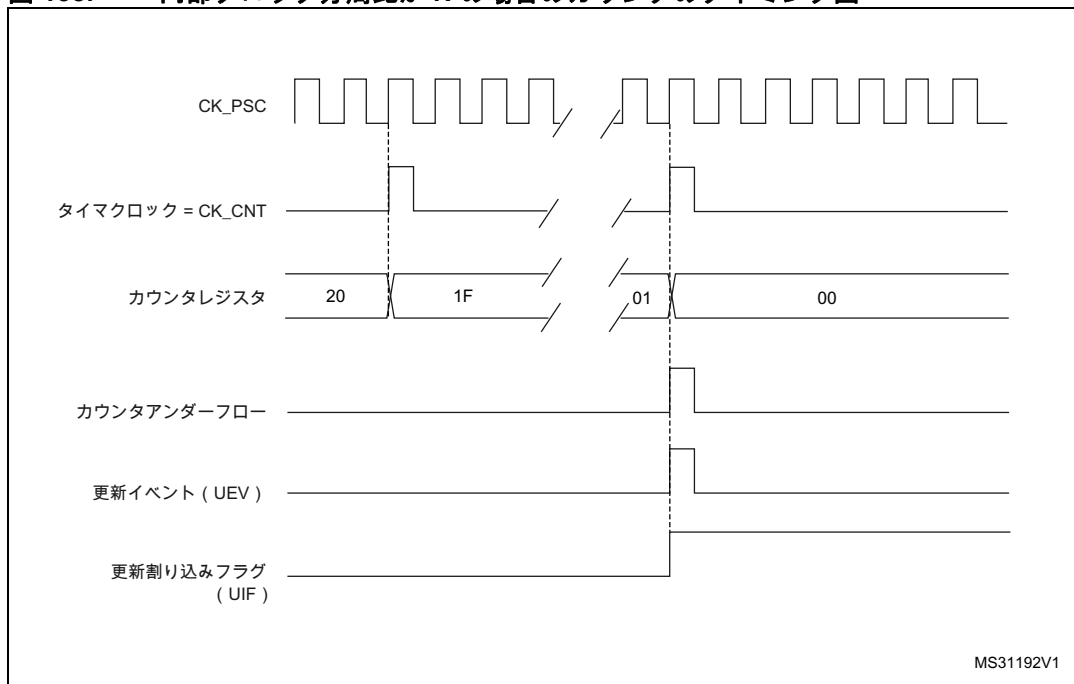


図 159. **ARPE = 1 (カウンタアンダーフロー) の場合の更新イベント時のカウンタのタイミング図**

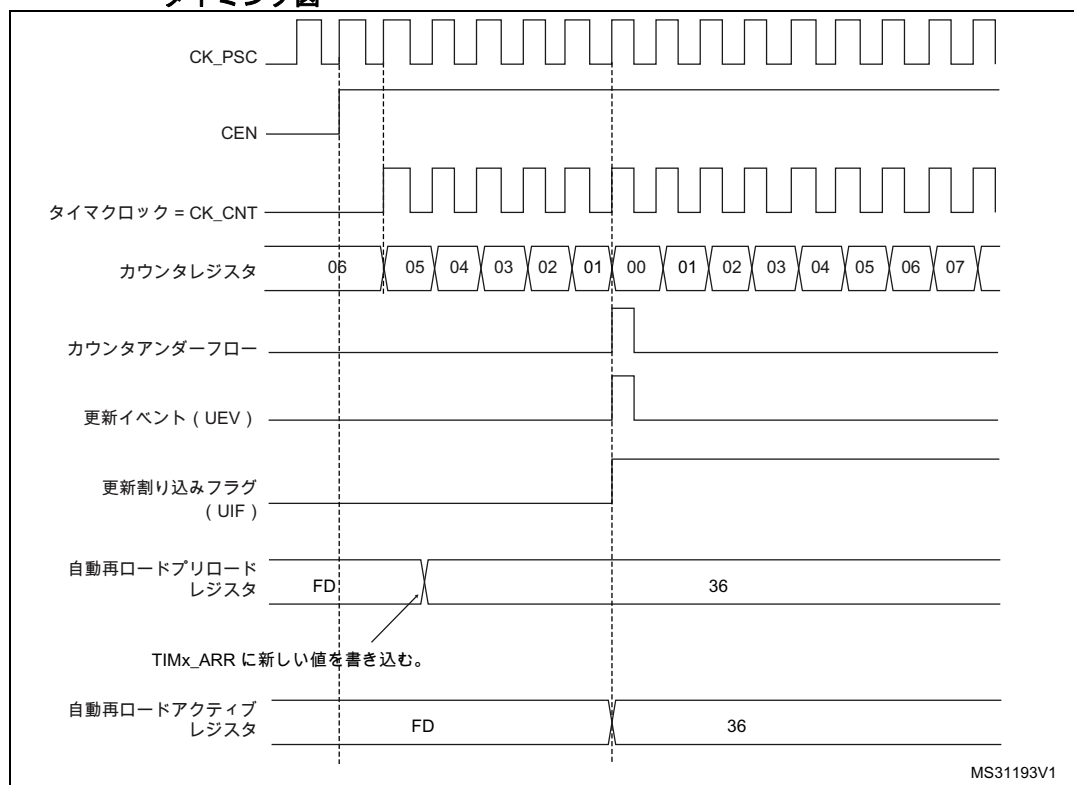
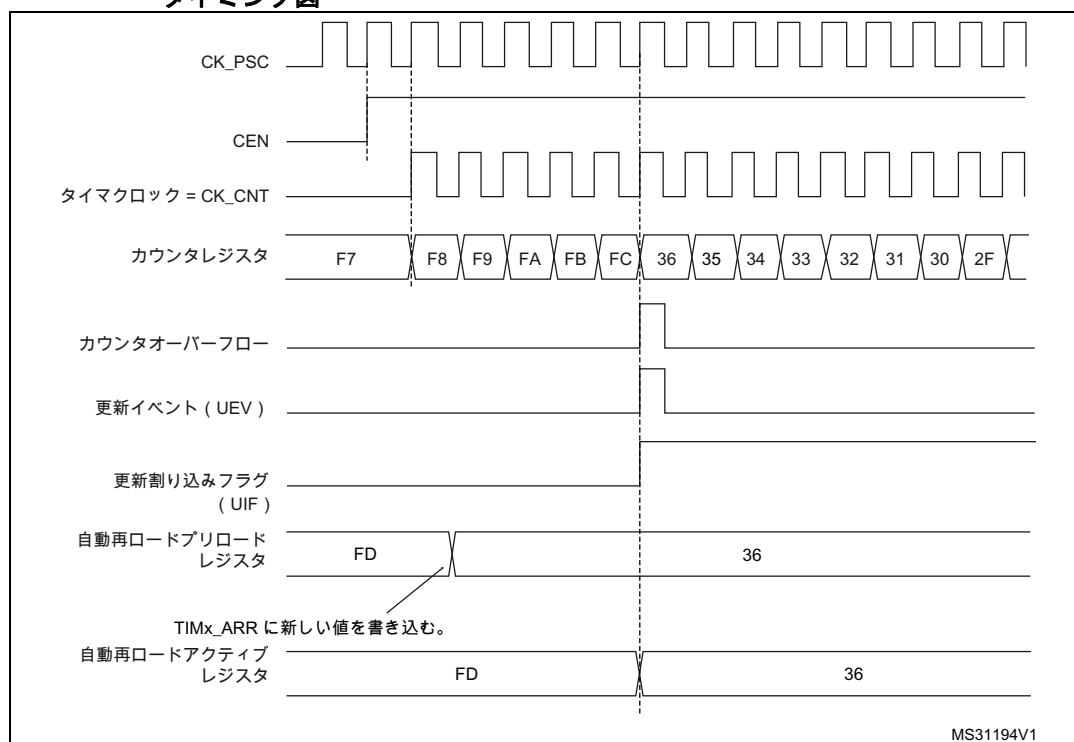


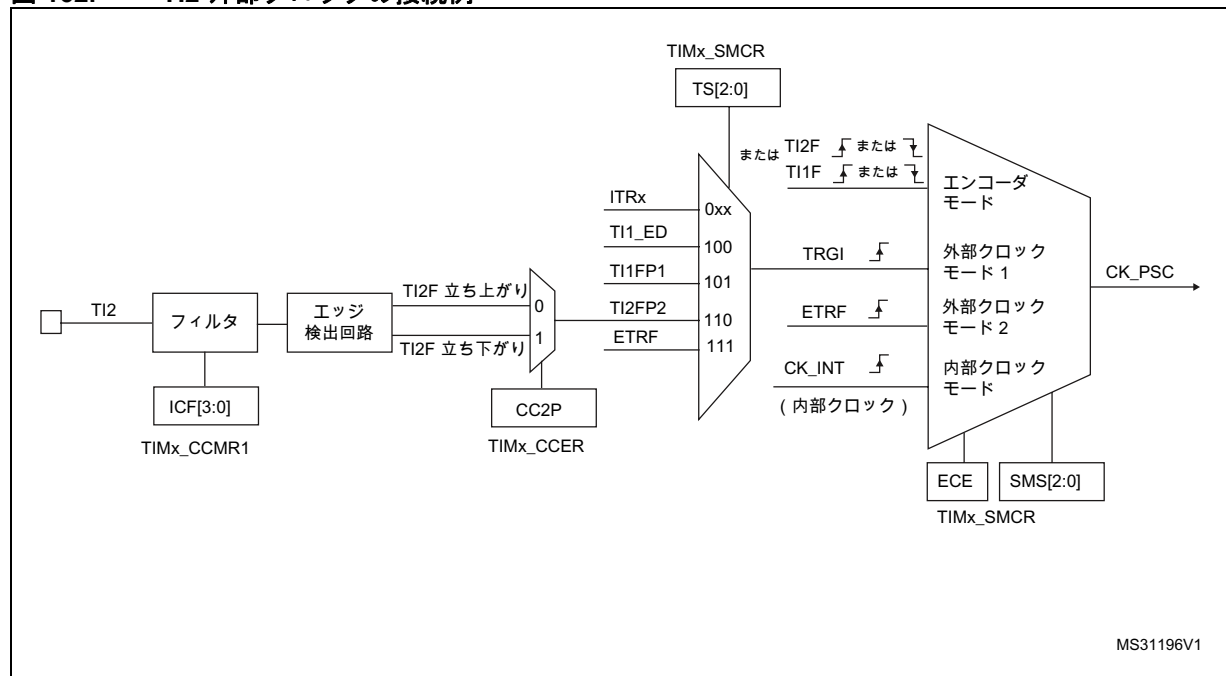
図 160. **ARPE = 1 (カウンタオーバーフロー) の場合の更新イベント時のカウンタのタイミング図**



外部クロックソースモード 1 (TIM21)

このモードは TIMx_SMCR レジスタの SMS = 111 のときに選択されます。カウンタは、選択された入力の立ち上がりまたは立ち下がりエッジでカウントすることができます。

図 162. TI2 外部クロックの接続例



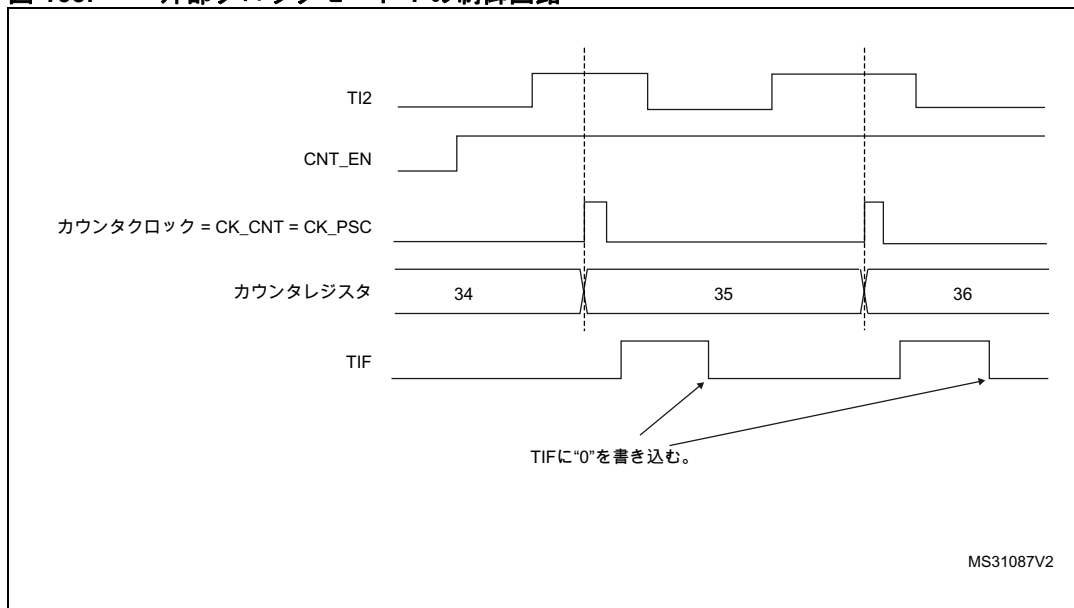
たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、チャンネル 2 が TI2 入力の立ち上がりエッジを検出するように設定します。
2. TIMx_CCMR1 レジスタの IC2F[3:0] ビットに書き込むことによって、入力フィルタ時間を設定します（フィルタを使用しない場合は、IC2F = 0000 にしておきます）。
3. TIMx_CCER レジスタの CC2P ビットと CC2NP ビットに“0”を書き込むことによって、立ち上がりエッジ極性を選択します。
4. TIMx_SMCR レジスタの SMS ビットに“111”を書き込むことによって、タイマを外部クロックモード 1 に設定します。
5. TIMx_SMCR レジスタの TS ビットに“110”を書き込むことによって、トリガ入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを有効にします。

注： キャプチャプリスケアラはトリガには使用されないで、設定は不要です。

TI2 の立ち上がりエッジが発生すると、カウンタは 1 カウントを行い、TIF フラグがセットされます。TI2 の立ち上がりエッジから実際のカウンタクロックまでの間には、TI2 入力の再同期回路による遅延があります。

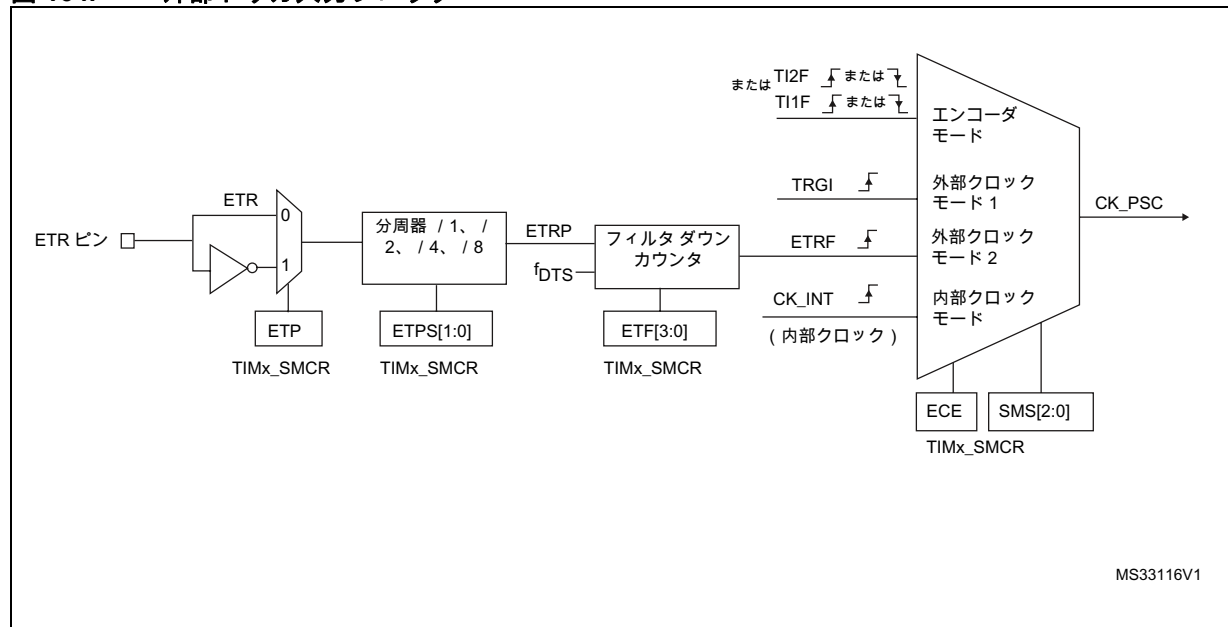
図 163. 外部クロックモード 1 の制御回路



外部クロックソースモード 2

このモードは、TIMx_SMCR レジスタの ECE ビットに "1" を書き込むことによって選択されます。カウンタは、外部トリガ入力 ETR の立ち上がりまたは立ち下がりエッジごとにカウントできます。[図 164](#) に、外部トリガ入力ブロックの概要を示します。

図 164. 外部トリガ入力ブロック



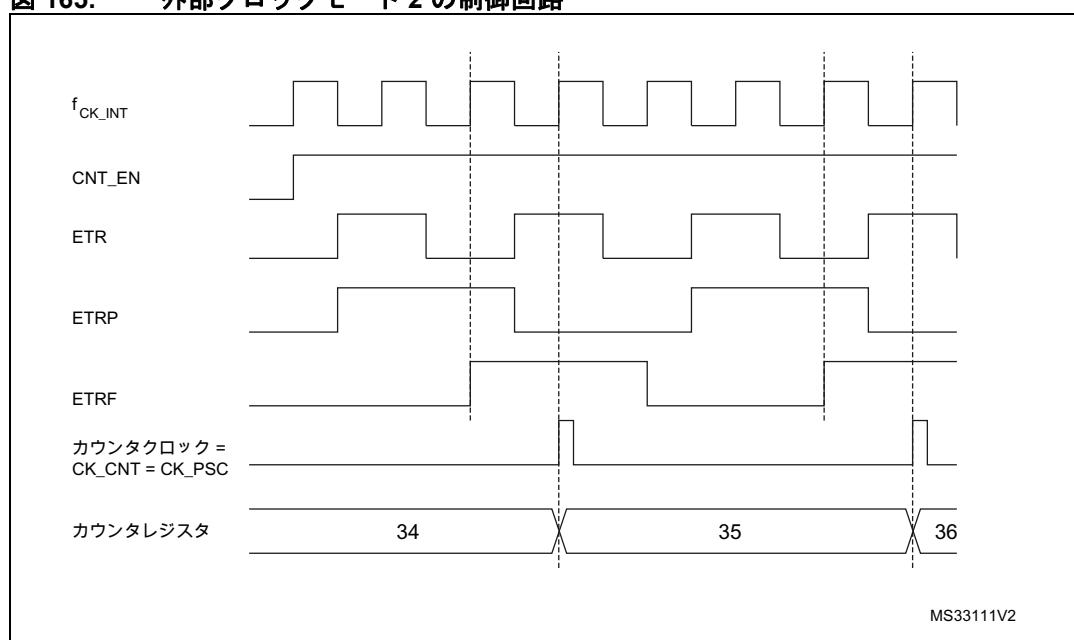
たとえば、ETR の 2 回の立ち上がりエッジごとにカウントするようにアップカウンタを設定するには、以下の手順に従います。

1. この例ではフィルタは不要なので、TIMx_SMCR レジスタの ETF[3:0] に“0000”を書き込みます。
2. TIMx_SMCR レジスタの ETPS[1:0] ビットに“01”を書き込むことによって、プリスケアラを設定します。
3. TIMx_SMCR レジスタの ETP ビットに“0”を書き込むことによって、ETR ピンの立ち上がりエッジ検出を選択します。
4. TIMx_SMCR レジスタの ECE ビットに“1”を書き込むことによって、外部クロックモード 2 を有効にします。
5. TIMx_CR1 レジスタの CEN ビットに“1”を書き込むことによって、カウンタを有効にします。

カウンタは 2 回の ETR 立ち上がりエッジごとに 1 回カウントします。

ETR の立ち上がりエッジから実際のカウンタクロックまでの間に、ETRP 信号の再同期回路による遅延があります。

図 165. 外部クロックモード 2 の制御回路



22.3.4 キャプチャ／比較チャネル

各キャプチャ／比較チャネルは、キャプチャ／比較レジスタ（シャドウレジスタを含む）、キャプチャの入カステージ（デジタルフィルタ、マルチプレクサ、プリスケアラ）、および出カステージ（コンパレータと出力制御）で構成されています。

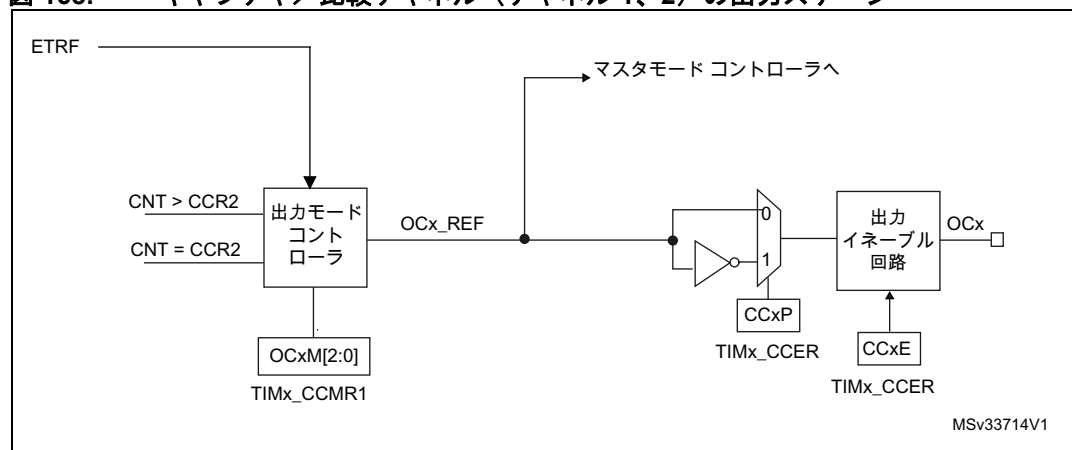
図 166 から 図 168 に、1 つのキャプチャ／比較 チャネルの概要を示します。

入カステージは、対応する Tlx 入力をサンプリングして、フィルタリングを行った TlxF 信号を生成します。次に、極性選択付きのエッジ検出回路が、スレーブモードコントローラによってトリガ入力として、またはキャプチャコマンドとして使用される信号（TlxFPx）を生成します。この信号はプリスケアラを通じて、キャプチャレジスタ（ICxPS）に渡されます。

The diagram illustrates the internal logic of the Sleep Mode Controller. It starts with an input **TI1** which is divided by **f_{DTS}** and fed into a **フィルタ ダウンカウンタ** (Filter Down Counter). The output of this counter is **TI1F**, which is connected to the **エッジ検出回路** (Edge Detection Circuit). The edge detection circuit outputs **TI1F 立ち上がり** (TI1F rising edge) and **TI1F 立ち下がり** (TI1F falling edge) signals. These signals are combined via an OR gate to produce the **TI1F_ED** signal, which is sent to the **スリープモードコントローラへ** (to the Sleep Mode Controller). The rising and falling edge signals are also fed into a 2-to-1 multiplexer (labeled 0 and 1) to produce the **TI1FP1** signal. The **TI1FP1** signal is then fed into another 2-to-1 multiplexer (labeled 01, 10, 11). The inputs to this multiplexer are **TI1FP1** (input 01), **TI2FP1** (input 10), and **TRC** (input 11). The **TRC** signal is derived from the **CC1P/CC1NP** input through the **TIMx_CCER** register. The output of the second multiplexer is the **IC1** signal, which is then divided by a **分周器 /1, /2, /4, /8** (Divider) to produce the final **IC1PS** output. The divider is controlled by the **ICPS[1:0]** bits from the **TIMx_CCMR1** register. The **CC1S[1:0]** bits from **TIMx_CCMR1** and the **CC1E** bit from **TIMx_CCER** are also shown as inputs to the logic.

[illegible]

図 168. キャプチャ／比較チャネル（チャネル 1、2）の出力ステージ



キャプチャ／比較ブロックは、1つのプリロードレジスタと1つのシャドウレジスタで構成されています。書き込みおよび読み出しアクセスは、常にプリロードレジスタに対して行われます。

キャプチャモードでは、キャプチャ動作は実際にはシャドウレジスタで行われ、その値がプリロードレジスタにコピーされます。

比較モードでは、プリロードレジスタの内容がシャドウレジスタにコピーされて、カウンタと比較されます。

22.3.5 入力キャプチャモード

入力キャプチャモードでは、対応する ICx 信号によって遷移が検出された後、カウンタの値をラッチするために、キャプチャ／比較レジスタ (TIMx_CCRx) が使用されます。キャプチャが発生すると、対応する CCxIF フラグ (TIMx_SR レジスタ) がセットされ、割り込みまたは DMA リクエストを送信できます (有効な場合)。CCxIF フラグがすでにハイのときにキャプチャが発生した場合は、オーバーキャプチャフラグ CCxOF (TIMx_SR レジスタ) がセットされます。CCxIF フラグは、ソフトウェアで“0”を書き込むことによって、または、TIMx_CCRx レジスタに格納されたキャプチャデータを読み出すことによってクリアできます。CCxOF は、“0”を書き込むとクリアされます。

次の例は、TI1 入力立ち上がったときに、カウンタの値を TIMx_CCR1 にキャプチャする方法を示します。次の手順を使用してください。

1. アクティブな入力を選択します。TIMx_CCR1 は TI1 入力にリンクされていないので、TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込みます。CC1S の値が“00”から変化するたびに、チャンネルは入力モードに設定され、TIMx_CCR1 レジスタは読み出し専用になります。
2. タイマに接続する信号に対して必要とする入力フィルタ時間を設定します (入力が TIx 入力のうちの1つである場合、TIMx_CCMRx レジスタの ICxF ビットを設定)。入力信号の反転時、最大で内部クロックの5サイクルの間、信号が安定しないと想定してみます。この場合、フィルタ時間を5クロックサイクルより長くプログラムする必要があります。新しいレベルの連続した8個のサンプルが検出されたときに、TI1 で遷移を検証できます (周波数 f_{DTS} でサンプリング)。次に、TIMx_CCMR1 レジスタの IC1F ビットに“0011”を書き込みます。

3. TI1 チャンネルのアクティブ遷移のエッジを選択します。このためには、TIMx_CCER レジスタの CC1P ビットと CC1NP ビットに“00”を設定します（この場合、立ち上がりエッジの選択）。
4. 入力プリスケラをプログラムします。この例では有効な遷移ごとにキャプチャを行いたいのので、プリスケラを無効にします（TIMx_CCMR1 レジスタの IC1PS ビットに“00”を書き込む）。
5. TIMx_CCER レジスタの CC1E ビットをセットすることによって、カウンタからキャプチャレジスタへのキャプチャを有効にします。
6. 必要に応じて、TIMx_DIER レジスタの CC1IE ビットをセットすることによって、関連する割り込みリクエストを有効にします。

入力キャプチャが発生すると、

- アクティブ遷移時に、カウンタの値が TIMx_CCR1 レジスタに格納されます。
- CC1IF フラグがセットされます（割り込みフラグ）。CC1OF ビットは、少なくとも 2 回連続でキャプチャが発生した場合にもセットされますが、フラグはクリアされません。
- CC1IE ビットに応じて、割り込みが生成されます。

オーバーキャプチャを処理するために、オーバーキャプチャフラグの前にデータを読み出すことが推奨されます。これにより、フラグ読み出し後やデータ読み出し前に発生するオーバーキャプチャの見落としを避けることができます。

注： TIMx_EGR レジスタの対応する CCxG ビットをセットすることで、IC 割り込みリクエストをソフトウェアで発生させることができます。

22.3.6 PWM 入力モード

このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

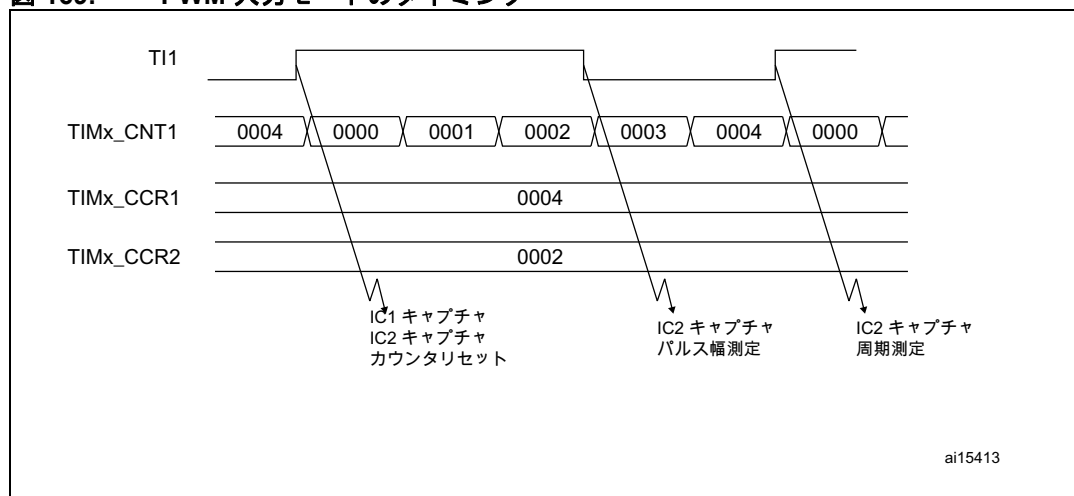
- 2 つの ICx 信号が、同じ TIx 入力に配置されます。
- これら 2 つの ICx 信号は、逆の極性のエッジでアクティブです。
- 2 つの TIxFP 信号の 1 つがトリガ入力として選択され、スレーブモードコントローラはリセットモードに設定されます。

たとえば、次の手順を使用して、TI1 に適用された PWM の周期 (TIMx_CCR1 レジスタ) とデューティサイクル (TIMx_CCR2 レジスタ) を測定できます（手順は、CK_INT 周波数とプリスケラ値によって、若干異なることがあります）。

1. TIMx_CCMR1 レジスタの CC1S ビットに“01”を書き込むことによって (TI1 を選択)、TIMx_CCR1 のアクティブ入力を選択します。
2. CC1P ビットと CC1NP ビットに“00”を書き込むことによって (立ち上がりエッジでアクティブ)、TI1FP1 のアクティブな極性を選択します (TIMx_CCR1 のキャプチャとカウンタクリアの両方に使用)。
3. TIMx_CCMR1 レジスタの CC2S ビットに“10”を書き込むことによって (TI1 を選択)、TIMx_CCR2 のアクティブ入力を選択します。
4. CC2P ビットと CC2NP ビットに“11”を書き込むことによって (立ち下がりエッジでアクティブ)、TI1FP2 のアクティブ極性を選択します (TIMx_CCR2 のキャプチャに使用されます)。
5. TIMx_SMCR レジスタの TS ビットに“101”を書き込むことによって (TI1FP1 を選択)、有効なトリガ入力を選択します。
6. TIMx_SMCR レジスタの SMS ビットに“100”を書き込むことによって、スレーブモードコントローラをリセットモードに設定します。
7. TIMx_CCER レジスタの CC1E と CC2E ビットに“1”を書き込むことによって、キャプチャを有効にします。



図 169. PWM 入力モードのタイミング



1. TI1FP1 と TI2FP2 のみがスレーブモードコントローラに接続されているので、PWM 入力モードは TIMx_CH1/TIMx_CH2 信号でのみ使用できます。

22.3.7 強制出力モード

出力モード (TIMx_CCMRx レジスタの CCxS ビット = 00) では、出力比較レジスタとカウンタの間の比較に関係なく、各出力比較信号 (OCxREF、次に OCx) をソフトウェアによって直接、強制的にアクティブまたはインアクティブレベルにできます。

出力比較信号 (OCxREF/OCx) を強制的にアクティブレベルにするには、対応する TIMx_OCMRx レジスタの OCxM ビットに“101”を書き込みます。これにより、OCxREF は強制的にハイになり (OCxREF は常にアクティブハイ)、OCx は CCxP 極性ビットと逆の値になります。

例: CCxP = 0 (OCx アクティブハイ) => OCx は強制的にハイレベルになります。

OCxREF 信号は、TIMx_CCMRx レジスタの OCxM ビットに“100”を書き込むことによって、強制的にローにできます。

いずれにしても、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されるので、フラグをセットすることができます。それに応じて、割り込みリクエストを送信できます。これについては、次の出力比較モードのセクションで説明します。

22.3.8 出力比較モード

この機能は、出力波形を制御したり、一定時間が経過したことを示すために使用されます。

キャプチャ/比較レジスタとカウンタの値が一致すると、出力比較は次のように機能します。

1. 対応する出力ピンに、出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) と出力極性 (TIMx_CCER レジスタの CCxP ビット) によって定義されたプログラム可能な値を割り当てます。一致した際、出力ピンは、レベルを維持するか (OCxM = 000)、アクティブにセットされるか (OCxM = 001)、インアクティブにセットされるか (OCxM = 010)、または反転されます (OCxM = 011)。
2. 割り込みステータスレジスタのフラグをセットします (TIMx_SR レジスタの CCxIF ビット)。
3. 対応する割り込みマスク (TIMx_DIER レジスタの CCxIE ビット) がセットされている場合は、割り込みを生成します。

TIMx_CCRx レジスタは、プリロードレジスタを使用するしないにかかわらず、TIMx_OCMRx レジスタの OCxPE ビットを使用してプログラムできます。

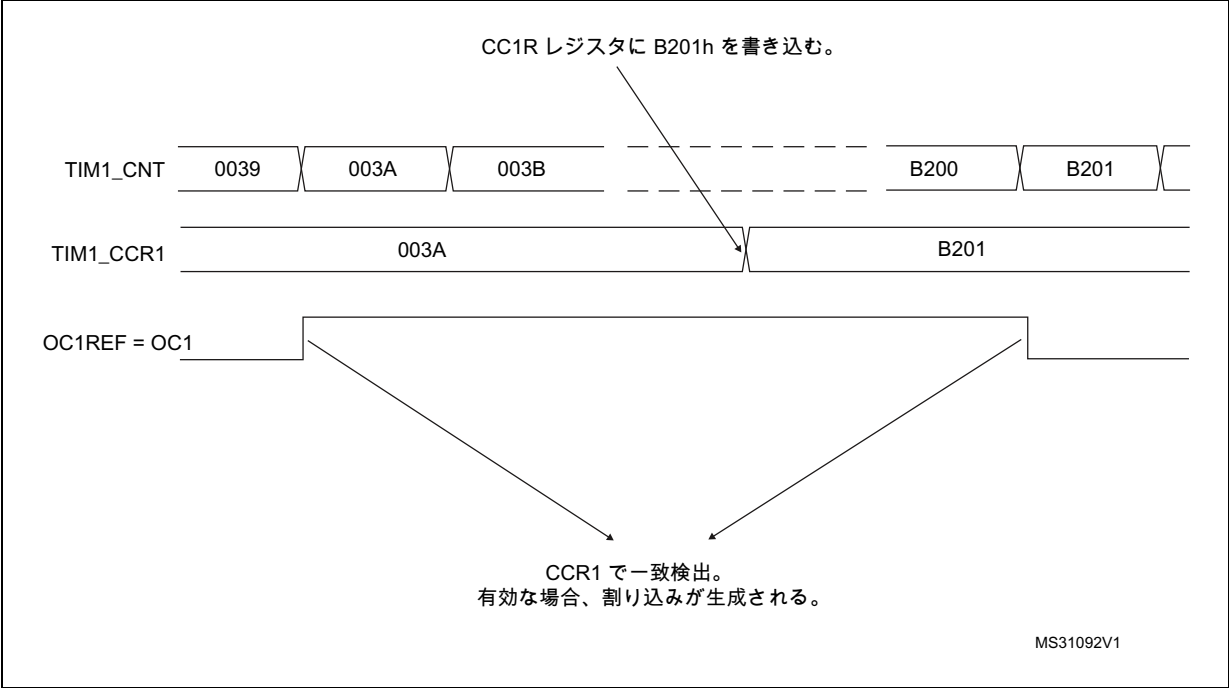
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミング分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用できます (ワンパルスモード)。

手順 :

1. カウンタクロックを選択します (内部、外部、プリスケアラ)。
2. TIMx_ARR レジスタと TIMx_CCRx レジスタに目的のデータを書き込みます。
3. 割り込みリクエストを生成する場合は、CCxIE ビットをセットします。
4. 出力モードを選択します。例 :
 - CNT と CCRx が一致したときに OCx 出力ピンを反転するには、OCxM ビットに“011”を書き込みます。
 - プリロードレジスタを無効にするには、OCxPE ビットに“0”を書き込みます。
 - アクティブハイ極性を選択するには、CCxP ビットに“0”を書き込みます。
 - 出力を有効にするには、CCxE ビットに“1”を書き込みます。
5. TIMx_CR1 レジスタの CEN ビットをセットすることによって、カウンタを有効にします。

いつでもソフトウェアによって TIMx_CCRx レジスタを更新して、出力波形を制御できます。ただし、プリロードレジスタが有効でない場合に限り (OCxPE=0)。そうでない場合、TIMx_CCRx シャドウレジスタは、次の更新イベント UEV でのみ更新されます。例を 図 170 に示します。

図 170. 出力比較モード、OC1 の反転



22.3.9 PWM モード

パルス幅変調(PWM)モードでは、TIMx_ARR レジスタの値によって決められた周波数と TIMx_CCRx レジスタの値によって決められたデューティサイクルで信号を生成できます。

PWM モードは、TIMx_CCMRx レジスタの OCxM ビットを書き込むことによって、チャンネルごとに選択できます (OCx 出力ごとに 1 つの PWM)。TIM20 と TIM21 では、エッジアラインモードのみが使用できます。TIMx_CCMRx レジスタの OCxPE ビットをセットすることによって、対応するプリロードレジスタを有効にする必要があります、最終的には、TIMx_CR1 レジスタの ARPE ビットをセットすることによって、自動再ロードプリロードレジスタも (アップカウントまたはセンターアラインモード) 有効にする必要があります。

プリロードレジスタは、更新イベントが発生したときにのみシャドウレジスタに転送されるので、カウンタを開始する前に、TIMx_EGR レジスタの UG ビットをセットすることによって、すべてのレジスタを初期化しておく必要があります。

OCx の極性は、TIMx_CCER レジスタの CCxP ビットを使用して、ソフトウェアでプログラムできます。アクティブハイまたはアクティブローにプログラムすることができます。OCx 出力は、TIMx_CCER レジスタの CCxE ビットによって有効化されます。詳細については、TIMx_CCERx レジスタの説明を参照してください。

PWM モード(1 または 2)では、TIMx_CNT と TIMx_CCRx を常に比較して、 $TIMx_CNT \leq TIMx_CCRx$ かどうか判断されます。

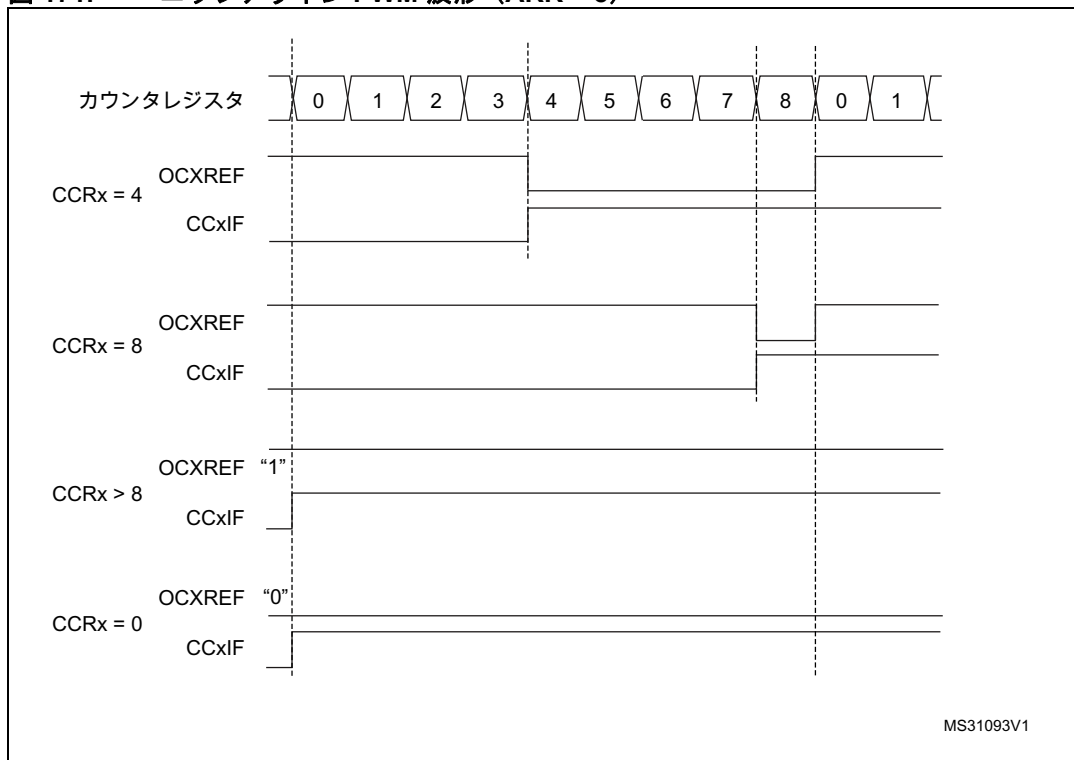
カウンタはカウントアップしているので、タイマはエッジアラインモードでのみ PWM を生成できます。

- アップカウント構成

TIMx_CR1 レジスタの DIR ビットがローのときには、アップカウントがアクティブです。[アップカウントモード \(477 ページ\)](#) を参照してください。

次の例では、PWM モード 1 を使用しています。PWM 基準信号 OCxREF は、 $TIMx_CNT < TIMx_CCRx$ の間はハイに、そうでない場合はローになります。TIMx_CCRx の比較値が自動再ロード値 (TIMx_ARR レジスタの) より大きい場合、OCxREF は“1”に保持されます。比較値が 0 の場合、OCxREF は“0”に保持されます。[図 171](#) に、TIMx_ARR = 8 のときのエッジアライン PWM 波形の例を示します。

図 171. エッジアライン PWM 波形 (ARR = 8)



● ダウンカウント構成

TIMx_CR1 レジスタの DIR ビットがハイのときには、ダウンカウントがアクティブです。[ダウンカウントモード \(480 ページ\)](#) を参照してください。

PWM モード 1 では、基準信号 OCxRef は、TIMx_CNT > TIMx_CCRx の間はローであり、そうでない場合はハイになります。TIMx_CCRx の比較値が TIMx_ARR の自動再ロード値より大きい場合、OCxREF は“1”です。このモードでは、0 % の PWM 信号を生成することはできません。

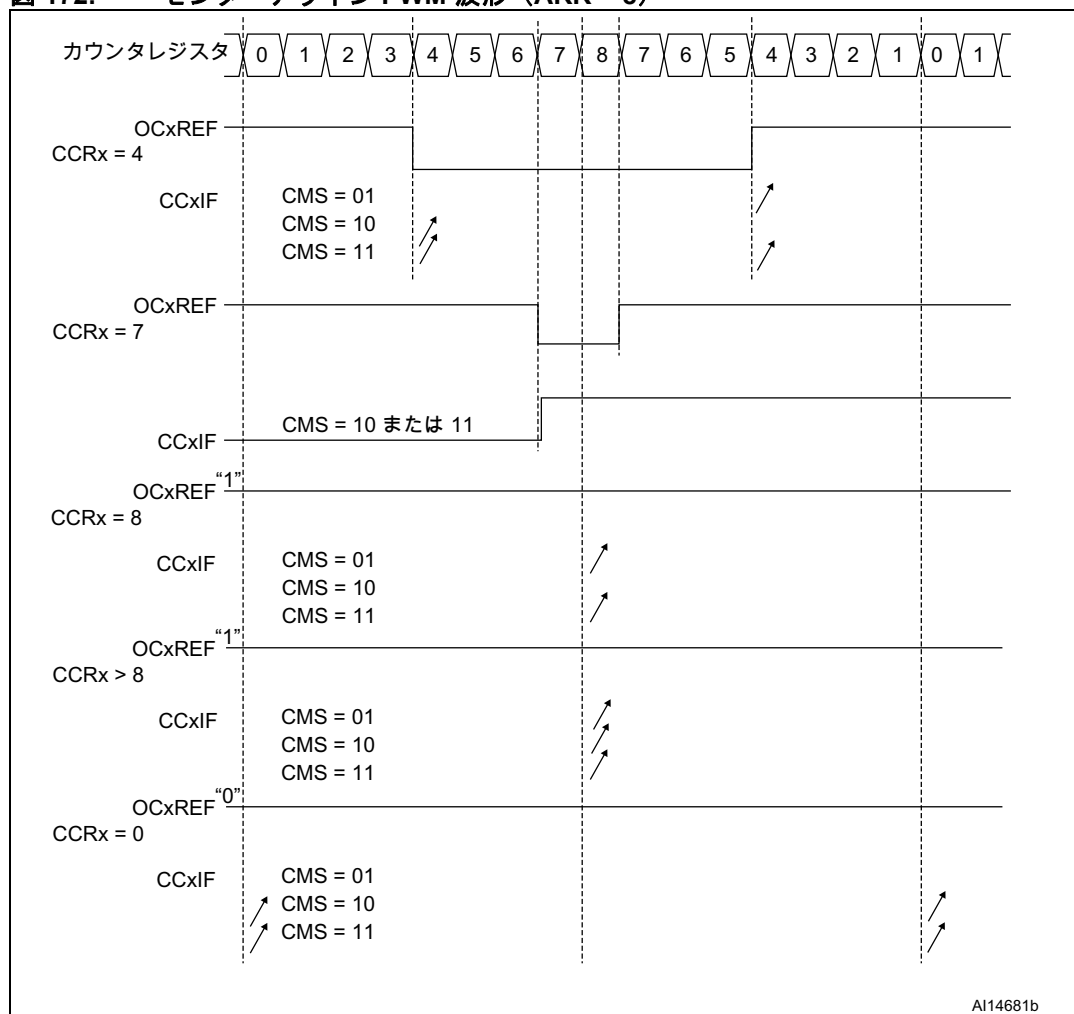
PWM センターアラインモード

センターアラインモードは、TIMx_CR1 レジスタの CMS ビットが“00”でないときにアクティブです（その他すべての構成は、OCxRef/OCx 信号に対して同じ効果を持ちます）。比較フラグは、CMS ビットの設定に応じて、カウンタがカウントアップ、カウントダウン、またはカウントアップとカウントダウンしているときにセットされます。TIMx_CR1 レジスタの方向ビット (DIR) はハードウェアによって更新されており、ソフトウェアで値を変更することはできません。[センターアラインモード \(アップ/ダウンカウント\) \(483 ページ\)](#) を参照してください。

図 172 に、次の条件でのセンターアライン PWM 波形の例を示します。

- TIMx_ARR = 8、
- PWM モードは PWM モード 1、
- フラグは、TIMx_CR1 レジスタの CMS = 01 の場合に選択されたセンターアラインモード 1 に対応して、カウンタがカウントダウンするときにセットされます。

図 172. センターアライン PWM 波形 (ARR = 8)



センターアラインモードの使用に関するヒント

- センターアラインモードを開始するときには、現在のアップ/ダウン設定が使用されます。これは、TIMx_CR1 レジスタの DIR ビットに書き込まれた値に応じて、カウンタがカウントアップまたはカウントダウンすることを意味します。さらに、DIR ビットと CMS ビットをソフトウェアによって同時に変更することはできません。
- センターアラインモードで動作中のカウンタへの書き込みは、予期しない結果を招くことがあるので推奨されません。特に、次のようなケースに注意してください。
 - 自動再ロード値より大きい値をカウンタに書き込んだ場合 (TIMx_CNT > TIMx_ARR)、方向は更新されません。たとえば、カウンタがカウントアップしていた場合、カウンタはカウントアップを続けます。
 - カウンタに 0 または TIMx_ARR 値が書き込まれた場合、方向は更新されますが、更新イベント UEV は生成されません。
- センターアラインモードを使用する最も安全な方法は、カウンタを開始する直前に、ソフトウェアによって更新を生成し (TIMx_EGR レジスタの UG ビットをセット)、動作中はカウンタへの書き込みを行わないことです。

22.3.10 外部イベントによる OCxREF 信号のクリア

特定のチャンネルの OCxREF 信号は、ETRF 入力にハイレベルを適用する（対応する TIMx_CCMRx レジスタの OCxCE イネーブルビットを“1”にセットする）ことによってローに駆動できます。OCxREF 信号は、次の更新イベント UEV が発生するまで、ローレベルを保ちます。

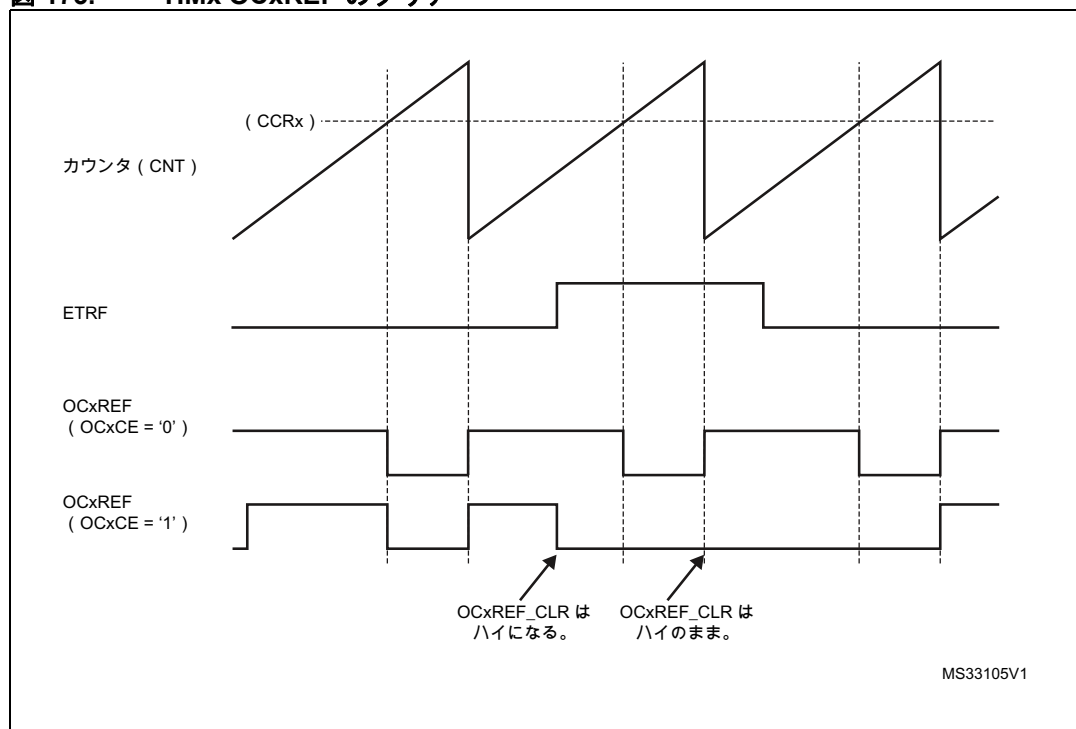
この機能は、出力比較モードと PWM モードでのみ使用でき、強制モードでは機能しません。

たとえば、ETR 信号を比較回路の出力に接続して、電流処理に使用することができます。この場合、ETR は次のように設定する必要があります。

1. 外部トリガプリスケラはオフ状態に維持します (TIMx_SMCR レジスタの ETPS[1:0] ビットを“00”にセット)。
2. 外部クロックモード 2 を無効にします (TIMx_SMCR レジスタの ECE ビットを“0”にセット)。
3. 外部トリガ極性 (ETP) と外部トリガフィルタ (ETF) は、ユーザのニーズに応じて設定できます。

図 173 に、イネーブルビット OCxCE の両方の値について、ETRF 入力が高レベルになったときの OCxREF 信号の動作を示します。この例では、TIMx タイマは PWM モードにプログラムされています。

図 173. TIMx OCxREF のクリア



注: 100% デューティサイクルの PWM の場合 (CCRx > ARR の場合)、次のカウンタオーバーフローで OCxREF が再度有効になります。

22.3.11 ワンパルスモード

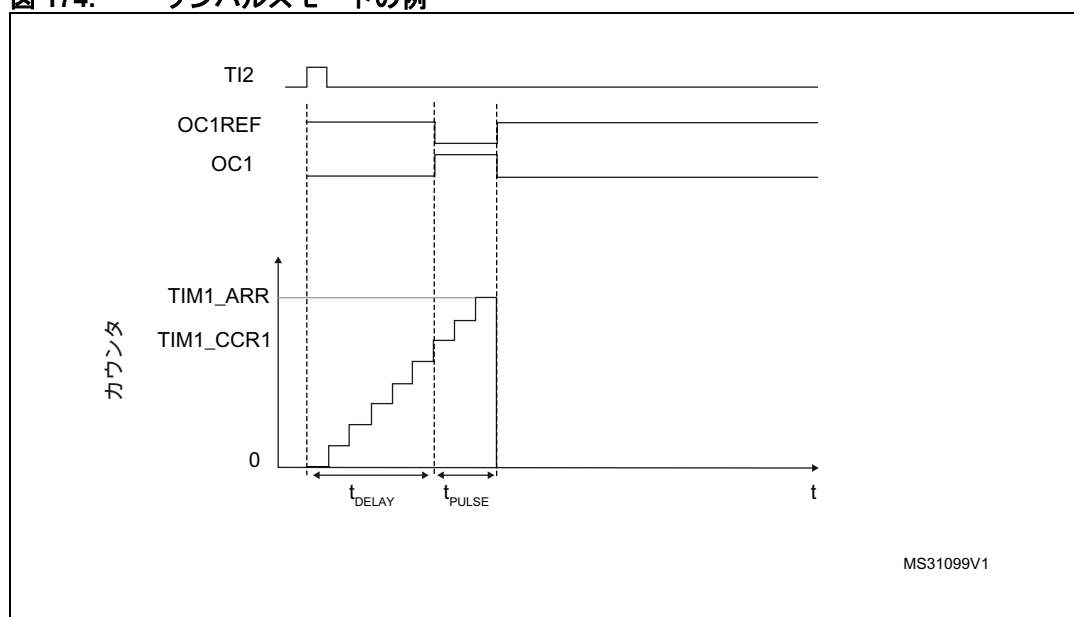
ワンパルスモード (OPM : One Pulse Mode) は、これまでに説明したモードの特殊ケースです。このモードでは、トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

カウンタの開始は、スレーブモードコントローラで制御できます。波形の生成は、出力比較モードまたは PWM モードで行うことができます。ワンパルスモードを選択するには、TIMx_CR1 レジスタの OPM ビットをセットします。これによって、カウンタは、次の更新イベント UEV で自動的に停止します。

パルスは、比較値がカウンタの初期値と異なる場合のみ、正しく生成されます。開始する前に (タイマがトリガを待っているときに)、設定が次のようであればなりません。

$$CNT < CCRx \leq ARR \text{ (特に } 0 < CCRx \text{ の場合)}$$

図 174. ワンパルスモードの例



たとえば、TI2 入力ピンで正のエッジが検出されたときに、OC1 にパルス幅が t_{PULSE} の正のパルスを遅延時間 t_{DELAY} の後に生成することもできます。

TI2FP2 をトリガ 1 として使用します。

1. TIMx_CCMR1 レジスタの CC2S ビットに“01”を書き込むことによって、TI2FP2 を TI2 に配置します。
2. TI2FP2 は、立ち上がりエッジを検出して、TIMx_CCER レジスタの CC2P ビットと CC2NP ビットに“0”を書き込みます。
3. TI2FP2 をスレーブモードコントローラのトリガ (TRGI) として設定します。このためには、TIMx_SMCR レジスタの TS ビットに“110”を書き込みます。
4. TI2FP2 を使用してカウンタを開始します。このためには、TIMx_SMCR レジスタの SMS ビットに“110” (トリガモード) を書き込みます。

OPM 波形は、次のように比較レジスタに書き込むことによって定義されます (クロック周波数とカウンタプリスケアラを考慮に入れて)。

- t_{DELAY} は、TIMx_CCR1 レジスタに書き込まれた値によって定義されます。
- t_{PULSE} は、自動再ロード値と比較値の差 (TIMx_ARR - TIMx_CCR1) によって定義されます。
- 比較一致が発生したときに 0 から 1 へ遷移し、カウンタが自動再ロード値に達したときに 1 から 0 へ遷移する波形を生成するとします。このためには、TIMx_CCMR1 レジスタの OC1M ビットに“111”を書き込むことによって、PWM モード 2 を有効にします。必要に応じて、TIMx_CCMR1 レジスタの OC1PE ビットに“1”を書き込み、TIMx_CR1 レジスタの ARPE ビットに書き込むことによって、プリロードレジスタを有効にすることもできます。この場合、TIMx_CCR1 レジスタに比較値を書き込み、TIMx_ARR レジスタに自動再ロード値を書き込みます。次に、UG ビットをセットすることによって更新を生成し、TI2 で外部トリガイイベントを待ちます。この例では、CC1P に“0”を書き込みます。

上の例では、TIMx_CR1 レジスタの DIR および CMS ビットはローでなければなりません。

必要なパルスは 1 つだけなので (シングルモード)、TIMx_CR1 レジスタの OPM ビットに“1”を書き込みます。こうすると、次の更新時 (カウンタが自動再ロード値に達して、“0”に戻るとき) にカウンタは停止します。TIMx_CR1 レジスタの OPM ビットが“0”にセットされると、繰り返しモードが選択されます。

特殊なケース : OCx 高速イネーブル

ワンパルスモードでは、TIx 入力のエッジ検出によって、カウンタを有効にする CEN ビットがセットされます。その後、カウンタと比較値の比較によって、出力が反転されます。ただし、このような動作には数クロックサイクルが必要なので、実現可能な最小遅延 (t_{DELAY} 分) を制限します。

最小遅延で波形を出力したい場合は、TIMx_CCMRx レジスタの OCxFE ビットをセットします。こうすると、OCxREF (および OCx) は、比較動作を行うことなく、強制的にトリガに反応します。新しいレベルは、比較が一致したときと同じです。OCxFE は、チャンネルが PWM1 または PWM2 モードに設定された場合のみ機能します。

22.3.12 エンコーダインタフェースモード

エンコーダインタフェースモードを選択するには、TIMx_SMCR レジスタの SMS ビットに、カウンタが TI2 エッジのみをカウントしている場合は“001”を、TI1 エッジのみをカウントしている場合は“010”を、TI1 と TI2 の両方のエッジをカウントしている場合は“011”を書き込みます。

TI1 と TI2 の極性を選択するには、TIMx_CCER レジスタの CC1P ビットと CC2P ビットをプログラムします。CC1NP と CC2NP はローに維持する必要があります。必要なときには、入力フィルタもプログラムできます。CC1NP と CC2NP はローに維持する必要があります。

2 つの入力 TI1 と TI2 は、インクリメンタルエンコーダとのインタフェースに使用されます。表 83 を参照してください。カウンタのクロック供給は、TI1FP1 または TI2FP2 (入力フィルタおよび極性選択の後) は TI1 と TI2、フィルタされず反転されない場合は TI1FP1=TI1、フィルタされず反転されない場合は TI2FP2=TI2) での有効な遷移ごとに行われます。ただし、カウンタは有効である (TIMx_CR1 レジスタの CEN ビットに“1”が書き込まれている) ことが前提です。2 つの入力の遷移シーケンスが評価されて、カウントパルスと方向信号を生成します。シーケンスに応じて、カウンタはカウントアップまたはカウントダウンし、TIMx_CR1 レジスタの DIR ビットがハードウェアによって変更されます。カウンタが TI1 のみ、TI2 のみ、または TI1 と TI2 の両方をカウントしている場合でも、DIR ビットは、いずれかの入力 (TI1 または TI2) の遷移のたびに計算されます。

エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。つまり、カウンタは 0 と TIMx_ARR レジスタの自動再ロード値の間で連続的にカウントします (方向に応じて、0 から ARR まで、または ARR から 0 まで)。したがって、開始前に TIMx_ARR を設定する必要があります。同様に、キャプチャ、比較、プリスケアラ、およびトリガ出力機能は、通常動作を続けます。




このモードでは、カウンタは直交 エンコーダの速度と方向に応じて自動的に変更されます。したがって、カウンタの内容は、常にエンコーダの位置を表します。カウンタの方向は、接続されているセンサの回転方向に対応します。次の表は、カウント方向とエンコーダ信号の可能な組み合わせを示します (TI1 と TI2 は同時に切り替わらないと想定しています)。

表 83. カウント方向とエンコーダ信号

| アクティブ エッジ | 他方の信号のレベル (TI2 に対する TI1FP1、 TI1 に対する TI2FP2) | TI1FP1 信号 | | TI2FP2 信号 | |
|------------------------|--|-----------|--------|-----------|--------|
| | | 立ち上がり | 立ち下がり | 立ち上がり | 立ち下がり |
| TI1 のみ カウント | ハイ | ダウン | アップ | カウントなし | カウントなし |
| | ロー | アップ | ダウン | カウントなし | カウントなし |
| TI2 のみ カウント | ハイ | カウントなし | カウントなし | アップ | ダウン |
| | ロー | カウントなし | カウントなし | ダウン | アップ |
| TI1 と TI2 の 両方をカウント | ハイ | ダウン | アップ | アップ | ダウン |
| | ロー | アップ | ダウン | ダウン | アップ |

外部インクリメンタルエンコーダは、外部インタフェースロジックなしに、MCU に直接接続できます。ただし、エンコーダの差分出力をデジタル信号に変換するために、通常、コンパレータが使用されます。これにより、耐ノイズ性が大幅に向上します。機械的なゼロ位置を示す 3 番目のエンコーダ出力は、外部割り込み入力に接続して、カウンタのリセットをトリガできます。

 175 に、カウント信号の生成と方向制御を含むカウンタの動作例を示します。また、両方のエッジが選択されているときの入力ジッタの補正方法も示します。この状況は、センサの位置が一方のスイッチングポイントの近くにあるときに生じることがあります。下の例では、以下のような設定となっています。

- CC1S = 01 (TIMx_CCMR1 レジスタ、TI1FP1 は TI1 に配置)
- CC2S = 01 (TIMx_CCMR2 レジスタ、TI2FP2 は TI2 に配置)
- CC1P = 0、CC1NP = 0 (TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1 = TI1)
- CC2P = 0、CC2NP = 0、IC1F = 0000 (TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2 = TI2)
- SMS = 011 (TIMx_SMCR レジスタ、両方の入力立ち上がり立ち下がり両エッジでアクティブ)
- CEN = 1 (TIMx_CR1 レジスタ、カウンタ有効)

図 175. エンコーダインタフェースモードにおけるカウンタの動作例

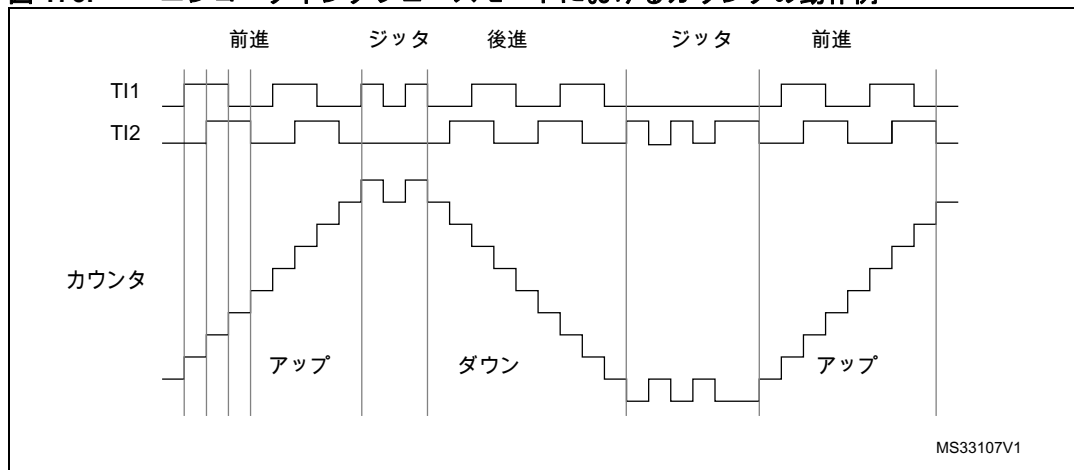
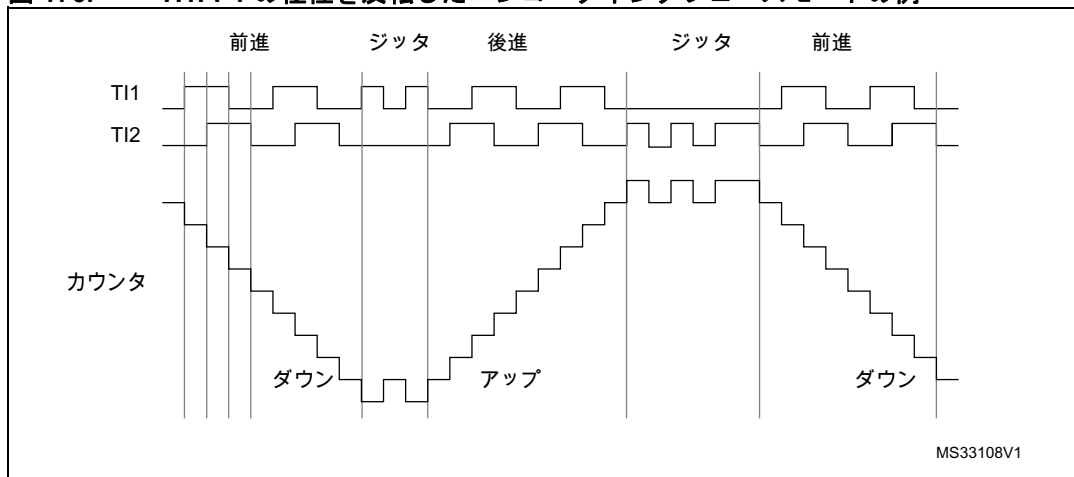


図 176 に、TI1FP1 の極性を反転したときのカウンタの動作例を示します (CC1P = "1" 以外は、上記と同じ設定)。

図 176. TI1FP1 の極性を反転したエンコーダインタフェースモードの例



タイマがエンコーダインタフェースモードに設定されている場合、タイマはセンサの現在位置に関する情報を提供します。キャプチャモードに設定された 2 番目のタイマを使用して、2 つのエンコーダイベントの時間差を測定することで、速度、加速度、減速度といった動的な情報を得ることができます。機械的なゼロ位置を示すエンコーダの出力をこの目的に使用できます。2 つのイベントの時間差に応じて、カウンタを定期的に読み出すこともできます。これを行うには、使用可能な場合は、カウンタの値を 3 番目の入力キャプチャレジスタにラッチします (キャプチャ信号は周期的でなければならない、別のタイマによって生成できます)。また、リアルタイムクロックによって生成される DMA リクエストを通じて、この値を読み出すこともできます。

22.3.13 TIM21/22 外部 トリガ同期

TIM21/22 タイマは、リセットモード、ゲートモード、およびトリガモードで外部トリガと同期を取ることができます。

スレーブモード：リセットモード

カウンタとそのプリスケアラは、トリガ入力のイベントに応じて再初期化できます。さらに、TIMx_CR1 レジスタの URS ビットがローの場合は、更新イベント UEV が生成されます。その場合、すべてのプリロードされたレジスタ (TIMx_ARR、TIMx_CCRx) が更新されます。

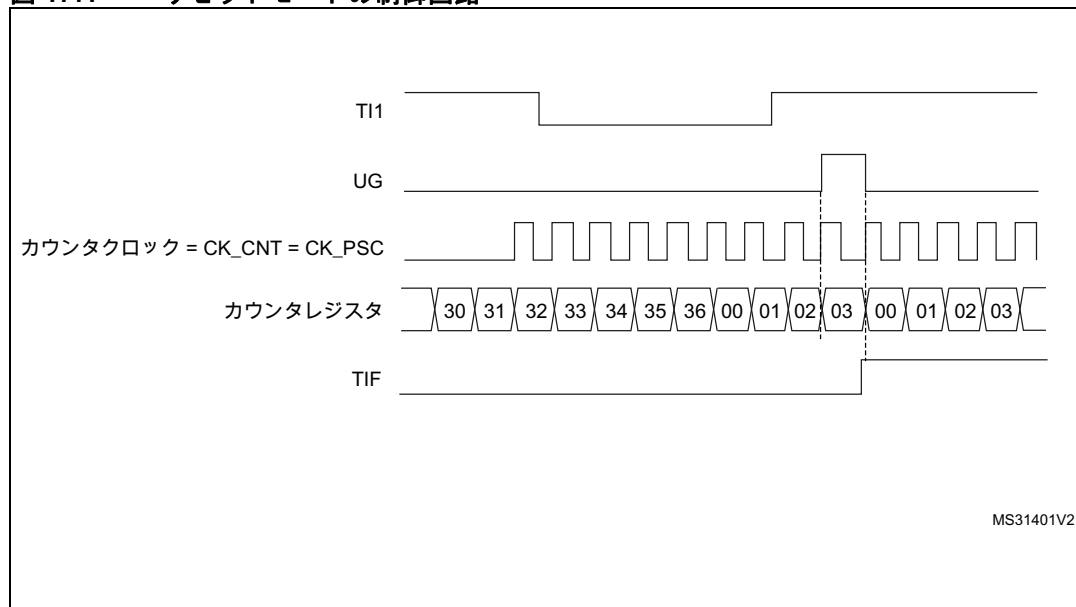
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

1. TI1 の立ち上がりエッジを検出するように、チャンネル 1 を設定します。入力フィルタ時間を設定します (この例ではフィルタは不要なので、IC1F = 0000 のままにしておく)。キャプチャプリスケアラはトリガには使用されないため、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します (TIMx_CCMR1 レジスタの CC1S = 01)。TIMx_CCER レジスタの CC1P と CC1NP に "00" を書き込んで、極性を有効にします (その後、立ち上がりエッジのみを検出)。
2. TIMx_SMCR レジスタの SMS ビットに "100" を書き込むことによって、タイマをリセットモードに設定します。TIMx_SMCR レジスタの TS ビットに "101" を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタの CEN ビットに "1" を書き込むことによって、カウンタを開始します。

カウンタは内部クロックでカウントを開始し、TI1 の立ち上がりエッジまで通常の動作を行います。TI1 が立ち上がると、カウンタはクリアされ、0 からリスタートします。同時に、トリガフラグがセットされ (TIMx_SR レジスタの TIF ビット)、(TIMx_DIER レジスタの TIE ビットが) 有効な場合は割り込みリクエストを送信できます。

次の図は、自動再ロードレジスタ TIMx_ARR = 0x36 の場合の動作を示します。TI1 の立ち上がりエッジから実際にカウンタがリセットされるまでの遅延は、TI1 入力の同期回路によるものです。

図 177. リセットモードの制御回路



スレーブモード：ゲートモード

選択された入力のレベルに応じて、カウンタを有効にできます。

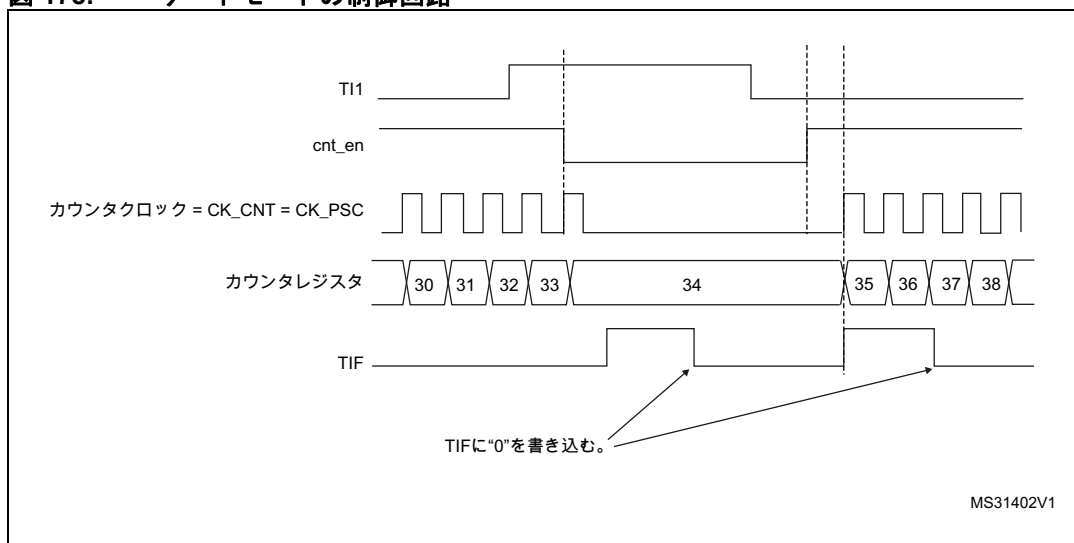
次の例では、アップカウンタは TI1 入力が高レベルのときだけカウントします。

1. TI1 のローレベルを検出するように、チャンネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F = 0000 のままにしておく）。キャプチャプリスケアラトリガには使用されないため、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します（TIMx_CCMR1 レジスタの CC1S = 01）。TIMx_CCER レジスタの CC1P ビットに“1”、CC1NP ビットに“0”を書き込んで、極性を有効にします（その後、立ち上がりエッジのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“101”を書き込むことによって、タイマをゲートモードに設定します。TIMx_SMCR レジスタの TS ビットに“101”を書き込むことによって、入力ソースとして TI1 を選択します。
3. TIMx_CR1 レジスタの CEN ビットに“1”を書き込んで、カウンタを有効にします（ゲートモードでは、CEN = 0 の場合、トリガ入力のレベルにかかわらず、カウンタは開始しません）。

カウンタは、TI1 がローになると内部クロックでカウントを開始して、TI1 がハイになると停止します。TIMx_SR レジスタの TIF フラグは、カウンタの開始時と停止時にセットされます。

TI1 の立ち上がりエッジから実際にカウンタが停止するまでの遅延は、TI1 入力の再同期回路によるものです。

図 178. ゲートモードの制御回路



スレーブモード：トリガモード

カウンタは、選択された入力のイベントに応じて開始できます。

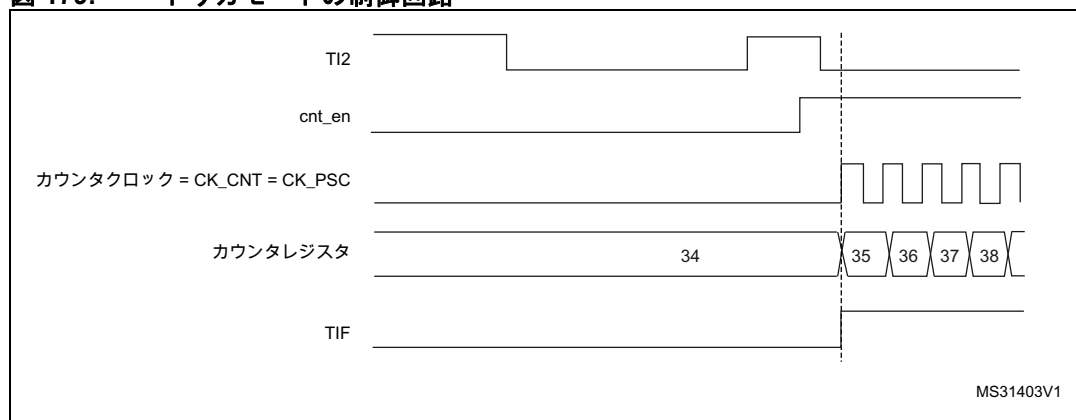
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

1. TIMx_SMCR レジスタで次のようにプログラムすることによって、外部トリガ入力回路を設定します。
 - ETF = 0000 : フィルタなし
 - ETPS = 00 : プリスケアラは無効
 - ETP = 0 : ETR の立ち上がりエッジを検出。ECE = 1 で外部クロックモード 2 を有効にする。
1. TI2 の立ち上がりエッジを検出するように、チャンネル 2 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC2F = 0000 のままにしておく）。キャプチャプリスケアラはトリガには使用されないの、設定は不要です。CC2S ビットは入力キャプチャソースのみを選択するように設定されます (TIMx_CCMR1 レジスタの CC2S = 01)。TIMx_CCER レジスタの CC2P ビットに“1”、CC2NP ビットに“0”を書き込んで、極性を有効にします（その後、ローレベルのみを検出）。
2. TIMx_SMCR レジスタの SMS ビットに“110”を書き込むことによって、タイマをトリガモードに設定します。TIMx_SMCR レジスタの TS ビットに“110”を書き込むことによって、入力ソースとして TI2 を選択します。

TI2 で立ち上がりエッジが発生すると、カウンタは内部クロックでのカウントを開始し、TIF フラグがセットされます。

TI2 の立ち上がりエッジから実際にカウンタが開始するまでの遅延は、TI2 入力の再同期回路によるものです。

図 179. トリガモードの制御回路



22.3.14 タイマ同期 (TIM21/22)

タイマは、タイマの同期または連結のために、内部で互いにリンクされます。詳細については、[セクション 21.3.15 : タイマの同期 \(444 ページ\)](#) を参照してください。

22.3.15 デバッグモード

マイクロコントローラがデバッグモードになると (Cortex®-M0+ コアは停止状態)、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、[セクション 33.9.2 : タイマ、ウォッチドッグ、および I2C のデバッグサポート](#) を参照してください。

22.4 TIM21/22 レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位で書き込む必要があります。読み出しアクセスは、バイト（8 ビット）、ハーフワード（16 ビット）またはワード（32 ビット）単位で行うことができます。

22.4.1 TIM21/22 制御レジスタ 1 (TIMx_CR1)

アドレスオフセット：0x00

リセット値：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|----------|----|------|----------|----|-----|-----|-----|------|-----|
| Res. | Res. | Res. | Res. | Res. | Res. | CKD[1:0] | | ARPE | CMS[1:0] | | DIR | OPM | URS | UDIS | CEN |
| | | | | | | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:10 予約済みであり、リセット値に保持する必要があります。

ビット 9:8 **CKD**：クロック分周

このビットフィールドは、タイマクロック（CK_INT）周波数と、デジタルフィルタ（TIX）によって使用されるサンプリングクロックとの間の分周比を示します。

00：t_{DTS} = t_{CK_INT}

01：t_{DTS} = t_{CK_INT}

10：t_{DTS} = t_{CK_INT}

11：予約済み

ビット 7 **ARPE**：自動再ロードプリロードイネーブル

0：TIMx_ARR レジスタはバッファされません。

1：TIMx_ARR レジスタはバッファされます。

ビット 6:5 **CMS[1:0]**：センターアラインモード選択

00：エッジアラインモード。カウンタは、方向ビット（DIR）に応じて、カウントアップまたはカウントダウンします。

01：センターアラインモード 1。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS = 00）の出力比較割り込みフラグは、カウンタがカウントダウンしているときのみセットされます。

10：センターアラインモード 2。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS = 00）の出力比較割り込みフラグは、カウンタがカウントアップしているときのみセットされます。

11：センターアラインモード 3。カウンタはカウントアップとカウントダウンを交互に行います。出力に設定されたチャネル（TIMx_CCMRx レジスタの CCxS = 00）の出力比較割り込みフラグは、カウンタがカウントアップおよびカウントダウンしているときにセットされます。

注： カウンタが有効（CEN = 1）なときに、エッジアラインモードからセンターアラインモードに切り替えることはできません。

ビット 4 **DIR**：方向

0：カウンタはアップカウンタとして使用されます。

1：カウンタはダウンカウンタとして使用されます。

ビット 3 **OPM**：ワンパルスモード

0：カウンタは更新イベントで停止しません。

1：カウンタは次の更新イベントでカウントを停止します（CEN ビットをクリア）。

ビット 2 URS : 更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアでセット／クリアされます。

0 : 次のイベントのいずれかが更新割り込みを生成します（有効な場合）。

- カウンタオーバーフロー
- UG ビットのセット

1 : カウンタオーバーフローのみが更新割り込みを生成します（有効な場合）。

ビット 1 UDIS : 更新ディセーブル

このビットは、更新イベント（UEV）生成を有効／無効にするために、ソフトウェアによってセット／クリアされます。

0 : UEV は有効です。UEV は、次のいずれかのイベントによって生成されます。

- カウンタオーバーフロー
- UG ビットのセット

パッファされたレジスタにはプリロード値がロードされます。

1 : UEV は無効です。UEV は生成されず、シャドウレジスタ（ARR、PSC、CCR_x）は値を維持します。カウンタとプリスケアラは、UG ビットがセットされた場合に再初期化されます。

ビット 0 CEN : カウンタイネーブル

0 : カウンタは無効です。

1 : カウンタは有効です。

ワンパルスモードでは、更新イベントが発生すると、CEN は自動的にクリアされます。

22.4.2 TIM21/22 制御レジスタ 2（TIMx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

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

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS**：マスタモード選択

これらのビットにより、同期のためにマスタモードでスレーブタイマに送信される情報を選択することができます（TRGO）。組み合わせは、次のとおりです。

000：リセット - TIMx_EGR レジスタの UG ビットがトリガ出力（TRGO）として使用されます。トリガ入力によってリセットが生成される場合（スレーブモードコントローラがリセットモードに設定されているとき）、TRGO 信号は実際のリセットより遅延します。

001：イネーブル - カウンタイネーブル信号 CNT_EN がトリガ出力（TRGO）として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和（OR）によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ／スレーブモードが選択されている場合を除き、TRGO には遅延が存在します（TIMx_SMCR レジスタの MSM ビットの説明を参照）。

010：更新 - 更新イベントがトリガ出力（TRGO）として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケアラとして使用できます。

011：パルス比較 - キャプチャまたは比較一致が発生すると、CC1IF フラグがセットされるとき（すでにハイであった場合も）、トリガ出力は正のパルスを送信します。（TRGO）

100：比較 - OC1REF 信号がトリガ出力（TRGO）として使用されます。

101：比較 - OC2REF 信号がトリガ出力（TRGO）として使用されます。

110：予約済み

111：予約済み

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

22.4.3 TIM21/22 のスレーブモード制御レジスタ (TIMx_SMCR)

アドレスオフセット : 0x08

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|-----|-----|-----------|-----|----------|-----|-----|-----|-----|---------|-----|-----|------|----------|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ETP | ECE | ETPS[1:0] | | ETF[3:0] | | | | MSM | TS[2:0] | | | Res. | SMS[2:0] | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | r/w | r/w | r/w |

ビット 15 **ETP** : 外部トリガ極性

- このビットは、ETR または $\overline{\text{ETR}}$ がトリガ動作に使用されるかを選択します。
- 0 : ETR は反転されず、ハイレベルまたは立ち上がりエッジでアクティブです。
- 1 : ETR は反転され、ローレベルまたは立ち下がりエッジでアクティブです。

ビット 14 **ECE** : 外部クロックイネーブル

- このビットは、外部クロックモード 2 を有効にします。
- 0 : 外部クロックモード 2 は無効です。
- 1 : 外部クロックモード 2 は有効です。カウンタのクロックは、ETRF 信号のアクティブエッジによって供給されます。

注 : *ECE* ビットをセットすることは、*TRGI* が *ETRF* に接続された状態で外部クロックモード 1 を選択するすることと同じ効果があります (*SMS* = 111、*TS* = 111)。

外部クロックモード 2 とスレーブモード (すなわち、リセットモード、ゲートモード、またはトリガモード) を同時に使用することができます。ただし、この場合、TRGI を ETRF に接続することはできません (TS ビットが 111 でないことが必要)。

外部クロックモード 1 と外部クロックモード 2 が同時に有効な場合、外部クロック入力 は ETRF です。

ビット 13:12 **ETPS[1:0]** : 外部トリガプリスケアラ

- 外部トリガ信号 ETRP の周波数は、TIMxCLK 周波数の 1/4 までに制限されます。プリスケアラを有効にすると、ETRP 周波数を低減できます。これは、高速な外部クロックを入力するときに役立ちます。
- 00 : プリスケアラはオフです。
- 01 : ETRP 周波数は 2 分周されます。
- 10 : ETRP 周波数は 4 分周されます。
- 11 : ETRP 周波数は 8 分周されます。

ビット 11:8 ETF[3:0] : 外部トリガフィルタ

このビットフィールドは、ETRP 信号をサンプリングする周波数と、ETRP に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な遷移をもたらすには N 個のイベント発生が必要です。

0000 : フィルタなし、サンプリングは f_{DTS} で行われます。

0001 : $f_{SAMPLING} = f_{CK_INT}$ 、 $N = 2$

0010 : $f_{SAMPLING} = f_{CK_INT}$ 、 $N = 4$

0011 : $f_{SAMPLING} = f_{CK_INT}$ 、 $N = 8$

0100 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 6$

0101 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 8$

0110 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 6$

0111 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 8$

1000 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 6$

1001 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 8$

1010 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 16$

1011 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 16$

1100 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 16$

1101 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 32$

1110 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 32$

1111 : $f_{SAMPLING} = f_{DTS}$ 、 $N = 32$

ビット 7 MSM : マスタ/スレーブモード

0 : 影響なし。

1 : トリガ入力 (TRGI) に対するイベントの影響は、現在のタイマとそのスレーブとの間の完全な同期 (TRGO を通じて) を可能にするために遅延されます。これは、1 つの外部イベントで複数のタイマを同期する場合に役立ちます。

ビット 6:4 TS : トリガ選択

このビットフィールドは、カウンタの同期に使用されるトリガ入力を選択します。

- 000 : 内部トリガ 0 (ITR0)
- 001 : 内部トリガ 1 (ITR1)
- 010 : 予約済み
- 011 : 予約済み
- 100 : TI1 エッジ検出回路 (TI1F_ED)
- 101 : フィルタタイマ入力 1 (TI1FP1)
- 110 : フィルタタイマ入力 2 (TI2FP2)
- 111 : 予約済み

各タイマにとっての ITRx の意味の詳細については、表 84 : TIMx 内部トリガ接続 (513 ページ) を参照してください。

注: 遷移時の誤ったエッジ検出を避けるために、これらのビットは必ず使用されていないとき (SMS = 000 のときなど) に変更してください。

ビット 3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 SMS : スレーブモード選択

外部信号が選択されると、トリガ信号 (TRGI) のアクティブエッジが外部入力で選択された極性にリンクされます (入力制御レジスタおよび制御レジスタの説明を参照してください)。

- 000 : スレーブモードは無効です。GEN = 1 の場合、プリスケアラは内部クロックによって直接クロック供給されます。
- 001 : 予約済み
- 010 : 予約済み
- 011 : 予約済み
- 100 : リセットモード - 選択されたトリガ入力 (TRGI) の立ち上がりエッジでカウンタを再初期化し、レジスタの更新を生成します。
- 101 : ゲートモード - カウンタクロックは、トリガ入力 (TRGI) がハイのときに有効になります。トリガがローになると、カウンタは停止します (リセットはされません)。カウンタの開始と停止の両方が制御されます。
- 110 : トリガモード - カウンタは、トリガ TRGI の立ち上がりエッジで開始します (リセットはされません)。カウンタの開始のみが制御されます。
- 111 : 予約済み

注: トリガ入力として TI1F_ED が選択されている場合 (TS = 100)、ゲートモードを使用することはできません。TI1F_ED は TI1F の遷移ごとに 1 パルスを出力しますが、ゲートモードはトリガ信号のレベルをチェックします。

表 84. TIMx 内部トリガ接続⁽¹⁾

| スレーブ TIM | ITR0 (TS = 000) | ITR1 (TS = 001) |
|----------|-----------------|-----------------|
| TIM21 | TIM2 | TIM22 |
| TIM22 | TIM21 | TIM2 |

1. 製品にタイマが存在しない場合、対応するトリガ ITRx は使用できません。

22.4.4 TIM21/22 割り込み有効レジスタ (TIMx_DIER)

アドレスオフセット : 0x0C

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|-----|------|------|------|-------|-------|-----|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TIE | Res. | Res. | Res. | CC2IE | CC1IE | UIE |
| | | | | | | | | | r/w | | | | r/w | r/w | r/w |

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 **TIE** : トリガ割り込みイネーブル

0 : トリガ割り込みは無効です。

1 : トリガ割り込みは有効です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **CC2IE** : キャプチャ/比較 2 割り込みイネーブル

0 : CC2 割り込みは無効です。

1 : CC2 割り込みは有効です。

ビット 1 **CC1IE** : キャプチャ /比較 1 割り込みイネーブル

0 : CC1 割り込みは無効です。

1 : CC1 割り込みは有効です。

ビット 0 **UIE** : 更新割り込みイネーブル

0 : 更新割り込みは無効です。

1 : 更新割り込みは有効です。

22.4.5 TIM21/22 のステータスレジスタ (TIMx_SR)

アドレスオフセット : 0x10

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|-------|-------|------|------|-------|------|------|------|-------|-------|-------|
| Res. | Res. | Res. | Res. | Res. | CC2OF | CC1OF | Res. | Res. | TIF | Res. | Res. | Res. | CC2IF | CC1IF | UIF |
| | | | | | rc_w0 | rc_w0 | | | rc_w0 | | | | rc_w0 | rc_w0 | rc_w0 |

ビット 15:11 予約済みであり、リセット値に保持する必要があります。

ビット 10 **CC2OF** : キャプチャ/比較 2 オーバーキャプチャフラグ

CC1OF の説明を参照してください。

ビット 9 **CC1OF** : キャプチャ/比較 1 オーバーキャプチャフラグ

このフラグは、対応するチャネルが入力キャプチャモードに設定されているときのみ、ハードウェアによってセットされます。“0”を書き込むことによってソフトウェアによってクリアされます。

0 : オーバーキャプチャは検出されていません。

1 : CC1IF フラグがすでにセットされているときに、カウンタの値が TIMx_CCR1 レジスタにキャプチャされました。

ビット 8:7 予約済みであり、リセット値に保持する必要があります。

ビット 6 TIF : トリガ割り込みフラグ

このフラグは、トリガイイベント時（スレーブモードコントローラがゲートモード以外のすべてのモードで有効なときに、TRGI 入力でアクティブエッジが検出されたとき）にハードウェアによってセットされます。ゲートモードが選択されている場合、カウンタが開始または停止したときにセットされます。ソフトウェアによってクリアされます。

0 : トリガイイベントは発生していません。

1 : トリガ割り込みが保留中です。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 CC2IF : キャプチャ／比較 2 割り込みフラグ

CC1IF の説明を参照してください。

ビット 1 CC1IF : キャプチャ／比較 1 割り込みフラグ**チャンネル CC1 が出力として設定されている場合**

このフラグは、カウンタが比較値と一致したときに、ハードウェアによってセットされます。ソフトウェアによってクリアされます。

0 : 一致していません。

1 : カウンタ TIMx_CNT の内容が TIMx_CCR1 レジスタの内容と一致しました。TIMx_CCR1 の内容が TIMx_ARR の内容より大きいときには、カウンタオーバーフロー時に CC1IF ビットがハイになります。

チャンネル CC1 が入力として設定されている場合

このビットは、キャプチャ時にハードウェアによってセットされます。ソフトウェアによって、または TIMx_CCR1 レジスタを読み出すことによってクリアされます。

0 : 入力キャプチャは発生していません。

1 : カウンタの値が TIMx_CCR1 レジスタにキャプチャされました（選択された極性に一致するエッジが IC1 で検出されました）。

ビット 0 UIF : 更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- オーバーフローが発生し、かつ TIMx_CR1 レジスタで UDIS = 0 の場合。
- TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、TIMx_EGR レジスタの UG ビットを使用して、CNT がソフトウェアによって再初期化されたとき。
- TIMx_CR1 レジスタで URS = 0 かつ UDIS = 0 であり、トリガイイベントによって CNT が再初期化されたとき（同期制御レジスタの説明を参照）。

22.4.6 TIM21/22 のイベント生成レジスタ（TIMx_EGR）

アドレスオフセット：0x14

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|----|------|------|------|------|------|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TG | Res. | Res. | Res. | CC2G | CC1G | UG |
| | | | | | | | | | w | | | | w | w | w |

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

- ビット 6 **TG**：トリガ生成
- このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
- 0：影響なし。
- 1：TIMx_SR レジスタの TIF フラグがセットされます。有効な場合は、関連する割り込が発生します。

ビット 5:3 予約済みであり、リセット値に保持する必要があります。

- ビット 2 **CC2G**：キャプチャ／比較 2 生成
- CC1G の説明を参照してください。

- ビット 1 **CC1G**：キャプチャ／比較 1 生成
- このビットは、イベントを生成するためにソフトウェアによってセットされ、ハードウェアによって自動的にクリアされます。
- 0：影響なし。
- 1：チャンネル 1 でキャプチャ／比較イベントが生成されます。
- チャンネル CC1 が出力として設定されている場合**
- CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。
- チャンネル CC1 が入力として設定されている場合**
- 現在のカウンタ値が TIMx_CCR1 レジスタにキャプチャされます。CC1IF フラグがセットされ、対応する割り込みが送信されます（有効な場合）。CC1IF フラグがすでにハイの場合、CC1OF フラグがセットされます。

- ビット 0 **UG**：更新生成
- このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。
- 0：影響なし。
- 1：カウンタを再初期化し、レジスタの更新を生成します。プリスケアラのカウンタもクリアされます（分周比は変化しません）。カウンタはクリアされます。

22.4.7 TIM21/22 のキャプチャ／比較モードレジスタ 1 (TIMx_CCMR1)

アドレスオフセット : 0x18

リセット値 : 0x0000

チャンネルは、入力（キャプチャモード）または出力（比較モード）で使用できます。チャンネルの方向は対応する CCxS ビットを設定することによって定義されます。このレジスタの他のすべてのビットは、入力モードと出力モードで機能が異なります。特定のビットについて、OCxx は、チャンネルが出力モードに設定されているときの機能を記述し、ICxx は、チャンネルが入力モードに設定されているときの機能を記述します。したがって、同じビットが入力ステージと出力ステージで異なる意味を持つことに注意してください。

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|-----------|----|----|-------------|-------|-----------|----|------|-----------|----|----|-------------|-------|-----------|----|
| Res. | OC2M[2:0] | | | OC2PE | OC2FE | CC2S[1:0] | | Res. | OC1M[2:0] | | | OC1PE | OC1FE | CC1S[1:0] | |
| | IC2F[3:0] | | | IC2PSC[1:0] | | | | | IC1F[3:0] | | | IC1PSC[1:0] | | | |
| | rw | rw | rw | rw | rw | rw | rw | | rw | rw | rw | rw | rw | rw | rw |

出力比較モード

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **OC2M[2:0]** : 出力比較 2 モード

ビット 11 **OC2PE** : 出力比較 2 プリロードイネーブル

ビット 10 **OC2FE** : 出力比較 2 高速イネーブル

ビット 9:8 **CC2S[1:0]** : キャプチャ／比較 2 選択

このビットフィールドは、チャンネルの方向（入力／出力）と、使用される入力を定義します。

00 : CC2 チャンネルは出力として設定されます。

01 : CC2 チャンネルは入力として設定され、IC2 は TI2 に配置されます。

10 : CC2 チャンネルは入力として設定され、IC2 は TI1 に配置されます。

11 : CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC2S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC2E = 0) のときにのみ書き込み可能です。**

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 OC1M : 出力比較 1 モード

これらのビットは、OC1 および OC1N が導き出される出力基準信号 OC1REF の動作を定義します。OC1REF はアクティブハイですが、OC1 および OC1N のアクティブレベルは、それぞれ CC1P および CC1NP ビットに依存します。

000 : 停止 - 出力比較レジスタ TIMx_CCR1 とカウンタ TIMx_CNT の比較結果は出力に影響しません (このモードはタイミングベースを生成するために使用)。

001 : 一致時にチャンネル 1 をアクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ/比較レジスタ 1 (TIMx_CCR1) に一致したとき、強制的にハイになります。

010 : 一致時にチャンネル 1 をインアクティブレベルに設定します。OC1REF 信号は、TIMx_CNT カウンタがキャプチャ/比較レジスタ 1 (TIMx_CCR1) に一致したとき、強制的にローになります。

011 : 反転 - TIMx_CNT = TIMx_CCR1 のとき、OC1REF は反転します。

100 : 強制インアクティブレベル - OC1REF は強制的にローになります。

101 : 強制アクティブレベル - OC1REF は強制的にハイになります。

110 : PWM モード 1 - カウントアップ時、チャンネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。カウントダウン時、チャンネル 1 は、TIMx_CNT > TIMx_CCR1 の場合はインアクティブ (OC1REF = 0) に、そうでない場合はアクティブ (OC1REF = 1) になります。

111 : PWM モード 2 - カウントアップ時、チャンネル 1 は、TIMx_CNT < TIMx_CCR1 の場合はインアクティブに、そうでない場合はアクティブになります。カウントダウン時、チャンネル 1 は、TIMx_CNT > TIMx_CCR1 の場合はアクティブに、そうでない場合はインアクティブになります。

注 : PWM モード 1 または 2 では、比較結果が変化したとき、または出力比較モードが停止モードから PWM モードに変更されたときにのみ、OCREF のレベルが変化します。

ビット 3 OC1PE : 出力比較 1 プリロードイネーブル

0 : TIMx_CCR1 のプリロードレジスタは無効です。TIMx_CCR1 は、いつでも書き込み可能であり、新しい値はただちに有効になります。

1 : TIMx_CCR1 のプリロードレジスタは有効です。読み書きはプリロードレジスタに対して行われます。TIMx_CCR1 プリロード値は、更新イベントのたびにアクティブレジスタにロードされます。

注 : PWM モードは、ワンパルスモード (TIMx_CR1 レジスタの OPM ビットがセットされている) のときのみ、プリロードレジスタを検証せずに使用できます。そうでない場合、動作は保証されません。

ビット 2 OC1FE : 出力比較 1 高速イネーブル

このビットは、CC 出力に対するトリガ入力のイベントの影響を加速させるために使用されます。

0 : CC1 は、トリガがオンのときでも、カウンタと CCR1 の値に依存して、通常どおりに動作します。トリガ入力にエッジが発生してから CC1 出力が有効になるまでの最小遅延は、5 クロックサイクルです。

1 : トリガ入力のアクティブエッジは、CC1 出力に対して、比較一致のように働きます。このため、OC は、比較結果には関係なく、比較レベルにセットされます。トリガ入力をサンプリングし、CC1 出力を有効にするまでの遅延は、3 クロックサイクルに短縮されます。OC1FE は、チャンネルが PWM1 または PWM2 モードに設定されている場合のみ機能します。

ビット 1:0 CC1S : キャプチャ/比較 1 選択

このビットフィールドは、チャンネルの方向 (入力/出力) と、使用される入力を定義します。

00 : CC1 チャンネルは出力として設定されます。

01 : CC1 チャンネルは入力として設定され、IC1 は TI1 に配置されます。

10 : CC1 チャンネルは入力として設定され、IC1 は TI2 に配置されます。

11 : CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : CC1S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC1E = 0) のときにのみ書き込み可能です。

入力キャプチャモード

ビット 15:12 **IC2F** : 入力キャプチャ 2 フィルタ

ビット 11:10 **IC2PSC[1:0]** : 入力キャプチャ 2 プリスケアラ

ビット 9:8 **CC2S** : キャプチャ/比較 2 選択

このビットフィールドは、チャンネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC2 チャンネルは出力として設定されます。

01 : CC2 チャンネルは入力として設定され、IC2 は TI2 に配置されます。

10 : CC2 チャンネルは入力として設定され、IC2 は TI1 に配置されます。

11 : CC2 チャンネルは入力として設定され、IC2 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC2S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC2E = 0) のときにのみ書き込み可能です。**

ビット 7:4 **IC1F** : 入力キャプチャ 1 フィルタ

このビットフィールドは、TI1 入力をサンプリングする周波数と、TI1 に適用されるデジタルフィルタの長さを定義します。デジタルフィルタはイベントカウンタでできていて、出力に有効な遷移をもたらすには N 個のイベント発生が必要です。

0000 : フィルタなし、サンプリングは f_{DTS} で行われます。 1000 : $f_{SAMPLING} = f_{DTS}/8$, $N = 6$

0001 : $f_{SAMPLING} = f_{CK_INT}$, $N = 2$ 1001 : $f_{SAMPLING} = f_{DTS}/8$, $N = 8$

0010 : $f_{SAMPLING} = f_{CK_INT}$, $N = 4$ 1010 : $f_{SAMPLING} = f_{DTS}/16$, $N = 5$

0011 : $f_{SAMPLING} = f_{CK_INT}$, $N = 8$ 1011 : $f_{SAMPLING} = f_{DTS}/16$, $N = 6$

0100 : $f_{SAMPLING} = f_{DTS}$, $N = 6$ 1100 : $f/2 < 256 f_{SAMPLING} = f_{DTS}/16$, $N = 8$

0101 : $f_{SAMPLING} = f_{DTS}$, $N = 8$ 1101 : $f/2 < 256 f_{SAMPLING} = f_{DTS}/32$, $N = 5$

0110 : $f_{SAMPLING} = f_{DTS}$, $N = 6$ 1110 : $f/4 < 257 f_{SAMPLING} = f_{DTS}/32$, $N = 6$

0111 : $f_{SAMPLING} = f_{DTS}$, $N = 8$ 1111 : $f/4 < 257 f_{SAMPLING} = f_{DTS}/32$, $N = 8$

注 : **現在のシリコンのリビジョンでは、ICxF[3:0] = 1、2、または3のときに、上の式の f_{DTS} が f_{CK_INT} に置き換えられます。**

ビット 3:2 **IC1PSC** : 入力キャプチャ 1 プリスケアラ

このビットフィールドは、CC1 入力 (IC1) に作用するプリスケアラの比を定義します。

プリスケアラは、CC1E = 0 (TIMx_CCER レジスタ) になるとリセットされます。

00 : プリスケアラなし。キャプチャは、キャプチャ入力のエッジが検出されるたびに行われます。

01 : キャプチャは、2 イベントごとに行われます。

10 : キャプチャは、4 イベントごとに行われます。

11 : キャプチャは、8 イベントごとに行われます。

ビット 1:0 **CC1S** : キャプチャ/比較 1 選択

このビットフィールドは、チャンネルの方向（入力/出力）と、使用される入力を定義します。

00 : CC1 チャンネルは出力として設定されます。

01 : CC1 チャンネルは入力として設定され、IC1 は TI1 に配置されます。

10 : CC1 チャンネルは入力として設定され、IC1 は TI2 に配置されます。

11 : CC1 チャンネルは入力として設定され、IC1 は TRC に配置されます。このモードは、TS ビット (TIMx_SMCR レジスタ) で内部トリガ入力を選択されている場合のみ機能します。

注 : **CC1S ビットは、チャンネルがオフ (TIMx_CCER レジスタの CC1E = 0) のときにのみ書き込み可能です。**

22.4.8 TIM21/22 のキャプチャ／比較有効レジスタ (TIMx_CCER)

アドレスオフセット : 0x20

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|-------|------|------|------|-------|------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CC2NP | Res. | CC2P | CC2E | CC1NP | Res. | CC1P | CC1E |
| | | | | | | | | rW | | rW | rW | rW | | rW | rW |

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **CC2NP** : キャプチャ／比較 2 出力極性
CC1NP の説明を参照してください。

ビット 6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **CC2P** : キャプチャ／比較 2 出力極性
CC1P の説明を参照してください。

ビット 4 **CC2E** : キャプチャ／比較 2 出力イネーブル
CC1E の説明を参照してください。

ビット 3 **CC1NP** : キャプチャ／比較 1 相補出力極性
CC1 チャンネルが出力として設定されている場合 : CC1NP はクリア状態に維持する必要があります。
CC1 チャンネルが入力として設定されている場合 : CC1NP ビットは、TI1FP1/TI2FP1 の極性を定義するために CC1P と組み合わせて使用されます (CC1P の説明を参照)。

ビット 2 予約済みであり、リセット値に保持する必要があります。

ビット 1 **CC1P** : キャプチャ／比較 1 出力極性。
CC1 チャンネルが出力として設定されている場合
0 : OC1 はアクティブハイです。
1 : OC1 はアクティブローです。
CC1 チャンネルが入力として設定されている場合
CC1NP/CC1P ビットは、トリガまたはキャプチャ操作の TI1FP1 および TI2FP1 の極性を選択します。
00 : 非反転／立ち上がりエッジ
回路は TIxFP1 の立ち上がりエッジに反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されません (ゲートモードでのトリガ、エンコーダモード)。
01 : 反転／立ち下がりエッジ
回路は TIxFP1 の立ち下がりエッジに反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されます (ゲートモードでのトリガ、エンコーダモード)。
10 : 予約済み。この設定は使用しないでください。

注 : 11 : 非反転／両エッジ
回路は TIxFP1 の立ち上がりエッジと立ち下がりエッジの両方に反応し (キャプチャモード、リセットモードでのトリガ、外部クロックモード、またはトリガモード)、TIxFP1 は反転されません (ゲートモードでのトリガ)。この設定をエンコーダモードに使用することはできません。

ビット 0 **CC1E** : キャプチャ／比較 1 出力イネーブル。
CC1 チャンネルが出力として設定されている場合
0 : オフ - OC1 はアクティブではありません。
1 : オン - OC1 信号は、対応する出力ピンに出力されます。
CC1 チャンネルが入力として設定されている場合
このビットによって、キャプチャ／比較レジスタ 1 (TIMx_CCR1) へのカウンタ値のキャプチャが実際に行われるかどうかが決まります。
0 : キャプチャは無効です。
1 : キャプチャは有効です。

表 85. 標準 OCx チャンネルの出力制御ビット

| CCxE ビット | OCx 出力状態 |
|----------|------------------------------|
| 0 | 出力無効 (OCx = 0、OCx_EN = 0) |
| 1 | OCx = OCxREF + 極性、OCx_EN = 1 |

注：標準 OCx チャンネルに接続されている外部入出力ピンの状態は、OCx チャンネルの状態と、GPIO レジスタに依存します。

22.4.9 TIM21/22 のカウンタ (TIMx_CNT)

アドレスオフセット：0x24

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CNT[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 CNT[15:0]：カウンタ値

22.4.10 TIM21/22 のプリスケアラ (TIMx_PSC)

アドレスオフセット：0x28

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PSC[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 PSC[15:0]：プリスケアラ値

カウンタクロック周波数 CK_CNT は、 $f_{CK_PSC} / (PSC[15:0] + 1)$ に等しいです。

PSC は、更新イベントごとにアクティブなプリスケアラレジスタにロードされる値を含みます。

22.4.11 TIM21/22 の自動再ロードレジスタ (TIMx_ARR)

アドレスオフセット：0x2C

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ARR[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 ARR[15:0]：自動再ロード値

ARR は、実際の自動再ロードレジスタにロードされる値です。

APR の更新と動作の詳細については、[セクション 22.3.1: タイムベースユニット \(475 ページ\)](#) を参照してください。

自動再ロード値が null のときには、カウンタはブロックされます。



22.4.12 TIM21/22 のキャプチャ／比較レジスタ 1（TIMx_CCR1）

アドレスオフセット：0x34

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR1[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 CCR1[15:0]：キャプチャ／比較 1 値

チャンネル CC1 が出力として設定されている場合
CCR1 は、実際のキャプチャ／比較 1 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR1 レジスタの OC1PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 1 レジスタにコピーされます。
アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC1 出力に送信される値を含みます。
チャンネル CC1 が入力として設定されている場合
CCR1 は、最後の入力キャプチャ 1 イベント（IC1）によって転送されたカウンタ値です。

22.4.13 TIM21/22 のキャプチャ／比較レジスタ 2（TIMx_CCR2）

アドレスオフセット：0x38

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CCR2[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 CCR2[15:0]：キャプチャ／比較 2 値

チャンネル CC2 が出力として設定されている場合
CCR2 は、実際のキャプチャ／比較 2 レジスタにロードされる値（プリロード値）です。
TIMx_CCMR2 レジスタの OC2PE ビットでプリロード機能が選択されていない場合、プリロード値がロードされたままになります。そうでない場合、プリロード値は、更新イベントが発生すると、アクティブキャプチャ／比較 2 レジスタにコピーされます。
アクティブなキャプチャ／比較レジスタは、カウンタ TIMx_CNT と比較されて、OC2 出力に送信される値を含みます。
チャンネル CC2 が入力として設定されている場合
CCR2 は、最後の入力キャプチャ 2 イベント（IC2）によって転送されたカウンタ値です。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

22.4.14 TIM21 オプションレジスタ（TIM21_OR）

アドレスオフセット：0x50

リセット値：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|------|---------|---------|----|----|---------|----|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI2_RMP | TI1_RMP | | | ETR_RMP | |
| | | | | | | | | | | rw | rw | rw | rw | rw | rw |

ビット 15:6 予約済みであり、リセット値に保持する必要があります。

ビット 5 **TI2_RMP** : Timer21 TI2（TIM21_CH1 に接続）の再配置

このビットはソフトウェアによってセット／クリアされます。

0 : TIM21 TI2 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

1 : TIM21 TI2 入力は COMP2_OUT に接続されます。

ビット 4:2 **TI1_RMP** : Timer21 TI1（TIM21_CH1 に接続）の再配置

このビットはソフトウェアによってセット／クリアされます。

000 : TIM21 TI1 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

001 : TIM21 TI1 入力は RTC ウェイクアップ割り込みに接続されます。

010 : TIM21 TI1 入力は HSE_RTC クロックに接続されます。

011 : TIM21 TI1 入力は MSI クロックに接続されます。

100 : TIM21 TI1 入力は LSE クロックに接続されます。

101 : TIM21 TI1 入力は LSI クロックに接続されます。

110 : TIM21 TI1 入力は COMP1_OUT に接続されます。

111 : TIM21 TI1 入力は MCO クロックに接続されます。

ビット 1:0 **ETR_RMP** : Timer21 ETR の再配置

このビットはソフトウェアによってセット／クリアされます。

00 : TIM21 ETR 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

01 : TIM21 ETR 入力は COMP2_OUT に接続されます。

10 : TIM21 ETR 入力は COMP1_OUT に接続されます。

11 : TIM21 ETR 入力は LSE クロックに接続されます。

22.4.15 TIM22 オプションレジスタ (TIM22_OR)

アドレスオフセット : 0x50

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|------|------|------|---------|----|---------|----|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI1_RMP | | ETR_RMP | |
| | | | | | | | | | | | | rw | rw | rw | rw |

ビット 15:4 予約済みであり、リセット値に保持する必要があります。

ビット 3:2 **TI1_RMP** : Timer22 TI1 (TIM22_CH1 に接続) の再配置

このビットはソフトウェアによってセット／クリアされます。

00 : TIM22 TI1 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

01 : TIM22 TI1 入力は COMP2_OUT に接続されます。

10 : TIM22 TI1 入力は COMP1_OUT に接続されます。

11 : TIM22 TI1 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

ビット 1:0 **ETR_RMP** : Timer22 ETR の再配置

このビットはソフトウェアによってセット／クリアされます。

00 : TIM22 ETR 入力は GPIO に接続されます。デバイスデータシートのオルタネート機能配置表を参照してください。

01 : TIM22 ETR 入力は COMP2_OUT に接続されます。

10 : TIM22 ETR 入力は COMP1_OUT に接続されます。

11 : TIM22 ETR 入力は LSE クロックに接続されます。

22.4.16 TIM21/22 のレジスタマップ

次の表に、TIM21/22 のレジスタマップとリセット値を示します。

表 86. TIM21/22 レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----------|-----------|------|------|-----------|------|----------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|--------------|-------|------|--------------|------|------|------|----------|-------|-------|-----|
| 0x00 | TIMx_CR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CKD [1:0] | | ARPE | CMS [1:0] | | | DIR | OPM | URS | UDIS | CEN |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | |
| 0x04 | TIMx_CR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | | MMS[2:0] | | | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | | | |
| 0x08 | TIMx_SMCR | ETP | ECE | ETPS[1:0] | | ETF[3:0] | | | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MSM | TS[2:0] | | | Res. | SMS[2:0] | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | |
| 0x0C | TIMx_DIER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TIE | Res. | Res. | Res. | Res. | CC2IE | CC1IE | UIE | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | 0 | 0 | 0 | |
| 0x10 | TIMx_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CC2OF | CC1OF | Res. | TIF | Res. | Res. | Res. | Res. | CC2IF | CC1IF | UIF |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | 0 | | | | 0 | 0 | 0 | |

表 86. TIM21/22 レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------|------------------------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------------|---------------|---------------------|---------------------|---------------------|---------------|---------------|---------------|---------------------|---------------------|---------------------|---------------|-------|---------------|------|------|
| 0x14 | TIMx_EGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TG | Res. | Res. | Res. | Res. | CC2G | CC1G | UG |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | 0 | 0 | |
| 0x18 | TIMx_CCMR1 出力比較モード | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OC2M [2:0] | OC2M [2:0] | OC2PE | OC2FE | OC2FE | OC2FE | CC2S [1:0] | Res. | OC1M [2:0] | OC1PE | OC1FE | OC1FE | CC1S [1:0] | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | TIMx_CCMR1 入力キャプチャ モード | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | IC2F[3:0] | IC2 PSC [1:0] | IC2 PSC [1:0] | IC2 PSC [1:0] | CC2S [1:0] | CC2S [1:0] | IC1F[3:0] | IC1 PSC [1:0] | IC1 PSC [1:0] | IC1 PSC [1:0] | CC1S [1:0] | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x1C | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x20 | TIMx_CCER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CC2NP | Res. | CC2P | CC2E | CC1NP | Res. | CC1P | CC1F |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | 0 | 0 | 0 | | 0 | 0 |
| 0x24 | TIMx_CNT | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x28 | TIMx_PSC | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PSC[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x2C | TIMx_ARR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ARR[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x30 | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x34 | TIMx_CCR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CCR1[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x38 | TIMx_CCR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CCR2[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x3C～ 0x4C | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x38 | TIMx_CCR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CCR2[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

表 86. TIM21/22 レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|----------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---------|---------|---|---------|---|---|
| 0x50 | TIM21_OR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI2_RMP | TI1_RMP | | ETR_RMP | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x50 | TIM22_OR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TI1_RMP | | ETR_RMP | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

23 基本タイマ (TIM6)

23.1 概要

基本タイマ TIM6 は、プログラム可能なプリスケアラで駆動される 16 ビット自動再ロードカウンタで構成されます。

このタイマは、タイムベース生成を目的とした汎用タイマとして使用できますが、特にデジタルアナログコンバータ (DAC) の駆動にも使用されます。実際、このタイマは 内部で DAC に接続されており、トリガ出力を通じて DAC を駆動できます。

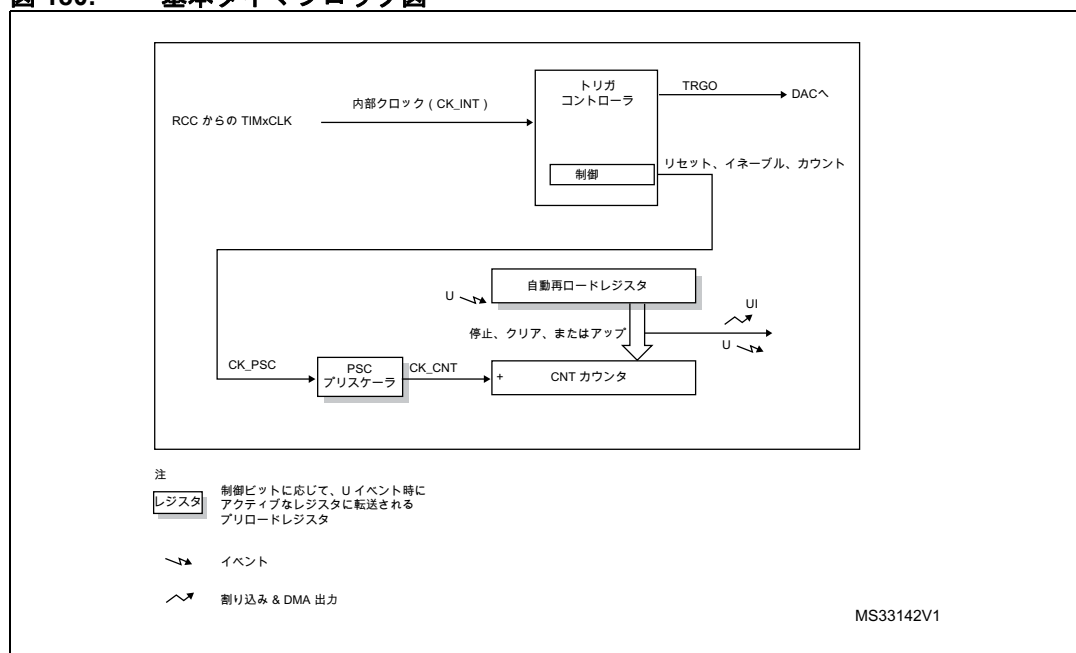
タイマは完全に独立していて、いかなるリソースも共有しません。

23.2 TIM6 の主な機能

基本タイマ (TIM6) の機能は、次のとおりです。

- 16 ビット自動再ロードアップカウンタ
- 16 ビットのプログラム可能なプリスケアラ (動作中も変更可能) で、カウンタクロック周波数を 1 から 65536 の間の値で分周可能。
- DAC をトリガする同期回路
- 更新イベント時の割り込み/DMA 生成: カウンタオーバーフロー

図 180. 基本タイマブロック図



23.3 TIM6 の機能説明

23.3.1 タイムベースユニット

プログラム可能なタイマのメインブロックは、自動再ロードレジスタを持つ 16 ビットアップカウンタです。カウンタのクロックは、プリスケアラで分周できます。

カウンタ、自動再ロードレジスタ、およびプリスケアラレジスタは、ソフトウェアで読み書きができます。カウンタが動作中でも、読み書きが可能です。

タイムベースユニットは、次のレジスタで構成されます。

- カウンタレジスタ (TIMx_CNT)
- プリスケアラレジスタ (TIMx_PSC)
- 自動再ロードレジスタ (TIMx_ARR)

自動再ロードレジスタはプリロードされます。自動再ロードレジスタの読み書きはプリロードレジスタへのアクセスとなります。プリロードレジスタの内容は、TIMx_CR1 レジスタの自動再ロードプリロードイネーブルビット (ARPE) に応じて、常時または更新イベント UEV ごとに、シャドウレジスタに転送されます。TIMx_CR1 レジスタの UDIS ビットが 0 である場合、カウンタがオーバーフロー値に達すると、更新イベントが送信されます。また、ソフトウェアによって生成することもできます。更新イベントの生成については、各設定の詳細で説明されています。

カウンタのクロックは、TIMx_CR1 レジスタのカウントイネーブルビット (CEN) がセットされているときのみ、プリスケアラ出力 CK_CNT から供給されます。

実際のカウントイネーブル信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケアラの説明

プリスケアラは、カウンタクロック周波数を 1 から 65536 までの間の値で分周することができます。16 ビットレジスタ (TIMx_PSC レジスタ) を使って制御される 16 ビットカウンタをベースとしています。TIMx_PSC 制御レジスタはバッファされているので、動作中に変更できます。新しいプリスケアラ比は、次の更新イベントから有効になります。

[図 181](#) と [図 182](#) に、プリスケアラ比を動作中に変更したときのカウンタの動作の例を示します。

図 181. プリスケーラ分周比が 1 から 2 に変化したときのカウンタのタイミング図

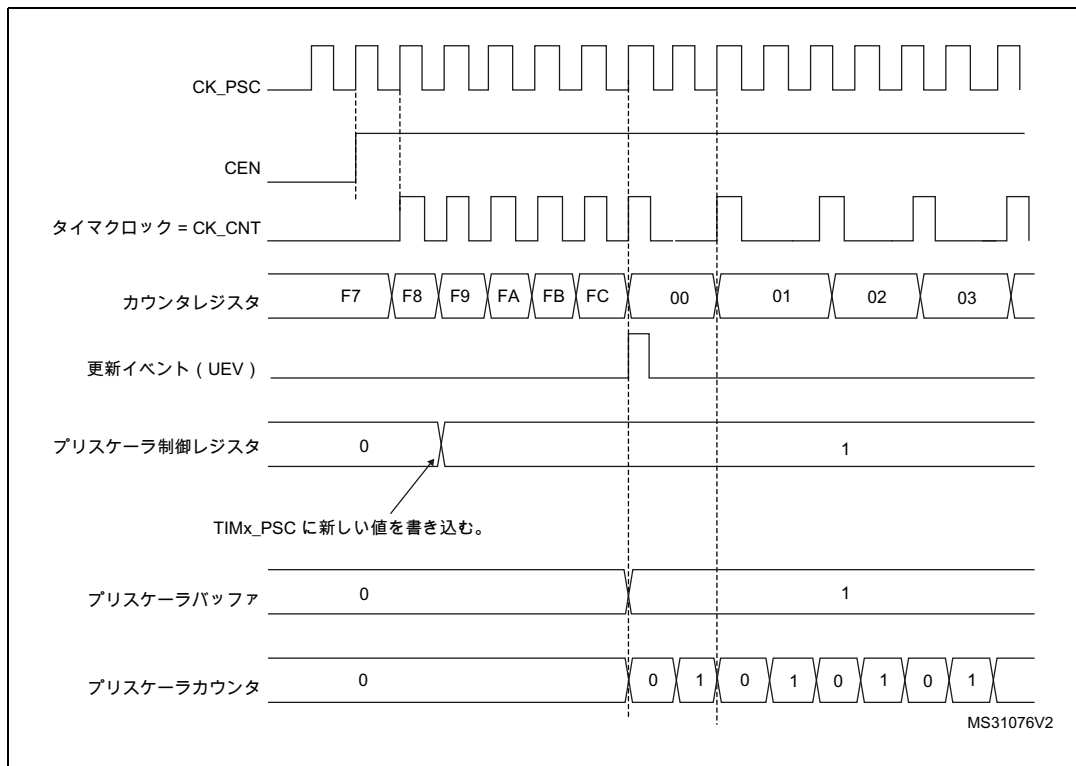
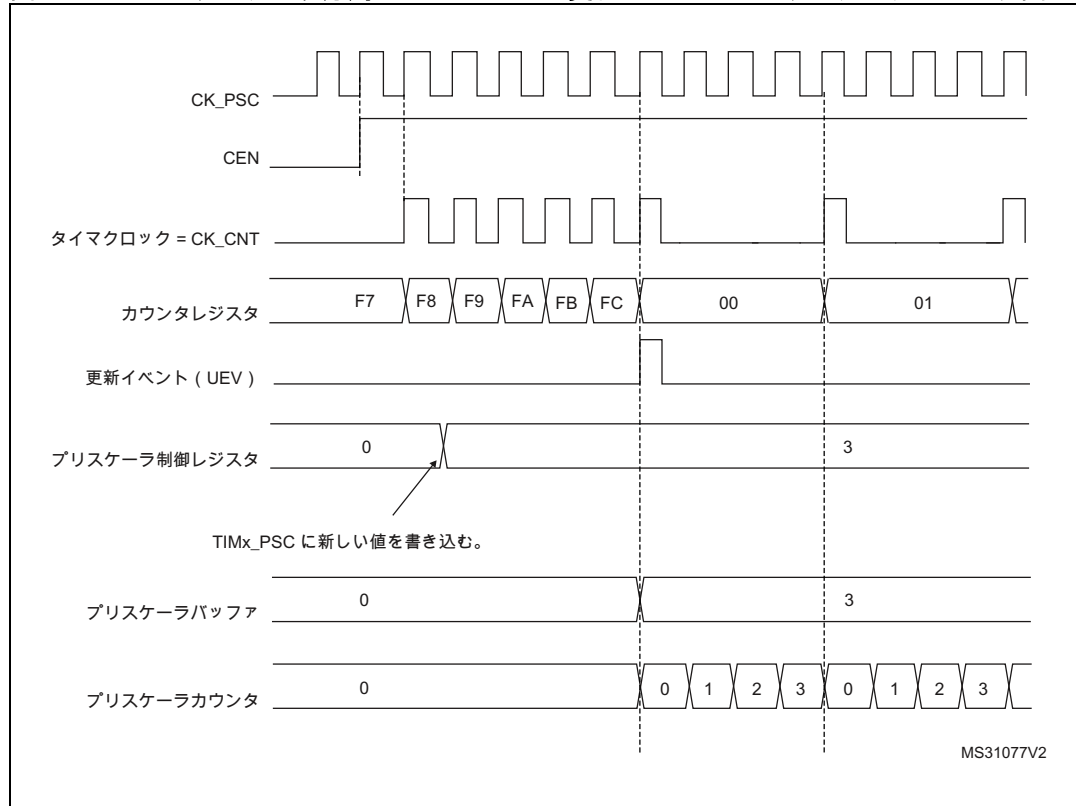


図 182. プリスケーラ分周比が 1 から 4 に変化したときのカウンタのタイミング図



23.3.2 カウントモード

カウンタは、0 から自動再ロード値 (TIMx_ARR レジスタの内容) までカウントした後、0 からカウントをリスタートして、カウンタオーバーフローイベントを生成します。

更新イベントは、カウンタオーバーフローごとに、または、TIMx_EGR レジスタの UG ビットをセットすることによって (ソフトウェアで、または、スレーブモードコントローラを使用して) 生成できます。

UEV イベントは、ソフトウェアで TIMx_CR1 レジスタの UDIS ビットをセットすることによって無効にできます。これは、プリロードレジスタに新しい値を書き込んでいるときにシャドウレジスタが更新されるのを防ぐためです。このようにすると、UDIS ビットに 0 が書き込まれるまで更新イベントは発生しませんが、カウンタとプリスケアラカウンタは両方とも 0 からリスタートします (ただし、プリスケアラ比は変化しません)。さらに、TIMx_CR1 レジスタの URS (更新リクエスト選択) ビットがセットされている場合、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (割り込みや DMA リクエストは送信されません)。

更新イベントが発生すると、すべてのレジスタが更新され、URS ビットの設定に応じて、更新フラグ (TIMx_SR レジスタの UIF ビット) がセットされます。

- プリスケアラのバッファにはプリロード値 (TIMx_PSC レジスタの内容) が再ロードされます。
- 自動再ロードシャドウレジスタは、プリロード値 (TIMx_ARR) で更新されます。

以下の図は、TIMx_ARR = 0x36 のときの、さまざまなクロック周波数におけるカウンタの動作の例を示します。

図 183. 内部クロック分周比が 1 の場合のカウンタのタイミング図

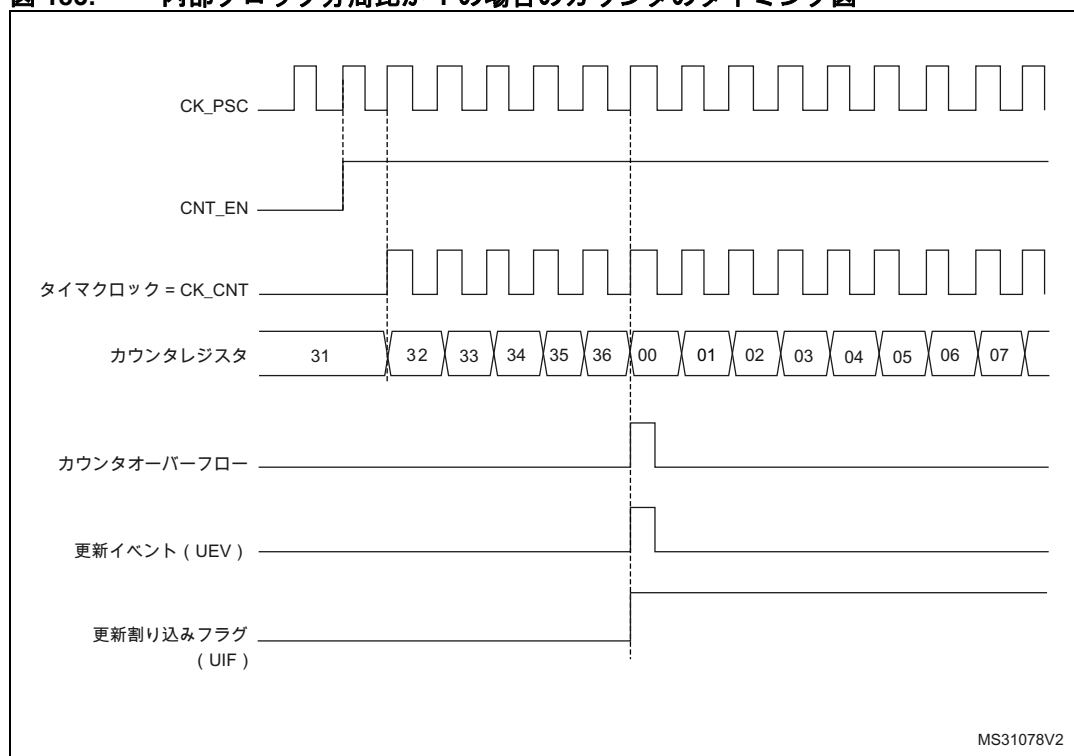


図 184. 内部クロック分周比が 2 の場合のカウンタのタイミング図

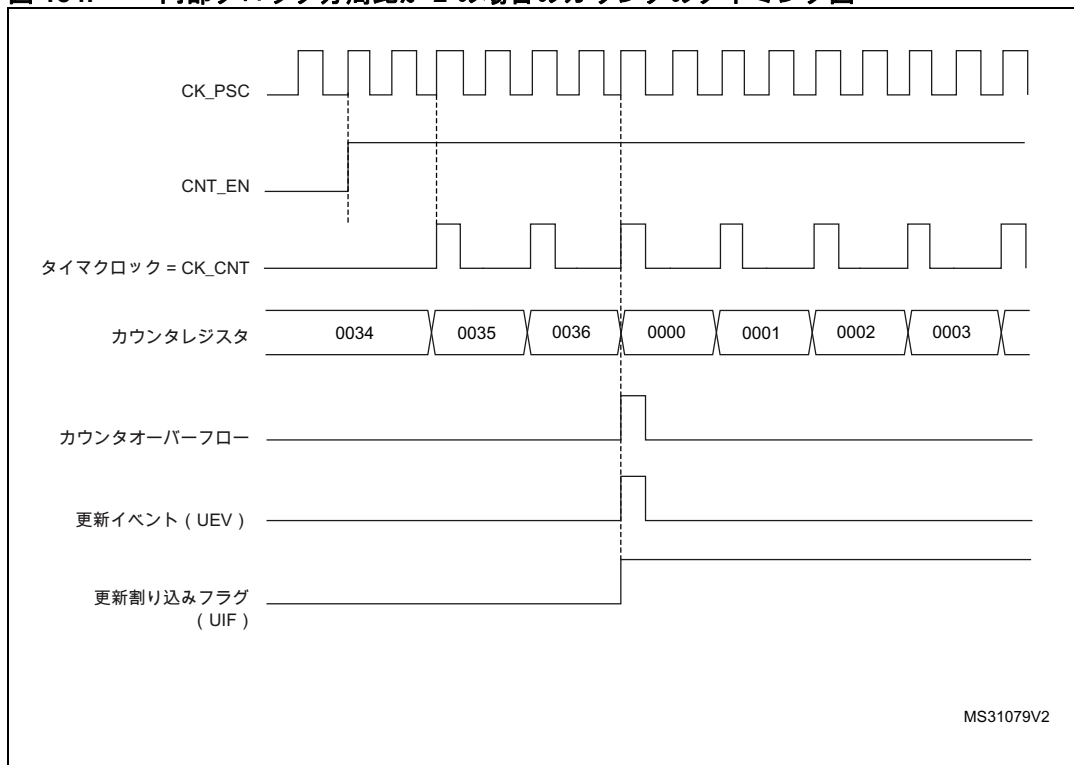


図 185. 内部クロック分周比が 4 の場合のカウンタのタイミング図

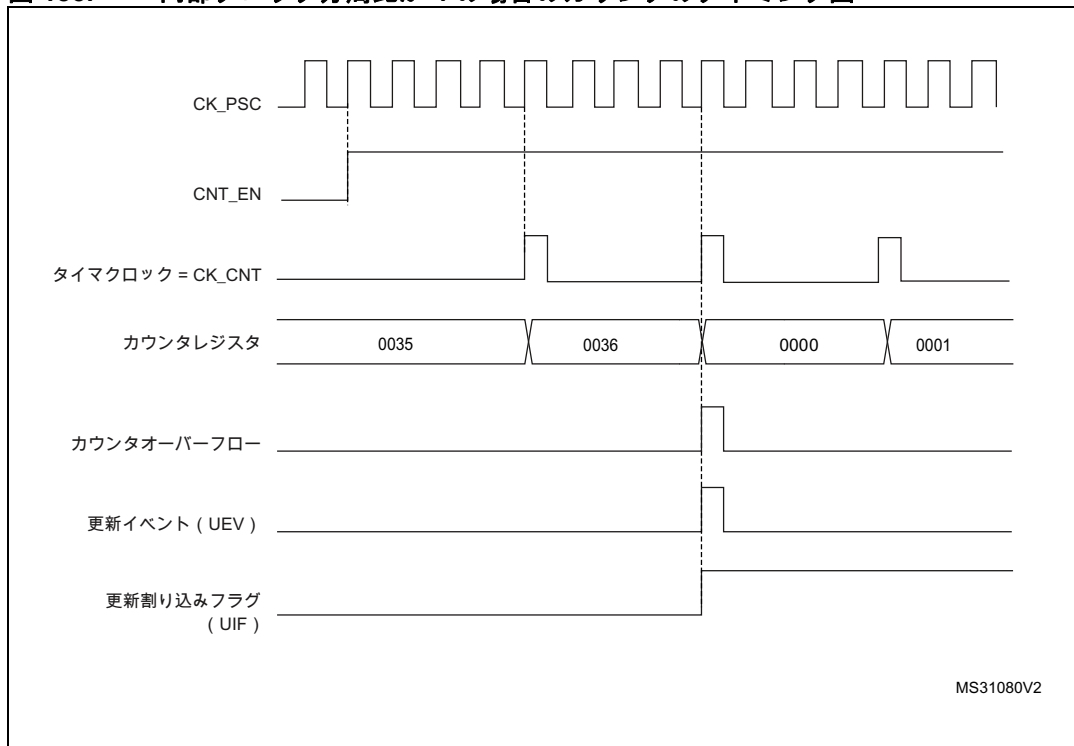


図 186. 内部クロック分周比が N の場合のカウンタのタイミング図

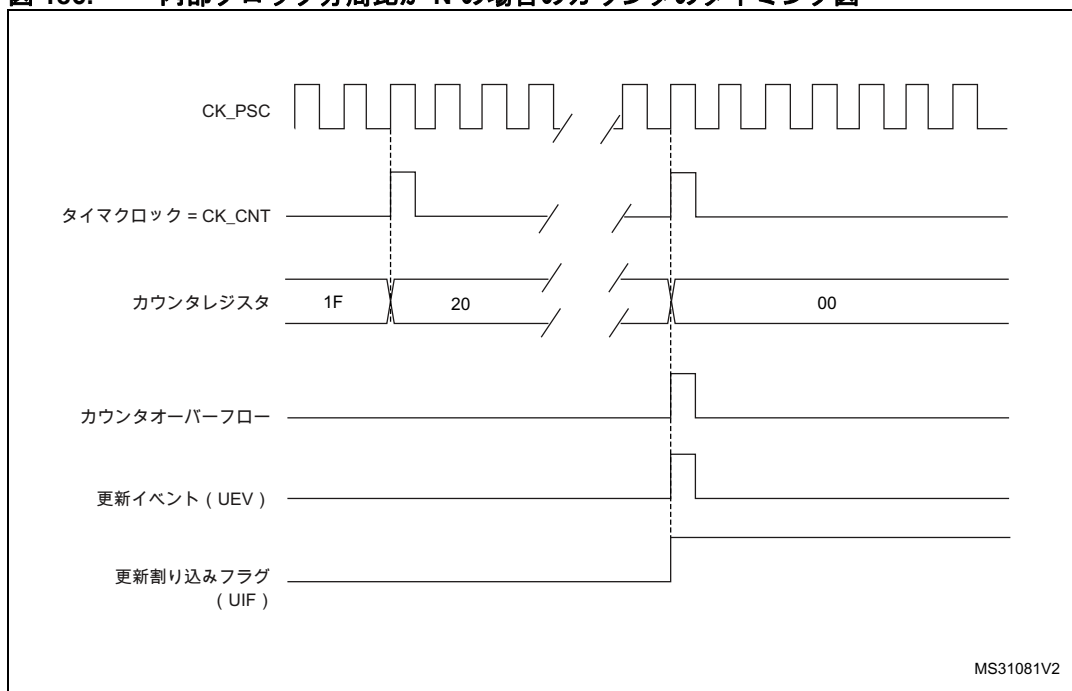
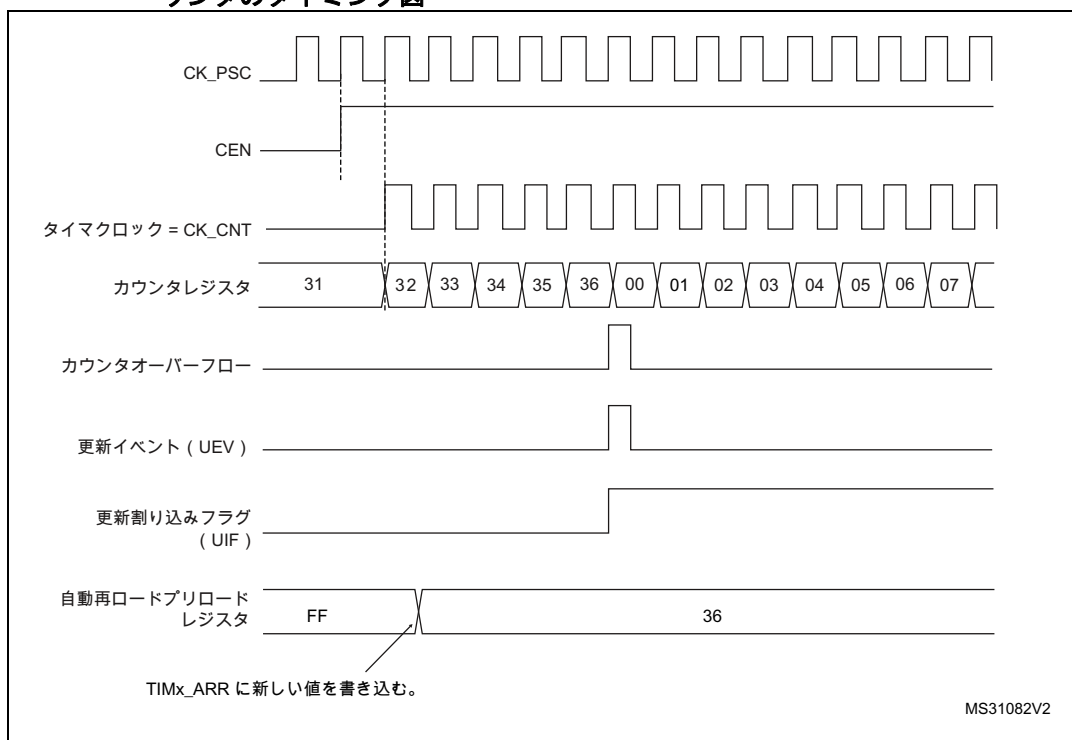


図 187. ARPE = 0 (TIMx_ARR はプリロードされない) の場合の更新イベント時のカウンタのタイミング図



CK_PSC

CEN

タイマクロック = CK_CNT

カウンタレジスタ

F0 F1 F2 F3 F4 F5 00 01 02 03 04 05 06 07

カウンタオーバーフロー

更新イベント (UEV)

更新割り込みフラグ (UIF)

自動再ロードプリロードレジスタ

F5 36

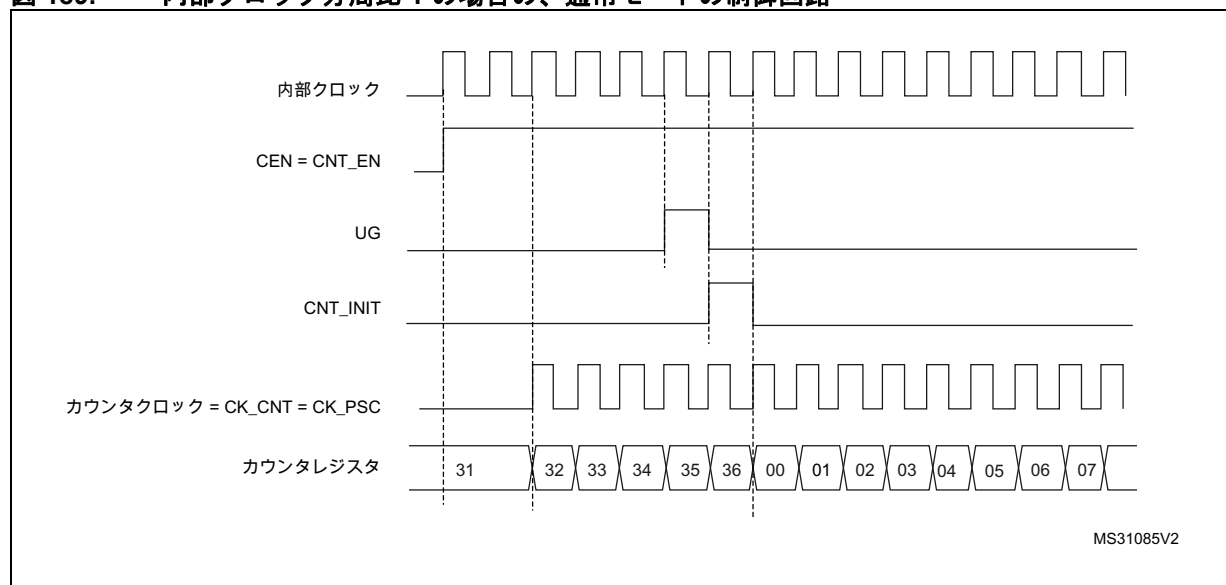
自動再ロードシャドウレジスタ

F5 36

TIMx_ARR に新しい値を書き込む。

MS31083V2

図 189. 内部クロック分周比 1 の場合の、通常モードの制御回路



23.3.4 デバッグモード

マイクロコントローラがデバッグモードになると (Cortex®-M0+ コアは停止状態)、TIMx カウンタは、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、[Section 31.16.2: Debug support for timers, watchdog, bxCAN and I2C](#)を参照してください。

23.4 TIM6 レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 : レジスタに関する略記](#)を参照してください。

ペリフェラルレジスタには、ハーフワード (16 ビット) またはワード (32 ビット) 単位で書き込む必要があります。読み出しアクセスは、バイト (8 ビット)、ハーフワード (16 ビット) またはワード (32 ビット) 単位で行うことができます。

23.4.1 TIM6 制御レジスタ 1 (TIMx_CR1)

アドレスオフセット : 0x00

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|------|------|------|-----|-----|------|-----|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ARPE | Res. | Res. | Res. | OPM | URS | UDIS | CEN |
| | | | | | | | | r/w | | | | r/w | r/w | r/w | r/w |

ビット 15:8 予約済みであり、リセット値に保持する必要があります。

ビット 7 **ARPE** : 自動再ロードプリロードイネーブル

0 : TIMx_ARR レジスタはパツファされません。

1 : TIMx_ARR レジスタはパツファされます。

ビット 6:4 予約済みであり、リセット値に保持する必要があります。

ビット 3 **OPM** : ワンパルスモード

0 : カウンタは更新イベントで停止しません。

1 : カウンタは次の更新イベントでカウントを停止します (CEN ビットをクリア)。

ビット 2 **URS** : 更新リクエストソース

このビットは、UEV イベントソースを選択するために、ソフトウェアでセット/クリアされます。

0 : 次のイベントのいずれかが更新割り込みまたは DMA リクエストを生成します (有効な場合)。該当するイベントは、次のとおりです。

- カウンタオーバーフロー/アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

1 : カウンタオーバーフロー/アンダーフローのみが更新割り込みまたは DMA リクエストを生成します (有効な場合)。

ビット 1 **UDIS** : 更新ディセーブル

このビットは、UEV イベント生成を有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : UEV は有効です。更新イベント (UEV) は、次のいずれかのイベントによって生成されます。

- カウンタオーバーフロー/アンダーフロー
- UG ビットのセット
- スレーブモードコントローラからの更新生成

パツファされたレジスタにはプリロード値がロードされます。

1 : UEV は無効です。更新イベントは生成されず、シャドウレジスタ (ARR、PSC) はそれぞれの値を維持します。ただし、UG ビットがセットされた場合や、スレーブモードコントローラからハードウェアリセットを受信した場合には、カウンタとプリスケータは再初期化されます。

ビット 0 **CEN** : カウンタイネーブル

0 : カウンタは無効です。

1 : カウンタは有効です。

注 : ゲートモードは、CEN ビットが事前にソフトウェアでセットされている場合にのみ動作します。ただし、トリガモードでは、ハードウェアによって自動的に CEN ビットをセットできます。

ワンパルスモードでは、更新イベントが発生すると、CEN が自動的にクリアされます。

23.4.2 TIM6 制御レジスタ 2 (TIMx_CR2)

アドレスオフセット : 0x04

リセット値 : 0x0000

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

ビット 15:7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **MMS** : マスタモード選択

これらのビットは、同期のためにマスタモードでスレーブタイマに送信される情報 (TRGO) を選択するために使用します。組み合わせは、次のとおりです。

000 : **リセット** - TIMx_EGR レジスタの UG ビットがトリガ出力 (TRGO) として使用されます。トリガ入力によってリセットが発生したとき (スレーブモードコントローラがリセットモードに設定されているとき) TRGO 信号は実際のリセットから遅れて発生します。

001 : **イネーブル** - カウンタイネーブル信号 CNT_EN がトリガ出力 (TRGO) として使用されます。これは、いくつかのタイマを同時に開始するときや、スレーブタイマが有効な時間枠を制御するときに役立ちます。カウンタイネーブル信号は、ゲートモードに設定されているとき、CEN 制御ビットとトリガ入力との論理和 (OR) によって生成されます。

カウンタイネーブル信号がトリガ入力によって制御されているとき、マスタ/スレーブモードが選択されている場合を除いて、TRGO に遅延が存在します (TIMx_SMCR レジスタの MSM ビットの説明を参照してください)。

010 : **更新** - 更新イベントがトリガ出力 (TRGO) として選択されます。たとえば、マスタタイマをスレーブタイマのプリスケアラとして使用できます。

ビット 3:0 予約済みであり、リセット値に保持する必要があります。

23.4.3 TIM6 DMA/割り込み有効レジスタ (TIMx_DIER)

アドレスオフセット : 0x0C

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|-----|------|------|------|------|------|------|------|-----|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | UDE | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UIE |
| | | | | | | | rw | | | | | | | | rw |

ビット 15:9 予約済みであり、リセット値に保持する必要があります。

ビット 8 **UDE** : 更新 DMA リクエストイネーブル

0 : 更新 DMA リクエストは無効です。

1 : 更新 DMA リクエストは有効です。

ビット 7:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIE** : 更新割り込みイネーブル

0 : 更新割り込みは無効です。

1 : 更新割り込みは有効です。

23.4.4 TIM6 のステータスレジスタ (TIMx_SR)

アドレスオフセット : 0x10

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UIF |
| | | | | | | | | | | | | | | | rc_w0 |

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UIF** : 更新割り込みフラグ

このビットは、更新イベント時にハードウェアによってセットされます。ソフトウェアによってクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みが保留中です。このビットは、レジスタが以下で更新されたときにハードウェアによってセットされます。

- TIMx_CR1 レジスタの UDIS = 0 であり、繰り返しカウンタ値でオーバーフローまたはアンダーフローが発生したとき。
- TIMx_CR1 レジスタの URS = 0 かつ UDIS = 0 の場合に、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアで CNT が再初期化されたとき。

23.4.5 TIM6 のイベント生成レジスタ (TIMx_EGR)

アドレスオフセット : 0x14

リセット値 : 0x0000

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

ビット 15:1 予約済みであり、リセット値に保持する必要があります。

ビット 0 **UG** : 更新生成

このビットは、ソフトウェアによってセットでき、ハードウェアによって自動的にクリアされます。

0 : 影響はありません。

1 : タイマカウンタを再初期化して、レジスタの更新を生成します。プリスケアラカウンタもクリアされます (プリスケアラ比は変化しません)。

23.4.6 TIM6 のカウンタ (TIMx_CNT)

アドレスオフセット : 0x24

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| CNT[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 **CNT[15:0]** : カウンタ値



23.4.7 TIM6 のプリスケアラ（TIMx_PSC）

アドレスオフセット：0x28

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| PSC[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 **PSC[15:0]**：プリスケアラ値
カウンタクロック周波数 CK_CNT は、 $f_{CK_PSC} / (PSC[15:0] + 1)$ に等しいです。
PSC は、更新イベントごとにアクティブなプリスケアラレジスタにロードされる値を含みます。

23.4.8 TIM6 の自動再ロードレジスタ（TIMx_ARR）

アドレスオフセット：0x2C

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|-----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ARR[15:0] | | | | | | | | | | | | | | | |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 15:0 **ARR[15:0]**：自動再ロード値
ARR は、実際の自動再ロードレジスタにロードされる値です。
APR の更新と動作の詳細については、[セクション 23.3.1：タイムベースユニット \(528 ページ\)](#) を参照してください。
自動再ロード値が null のときには、カウンタはブロックされます。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

23.4.9 TIM6 レジスタマップ

TIMx レジスタは、次の表のように、16 ビットアドレス可能レジスタとして配置されます。

表 87. TIM6 レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----------|-----------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|-----------|------|------|------|------|------|------|------|------|----------|------|------|------|------|------|------|------|-----|
| 0x00 | TIMx_CR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ARPE | Res. | Res. | Res. | Res. | OPM | URS | UDIS | CEN |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | 0 | 0 | 0 | |
| 0x04 | TIMx_CR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MMS[2:0] | | | Res. | Res. | Res. | Res. | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | | | | |
| 0x08 | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x0C | TIMx_DIER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UDE | Res. | Res. | Res. | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | | | 0 | UIE | |
| 0x10 | TIMx_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | UIF | |
| 0x14 | TIMx_EGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UG | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | |
| 0x18 | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x1C | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x20 | Res. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x24 | TIMx_CNT | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[15:0] | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x28 | TIMx_PSC | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PSC[15:0] | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x2C | TIMx_ARR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ARR[15:0] | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

24 低電力タイマ（LPTIM）

24.1 概要

LPTIM は、消費電力削減の究極的な進展を利用した 16 ビットタイマです。クロックソースの多様性により、LPTIM は選択された電源モードにかかわらず実行し続けることができます。内部クロックソースがなくても実行できるため、LPTIM は、一部のアプリケーションで役立つ「パルスカウンタ」として使用することができます。また、LPTIM はシステムを低電力モードからウェイクアップできるため、消費電力が極端に低い「タイムアウト機能」の実現に適しています。

LPTIM の柔軟性の高いクロック方式は、必要な機能性とパフォーマンスを提供しながら、消費電力を最小化します。

24.2 LPTIM の主な機能

- 16 ビットアップカウンタ
- 8 つの分周比（1、2、4、8、16、32、64、128）を持つ3 ビットプリスケアラ
- 選択可能なクロック
 - － 内部クロックソース：LSE、LSI、HSI16、または APB クロック
 - － ULPTIM 入力経由の外部クロック（LP オシレータが実行していないときに動作、パルスカウンタアプリケーションによって使用）
- 16 ビット ARR 自動再ロードレジスタ
- 16 ビット比較レジスタ
- 連続／ワンショットモード
- 選択可能なソフトウェア／ハードウェア入力トリガ
- プログラム可能なデジタルグリッチフィルタ
- 設定可能な出力：パルス、PWM
- 設定可能な I/O 極性
- エンコーダモード

24.3 LPTIM の実装

表 88 に、STM32L0x3 デバイスでの LPTIM の実装を示します。

表 88. STM32L0x3 LPTIM 機能

| LPTIM モード／機能 ⁽¹⁾ | LPTIM1 |
|-----------------------------|--------|
| エンコーダモード | X |

1. X = サポート。

CKSEL ビットに書き込むことによって、LPTIM が外部クロックソースと内部クロックソースのいずれを使用するかを決めることができます。

外部クロックソースを使用するように設定されたときには、CKPOL ビットを使用して外部クロック信号のアクティブエッジを選択します。両方のエッジがアクティブとして設定された場合は、内部クロック信号も供給されます（最初の構成）。この場合、内部クロック信号の周波数は、外部クロック信号の周波数の 4 倍以上である必要があります。

24.4.3 グリッチフィルタ

LPTIM 入力（外部または内部）はデジタルフィルタによって保護され、グリッチとノイズの心配が LPTIM 内部に伝播されるのを防ぎます。これは、誤ったカウントまたはトリガを避けるためです。

デジタルフィルタをアクティブにする前に、内部クロックソースを LPTIM に供給する必要があります。これは、フィルタの正しい動作を保証するために必要です。

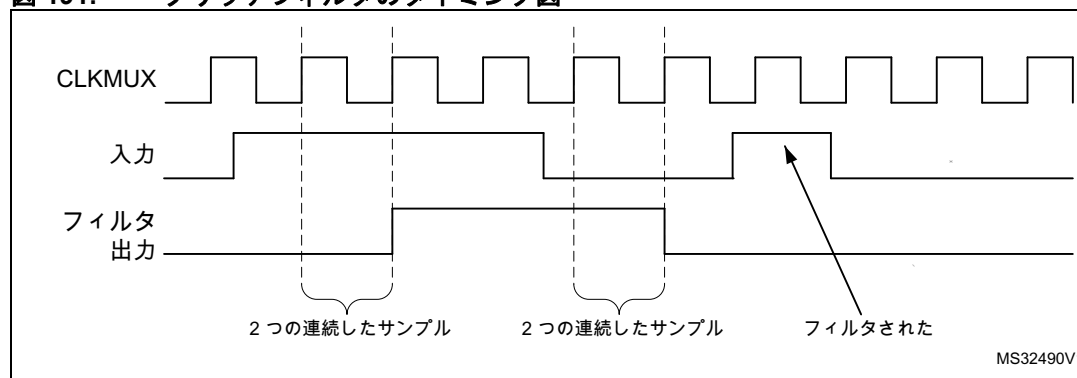
デジタルフィルタは、2 つのグループに分けられます。

- 最初のグループのデジタルフィルタは、LPTIM の外部入力を保護します。デジタルフィルタの感度は CKFLT ビットによって制御されます。
- 2 番目のグループのデジタルフィルタは、LPTIM の内部トリガ入力を保護します。デジタルフィルタの感度は TRGFLT ビットによって制御されます。

注： デジタルフィルタの感度は、グループごとに制御されます。同じグループ内の各デジタルフィルタの感度を個別に設定することはできません。

フィルタの感度は、信号のレベル変化を有効な遷移とみなすために、LPTIM 入力の 1 つで検出される連続した等しいサンプルの数に基づいて作用します。図 191 に、2 つの連続サンプルがプログラムされた場合のグリッチフィルタの動作例を示します。

図 191. グリッチフィルタのタイミング図



注： 内部クロック信号が供給されない場合は、CKFLT および TRGFLT ビットを 0 にセットすることによって、デジタルフィルタを無効にする必要があります。その場合、外部アナログフィルタを使用して、LPTIM の外部入力をグリッチから保護できます。

24.4.4 プリスケアラ

LPTIM 16 ビットカウンタの前には、設定可能な 2 のべき乗プリスケアラがあります。プリスケアラの分周比は PRESC[2:0] 3 ビットフィールドによって制御されます。下の表に、可能な分周比を示します。

表 89. プリスケアラの分周比

| プログラミング | 分周比 |
|---------|------|
| 000 | /1 |
| 001 | /2 |
| 010 | /4 |
| 011 | /8 |
| 100 | /16 |
| 101 | /32 |
| 110 | /64 |
| 111 | /128 |

24.4.5 トリガマルチプレクサ

LPTIM カウンタは、ソフトウェアによって、または 8 つのトリガ入力の 1 つのアクティブエッジの検出後に開始できます。

LPTIM のトリガソースを決めるには、TRIGEN[1:0] が使用されます。

- TRIGEN[1:0] が 00 の場合、LPTIM カウンタは、CNTSTRT または SNGSTRT ビットがソフトウェアによってセットされるとすぐに開始します。
- TRIGEN[1:0] の残りの 3 つの可能な値は、トリガ入力によって使用されるアクティブエッジを設定するために使用されます。LPTIM カウンタは、アクティブエッジが検出されるとすぐに開始します。

TRIGEN[1:0] が 00 以外のときには、カウンタの開始に使用される 8 つのトリガ入力の 1 つを選択するために、TRIGSEL[2:0] が使用されます。

外部トリガは、LPTIM の場合、非同期信号とみなされます。したがって、同期のために、トリガ検出後、タイマが実行を開始するまでに 2 カウンタクロック周期の遅延が必要です。

タイマがすでに開始しているときに新しいトリガイイベントが発生した場合、無視されます（タイムアウト機能が有効な場合を除きます）。

注： SNGSTRT/CNTSTRT ビットをセットする前に、タイマが有効になっている必要があります。タイマが無効なときにこれらのビットに書き込むと、ハードウェアによって破棄されます。

24.4.6 動作モード

LPTIM には 2 つの動作モードがあります。

- 連続モード：タイマはフリーランニングし、トリガイイベントによって開始し、タイマが無効にされるまで停止しません。
- ワンショットモード：タイマはトリガイイベントによって開始し、ARR 値に達すると停止します。

新しいトリガイイベントが発生すると、タイマは再開始します。カウンタの開始後、カウンタが ARR に達する前に発生したトリガイイベントは破棄されます。



ワンショットカウントを有効にするには、SNGSTRT ビットをセットする必要があります。

外部トリガが選択された場合、SNGSTRT がセットされた後に外部トリガイイベントが着信すると、カウンタはワンショットカウントを開始します。

ソフトウェアによる開始の場合 (TRIGEN[1:0] = 00)、SNGSTRT をセットすると、カウンタはワンショットカウントを開始します。

連続カウントを有効にするには、CNTSTRT ビットをセットする必要があります。

外部トリガが選択された場合、CNTSTRT がセットされた後に外部トリガイイベントが着信すると、カウンタは連続カウントを開始します。

ソフトウェアによる開始の場合 (TRIGEN[1:0] = 00)、CNTSTRT をセットすると、カウンタは連続カウントを開始します。

SNGSTRT および CNTSTRT ビットは、タイマが有効なときのみ (ENABLE ビットが 1 にセットされている)、セットできます。動作中にワンショットモードから連続モードに変更することが可能です。

以前に連続モードが選択されていた場合、SNGSTRT をセットすると、LPTIM はワンショットモードに切り替わります。カウンタ (アクティブな場合) は、ARR に達するとすぐに停止します。

以前にワンショットモードが選択されていた場合、CNTSTRT をセットすると、LPTIM は連続モードに切り替わります。カウンタ (アクティブな場合) は、ARR に達するとすぐに再開します。

24.4.7 タイムアウト機能

選択されたトリガ入力のアクティブエッジの検出を使用して、LPTIM カウンタをリセットできます。この機能は、TIMOUT ビットで制御されます。

最初のトリガイイベントでタイマが開始し、その後のトリガイイベントでカウンタがリセットされ、タイマが再開します。

低電力タイムアウト機能を実現できます。タイムアウト値は、比較値に対応します。予期された時間内にトリガが発生しなかった場合、比較一致イベントによって MCU がウェイクアップします。

24.4.8 波形生成

2 つの 16 ビットレジスタ LPTIMx_ARR (自動再ロードレジスタ) と LPTIMx_CMP (比較レジスタ) は、LPTIM 出力のいくつかの異なる波形を生成するために使用されます。

タイマは次の波形を生成できます。

- PWM モード: LPTIM 出力は、LPTIMx_CMP レジスタと LPTIMx_CNT レジスタの間で一致が発生するとすぐにセットされます。LPTIM 出力は、LPTIMx_ARR レジスタと LPTIMx_CNT レジスタの間で一致が発生するとすぐにリセットされます。
- ワンパルスモード: 出力波形は、最初のパルスについては PWM モードの波形と同様であり、その後、出力は永続的にリセットされます。
- セットワンスモード: 出力波形はワンパルスモードと同様ですが、出力は最後の信号レベルに保たれます (設定された出力極性に応じて)。

上記のモードでは、LPTIMx_ARR レジスタの値は LPTIMx_CMP レジスタの値より大きい必要があります。

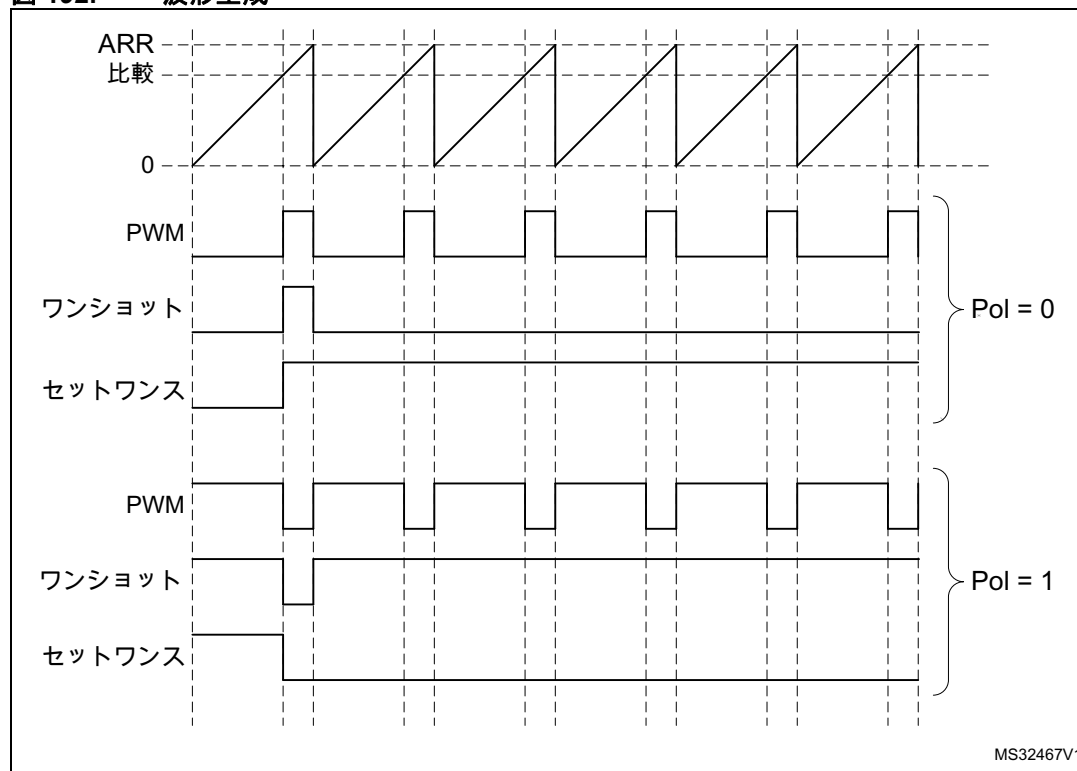
LPTIM 出力波形は、WAVE ビットによって次のように設定できます。

- WAVE ビットを 0 にリセットすると、LPTIM は、CNTSTRT または SNGSTRT のいずれのビットがセットされているかに応じて、PWM 波形またはワンパルス波形のいずれかを生成します。
- WAVE ビットを 1 にセットすると、LPTIM はセットワンス波形を生成します。

WAVPOL ビットは、LPTIM 出力の極性を制御します。変更はただちに有効になるので、極性が再設定されると、タイマが有効になる前でも、出力はただちに変更されます。

LPTIM クロック周波数を 2 分周した周波数までの信号を生成できます。図 192 に、LPTIM 出力で生成できる 3 つの波形を示します。また、WAVPOL ビットを使用して極性を変更したときの効果も示します。

図 192. 波形生成



24.4.9 レジスタの更新

LPTIMx_ARR レジスタと LPTIMx_CMP レジスタは、APB バス書き込み操作の直後に、またはタイマがすでに開始していた場合は現在の周期の終了時に更新されます。

PRELOAD ビットは、LPTIMx_ARR および LPTIMx_CMP レジスタの更新方法を制御します。

- PRELOAD ビットが 0 にリセットされたときには、LPTIMx_ARR および LPTIMx_CMP レジスタは書き込みアクセスの直後に更新されます。
- PRELOAD ビットが 1 にセットされたときには、LPTIMx_ARR および LPTIMx_CMP レジスタは、タイマがすでに開始していた場合、現在の周期の終了時に更新されます。

APB バスと LPTIM は異なるクロックを使用するので、APB 書き込みから、これらの値がカウンタコンパレータで使用可能になるまで、遅延があります。この遅延中は、これらのレジスタへの追加の書き込みを避ける必要があります。

LPTIMx_ISR レジスタの ARROK フラグと CMPOK フラグは、それぞれ、LPTIMx_ARR レジスタと LPTIMx_CMP レジスタへの書き込み操作が完了したことを示します。

LPTIMx_ARR レジスタまたは LPTIMx_CMP レジスタへの書き込みの後、同じレジスタへの新しい書き込み操作は、前の書き込み操作が完了してからでなければ実行できません。ARROK フラグまたは CMPOK フラグがセットされる前に連続した書き込みが行われると、予測不能な結果になります。

24.4.10 カウンタモード

LPTIM カウンタを使用して、LPTIM Input1 の外部イベントをカウントするか、内部クロックサイクルをカウントすることができます。COUNTMODE ビットは、カウンタの更新に使用するソースを制御します。

LPTIM が Input1 の外部イベントをカウントするように設定された場合、カウンタは、CKPOL[1:0] ビットに書き込まれた値に応じて、立ち上がりエッジ、立ち下がりエッジ、または両方のエッジで更新できます。

CKSEL ビットは、LPTIM のクロック動作方法と、ソース信号のエッジをカウンタの更新に使用できる条件を決めます。

- CKSEL = 0 : LPTIM のクロックは、内部クロックソースによって供給されます。
LPTIM が内部クロックソースによってクロック供給されるように設定され、LPTIM カウンタが LPTIM 外部 Input1 で検出されたアクティブエッジによって更新されるように設定されたときには、LPTIM に供給される内部クロックをプリスケールしてはなりません (PRESCL[2:0] = 000)。
LPTIM 外部 Input1 は、LPTIM に供給される内部クロックでサンプリングされます。結果として、イベントをミスしないためには、外部 Input1 信号の変化の周波数が、LPTIM に供給される内部クロックの周波数を超えない必要があります。
- CKSEL = 1 : LPTIM のクロックは、外部クロックソースによって供給されます。
この構成では、LPTIM は内部クロックソースを必要としません。LPTIM 外部 Input1 に入力された信号が LPTIM のシステムクロックとして使用されます。この構成は、埋め込みオシレータを有効にしない動作モードに適しています。
この構成の場合、LPTIM カウンタは、Input1 クロック信号の立ち上がりエッジまたは立ち下がりエッジで更新できますが、立ち上がりと立ち下がりの両方のエッジで更新することはできません。
LPTIM 外部 Input1 に入力された信号は LPTIM のクロック動作にも使用されるので、カウンタがインクリメントされる前 (LPTIM が有効にされた後)、初期遅延があります。より正確には、LPTIM 外部 Input1 の (LPTIM が有効になった後の) 最初の 5 つのアクティブエッジは失われます。

24.4.11 タイマ有効

LPTIMx_CR レジスタの ENABLE ビットは、LPTIM を有効化／無効化するために使用されます。ENABLE ビットをセットした後、LPTIM が実際に有効になるまで、2 カウンタクロックの遅延が必要です。

LPTIMx_CFGR および LPTIMx_IER レジスタの変更は、LPTIM が無効なときにのみ行う必要があります。

24.4.12 エンコーダモード

このモードでは、ロータリー素子の角度位置の検出に使用される直交エンコーダからの信号を処理できます。エンコーダインタフェースモードは、方向選択を含む外部クロックとして動作します。これは、カウンタが 0 と LPTIMx_ARR レジスタでプログラムされた自動再ロード値の間で (方向に応じて、0 から ARR まで、または ARR から 0 まで) 連続的にカウントすることを意味します。したがって、開始前に LPTIMx_ARR を設定する必要があります。Input1 と Input2 の 2 つの外部入力信号から、LPTIM カウンタのクロックのためのクロック信号が生成されます。この 2 つの信号の間の位相によって、カウント方向が決まります。

エンコーダモードは、LPTIM が内部クロックソースからクロック供給されるときにのみ使用できます。Input1 と Input2 の両方の信号周波数は、LPTIM 内部クロック周波数を 4 分周したものを超えてはなりません。これは、LPTIM の正しい動作を保証するために必要です。

方向の変更は、LPTIMx_ISR レジスタの Down と Up の 2 つのフラグによって通知されます。また、LPTIMx_IER レジスタを通じて有効化された場合、両方の方向変更イベントで割り込みを生成できます。

エンコーダモードを有効にするには、ENC ビットを 1 にセットする必要があります。LPTIM を、まず、連続モードに設定する必要があります。

エンコーダモードがアクティブなとき、LPTIM カウンタはインクリメンタルエンコーダの速度と方向に従って自動的に変更されます。したがって、その内容は常にエンコーダの位置を表します。カウンタ方向は、Up および Down フラグによって通知され、接続されているセンサの回転方向に対応します。

CKPOL[1:0] ビットを使用して設定されたエッジ感度に応じて、さまざまなカウントシナリオが可能です。次の表に、可能な組み合わせを示します（Input1 と Input2 は同時に切り替わらないと想定しています）。

表 90. エンコーダのカウントシナリオ

| アクティブエッジ | 他方の信号のレベル (Input2 に対する Input1、Input1 に対する Input2) | Input1 信号 | | Input2 信号 | |
|----------|---|-----------|--------|-----------|--------|
| | | 立ち上がり | 立ち下がり | 立ち上がり | 立ち下がり |
| 立ち上がりエッジ | ハイ | ダウン | カウントなし | アップ | カウントなし |
| | ロー | アップ | カウントなし | ダウン | カウントなし |
| 立ち下がりエッジ | ハイ | カウントなし | アップ | カウントなし | ダウン |
| | ロー | カウントなし | ダウン | カウントなし | アップ |
| 両エッジ | ハイ | ダウン | アップ | アップ | ダウン |
| | ロー | アップ | ダウン | ダウン | アップ |

次の図に、両方のエッジ感度が設定された場合のエンコーダモードのカウントシーケンスを示します。

注意： このモードでは、LPTIM のクロックは内部クロックソースによって供給される必要があるため、CKSEL ビットをリセット値（0）に維持する必要があります。また、プリスケアラの分周比はリセット値である 1 に等しくなければなりません（PRESC[2:0] ビットが 000 である必要があります）。

The diagram shows a clock signal at the top, followed by two timing signals, T1 and T2, and a counter signal. The counter signal is a staircase waveform that increases during the 'アップ' (Up) phase and decreases during the 'ダウン' (Down) phase. The 'アップ' phase is marked by a bracket and the label 'アップ' at the bottom. The 'ダウン' phase is marked by a bracket and the label 'ダウン' at the bottom. The counter signal is labeled 'カウンタ' on the left. The T1 and T2 signals are square waves that are high during the 'アップ' phase and low during the 'ダウン' phase. The T1 signal is labeled 'T1' and the T2 signal is labeled 'T2' on the left.

注: LPTIMx_ISR レジスタ (ステータスレジスタ) の対応するフラグがセットされた後で LPTIMx_IER レジスタ (割り込み有効レジスタ) のビットがセットされた場合、割り込みはアサートされません。

24.6 LPTIM レジスタ

24.6.1 LPTIM 割り込みおよびステータスレジスタ（LPTIMx_ISR）

アドレスオフセット：0x00

リセット値：0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------|------|------|------|------|------|------|------|------|------|------|-------|-------|---------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWN | UP | ARROK | CMPOK | EXTTRIG | ARRM | CMPM |
| | | | | | | | | | r | r | r | r | r | r | r |

ビット 31:7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **DOWN** : カウンタの方向をアップからダウンへ変更

エンコーダモードでは、DOWN ビットは、カウンタの方向がアップからダウンに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 5 **UP** : カウンタの方向をダウンからアップへ変更

エンコーダモードでは、UP ビットは、カウンタの方向がダウンからアップに変更されたことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 4 **ARROK** : 自動再ロードレジスタ更新 OK

ARROK は、LPTIMx_ARR レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 3 **CMPOK** : 比較レジスタ更新 OK

CMPOK は、LPTIMx_CMP レジスタへの APB バス書き込み操作が正常に完了したことをアプリケーションに知らせるために、ハードウェアによってセットされます。その場合、新しい書き込みを開始できます。

ビット 2 **EXTTRIG** : 外部トリガエッジイベント

EXTTRIG は、選択された外部トリガ入力で有効なエッジが発生したことをアプリケーションに知らせるために、ハードウェアによってセットされます。タイマがすでに開始していたためにトリガが無視された場合、このフラグはセットされません。

ビット 1 **ARRM** : 自動再ロード一致

ARRM は、LPTIMx_CNT レジスタの値が LPTIMx_ARR レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

ビット 0 **CMPM** : 比較一致

CMPM は、LPTIMx_CNT レジスタの値が LPTIMx_CMP レジスタの値に達したことをアプリケーションに知らせるために、ハードウェアによってセットされます。

24.6.2 LPTIM 割り込みクリアレジスタ（LPTIMx_ICR）

アドレスオフセット：0x04

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------------|------|-------------|-------------|---------------|------------|------------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWNC F | UPCF | ARROK CF | CMPOK CF | EXTTRI GCF | ARRMC F | CMPMC F |
| | | | | | | | | | w | w | w | w | w | w | w |

ビット 31:7 予約済みであり、リセット値のままにしておかなければなりません。

- ビット 6 **DOWNCF**：ダウンへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの DOWN フラグがクリアされます。
- ビット 5 **UPCF**：アップへの方向変更フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの UP フラグがクリアされます。
- ビット 4 **ARROKCF**：自動再ロードレジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARROK フラグがクリアされます。
- ビット 3 **CMPOKCF**：比較レジスタ更新 OK フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMPOK フラグがクリアされます。
- ビット 2 **EXTTRIGCF**：外部トリガ有効エッジフラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの EXTTRIG フラグがクリアされます。
- ビット 1 **ARRMCF**：自動再ロード一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの ARRM フラグがクリアされます。
- ビット 0 **CMPMCF**：比較一致フラグクリア
このビットに 1 を書き込むと LPT_ISR レジスタの CMP フラグがクリアされます。

24.6.3 LPTIM 割り込み有効レジスタ（LPTIMx_IER）

アドレスオフセット：0x08

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|--------|------|---------|---------|-----------|--------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWNIE | UPIE | ARROKIE | CMPOKIE | EXTTRIGIE | ARRMIE | CMPMIE |
| | | | | | | | | | rw | rw | rw | rw | rw | rw | rw |

ビット 31:7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **DOWNIE** : ダウンへの方向変更割り込み有効

- 0 : DOWN 割り込み無効
- 1 : DOWN 割り込み有効

ビット 5 **UPIE** : アップへの方向変更割り込み有効

- 0 : UP 割り込み無効
- 1 : UP 割り込み有効

ビット 4 **ARROKIE** : 自動再ロードレジスタ更新 OK 割り込み有効

- 0 : ARROK 割り込み無効
- 1 : ARROK 割り込み有効

ビット 3 **CMPOKIE** : 比較レジスタ更新 OK 割り込み有効

- 0 : CMPOK 割り込み無効
- 1 : CMPOK 割り込み有効

ビット 2 **EXTTRIGIE** : 外部トリガ有効エッジ割り込み有効

- 0 : EXTTRIG 割り込み無効
- 1 : EXTTRIG 割り込み有効

ビット 1 **ARRMIE** : 自動再ロード一致割り込み有効

- 0 : ARRM 割り込み無効
- 1 : ARRM 割り込み有効

ビット 0 **CMPMIE** : 比較一致割り込み有効

- 0 : CMPM 割り込み無効
- 1 : CMPM 割り込み有効

注意 : **LPTIMx_IER レジスタの変更は、LPTIM が無効（ENABLE ビットが 0 にリセットされている）のときにのみ行う必要があります。**

24.6.4 LPTIM 構成レジスタ（LPTIMx_CFGR）

アドレスオフセット：0x0C

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|---------|------|------|------|-------|------|------|------|---------------|---------|--------|-------|--------|--------|----|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | ENC | COUNT MODE | PRELOAD | WAVPOL | WAVE | TIMOUT | TRIGEN | | Res. |
| | | | | | | | rw | rw | rw | rw | rw | rw | rw | rw | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TRIGSEL | | | Res. | PRESC | | | Res. | TRGFLT | | Res. | CKFLT | | CKPOL | | CKSEL |
| rw | rw | rw | | rw | rw | rw | | rw | rw | | rw | rw | rw | rw | rw |

ビット 31:25 予約済みであり、リセット値のままにしておかなければなりません。

ビット 24 **ENC**：エンコーダモード有効

ENC ビットは、エンコーダモードを制御します。

- 0：エンコーダモード無効
- 1：エンコーダモード有効

ビット 23 **COUNTMODE**：カウンタモード有効

COUNTMODE ビットは、LPTIM がカウンタのクロックに使用するクロックソースを選択します。

- 0：カウンタは各内部クロックのパルスに従ってインクリメントされます。
- 1：カウンタは LPTIM 外部 Input1 の各有効なクロックパルスに従ってインクリメントされます。

ビット 22 **PRELOAD**：レジスタ更新モード

PRELOAD ビットは、LPTIMx_ARR および LPTIMx_CMP レジスタの更新方法を制御します。

- 0：レジスタは、各 APB バス書き込みアクセス後に更新されます。
- 1：レジスタは、現在の LPTIM 周期の終了時に更新されます。

ビット 21 **WAVPOL**：波形極性

WAVPOL ビットは、出力の極性を制御します。

- 0：LPTIM 出力は LPTIMx_ARR レジスタと LPTIMx_CMP レジスタの比較結果を反映します。
- 1：LPTIM 出力は LPTIMx_ARR レジスタと LPTIMx_CMP レジスタの比較結果の逆を反映します。

ビット 20 **WAVE**：波形

WAVE ビットは、出力波形を制御します。

- 0：PWM／ワンパルス波形（OPMODE ビットに応じて）
- 1：セットワンス波形

ビット 19 **TIMOUT**：タイムアウト有効

TIMOUT ビットは、タイムアウト機能を制御します。

- 0：タイマがすでに開始しているときに着信したトリガイベントは無視されます。
- 1：タイマがすでに開始しているときにトリガイベントが着信すると、カウンタがリセットされ、再開始します。

ビット 18:17 **TRIGEN**：トリガ有効および極性

TRIGEN ビットは、LPTIM カウンタが外部トリガによって開始されるかどうかを制御します。外部トリガオプションが選択された場合、トリガのアクティブエッジについて 3 つの構成が可能です。

- 00：ソフトウェアトリガ（カウンタの開始はソフトウェアによって行われます）。
- 01：立ち上がりエッジがアクティブエッジです。
- 10：立ち下がりエッジがアクティブエッジです。
- 11：両方のエッジがアクティブエッジです。

ビット 16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:13 TRIGSEL : トリガセクタ

TRIGSEL ビットは、次の 8 つのソースの中の LPTIM のトリガイベントの役目を果たすトリガソースを選択します。

000 : ext_trig0
001 : ext_trig1
010 : ext_trig2
011 : ext_trig3
100 : ext_trig4
101 : 予約済み
110 : ext_trig6
111 : ext_trig7

各タイマでの ITRx の意味については、[表 91 : LPTIM 外部トリガ接続 \(554 ページ\)](#) を参照してください。

ビット 12 予約済みであり、リセット値のままにしておかなければなりません。

ビット 11:9 PRESC : クロックプリスケアラ

PRESC ビットは、プリスケアラ分周比を設定します。次の分周比から選択できます。

000 : /1
001 : /2
010 : /4
011 : /8
100 : /16
101 : /32
110 : /64
111 : /128

ビット 8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7:6 TRGFLT : 設定可能なトリガ用デジタルフィルタ

TRGFLT 値は、有効なレベル遷移とみなされる前に、内部トリガでレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。

- 00 : トリガのアクティブレベル変更は、有効なトリガとみなされます。
- 01 : トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。
- 10 : トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。
- 11 : トリガのアクティブレベル変更が有効なトリガとみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4:3 **CKFLT** : 設定可能な外部クロック用デジタルフィルタ

- CKFLT 値は、有効なレベル遷移とみなされる前に、外部クロック信号でレベル変更が発生したときに検出されなければならない連続した等しいサンプルの数を設定します。この機能を使用するには、内部クロックソースが必要です。
- 00 : 外部クロック信号のレベル変更は、有効な遷移とみなされます。
 - 01 : 外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 2 クロック周期にわたって安定している必要があります。
 - 10 : 外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 4 クロック周期にわたって安定している必要があります。
 - 11 : 外部クロック信号のレベル変更が有効な遷移とみなされるためには、少なくとも 8 クロック周期にわたって安定している必要があります。

ビット 2:1 **CKPOL** : クロック極性

- LPTIM のクロックが、外部クロックソースによって供給される場合 :
- LPTIM のクロックが外部クロックソースによって供給されるとき、CKPOL ビットは、カウンタによって使用されるアクティブエッジを設定するために使用されます。
- 00 : 立ち上がりエッジが、カウントに使用されるアクティブエッジです。
 - 01 : 立ち下がりエッジが、カウントに使用されるアクティブエッジです。
 - 10 : 両方のエッジがアクティブエッジです。外部クロック信号の両方のエッジがアクティブエッジとみなされるときには、LPTIM のクロックは内部クロックソースからも供給される必要があります、その周波数は外部クロック周波数の 4 倍以上である必要があります。
 - 11 : 使用できません。

LPTIM がエンコーダモードで設定されている（ENC ビットがセットされている）場合 :

- 00 : エンコーダサブモード 1 がアクティブです。
- 01 : エンコーダサブモード 2 がアクティブです。
- 10 : エンコーダサブモード 3 がアクティブです。

エンコーダモードのサブモードの詳細については、[セクション 24.4.12 : エンコーダモード](#)を参照してください。

ビット 0 **CKSEL** : クロックセレクタ

- CKSEL ビットは、LPTIM が使用するクロックソースを選択します :
- 0 : LPTIM のクロックは内部クロックソースによって供給されます（APB クロックまたは埋め込みオシレータ）。
 - 1 : LPTIM のクロックは、LPTIM 外部 Input1 を通じて外部クロックソースによって供給されます。

注意 : LPTIMx_CFGR レジスタの変更は、LPTIM が無効（ENABLE ビットが 0 にリセットされている）のときにのみ行う必要があります。

表 91. LPTIM 外部トリガ接続

| TRIGSEL | 外部トリガ |
|-----------|----------------|
| ext_trig0 | PB6 または PC3 |
| ext_trig1 | RTC アラーム A |
| ext_trig2 | RTC アラーム B |
| ext_trig3 | RTC_TAMP1 入力検出 |
| ext_trig4 | RTC_TAMP2 入力検出 |
| ext_trig6 | COMP1_OUT |
| ext_trig7 | COMP2_OUT |

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

24.6.5 LPTIM 制御レジスタ（LPTIMx_CR）

アドレスオフセット：0x10

リセット値：0x0000 0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|------|-------------|-------------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNTSTR T | SNGSTR T | ENABLE |
| | | | | | | | | | | | | | rw | rw | rw |

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

ビット 2 CNTSTR：連続モードでタイマ開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。
ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM は連続モードで開始します。
ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、タイマは連続モードで開始します。
単一パルスモードでのカウント中にこのビットがセットされた場合、LPTIMx_ARR レジスタと LPTIMx_CNT レジスタが次に一致したときにタイマは停止せず、LPTIM カウンタは連続モードでのカウントを続行します。
このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 1 SNGSTR：シングルモードで LPTIM 開始

このビットは、ソフトウェアによってセットされ、ハードウェアによってクリアされます。
ソフトウェア開始（TRIGEN[1:0] = 00）の場合、このビットをセットすると、LPTIM はシングルパルスモードで開始します。
ソフトウェア開始が無効（TRIGEN[1:0] が 00 以外）の場合、このビットをセットすると、外部トリガが検出されるとすぐに、LPTIM はシングルパルスモードで開始します。
LPTIM が連続カウントモードのときにこのビットがセットされた場合、LPTIM は LPTIMx_ARR レジスタと LPTIMx_CNT レジスタが次に一致したときに停止します。
このビットをセットできるのは、LPTIM が有効なときだけです。ハードウェアによって自動的にリセットされます。

ビット 0 ENABLE：LPTIM 有効

ENABLE ビットは、ソフトウェアによってセット／クリアされます。
0：LPTIM は無効です。
1：LPTIM は有効です。

24.6.6 LPTIM 比較レジスタ（LPTIMx_CMP）

アドレスオフセット：0x14

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **CMP**：比較値。

CMP は、LPTIM によって使用される比較値です。

LPTIMx_CMP レジスタの内容の変更は、LPTIM が有効（ENABLE ビットが 1 にセットされている）のときにのみ行う必要があります。

24.6.7 LPTIM 自動再ロードレジスタ（LPTIMx_ARR）

アドレスオフセット：0x18

リセット値：0x0000 0001

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **ARR**：自動再ロード値。

ARR は、LPTIM の自動再ロード値です。

この値は、CMP[15:0] 値より大きくなければなりません。

LPTIMx_ARR レジスタの内容の変更は、LPTIM が有効（ENABLE ビットが 1 にセットされている）のときにのみ行う必要があります。

24.6.8 LPTIM カウンタレジスタ（LPTIMx_CNT）

アドレスオフセット：0x1C

リセット値：0x0000 0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:0 **CNT**：カウンタ値。

LPTIM が非同期クロックで実行しているとき、LPTIMx_CNT レジスタを読み出すと、信頼できない値が返されることがあります。したがって、この場合、2 つの連続した読み出しアクセスを実行して、返された 2 つの値が同じかどうかを確認する必要があります。

24.6.9 LPTIM レジスタマップ

次の表に LPTIM レジスタの一覧を示します。

表 92. LPTIM レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-------------|------|------|------|------|------|------|------|------|-----------|---------|--------|------|--------|--------|------|------|-----------|------|------|------|------|------|------|------|--------|------|---------|---------|-----------|-------|--------|------|
| 0x00 | LPTIMx_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWN | UP | AROK | CMPOK | EXTTRIG | ARM | CMPM | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | LPTIMx_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWNCF | UPCF | ARROKCF | CMPOKCF | EXTTRIGCF | ARMCF | CMPMCF | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | LPTIMx_IER | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DOWNIE | UPIE | ARROKIE | CMPOKIE | EXTTRIGIE | ARMIE | CMPMIE | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | LPTIMx_CFGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ENC | COUNTMODE | PRELOAD | WAVPOL | WAVE | TIMOUT | TRIGEN | Res. | Res. | TRIGSEL | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TRGFLT | Res. | Res. | CKFLT | CKPOL | CKSEL | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x10 | LPTIMx_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x14 | LPTIMx_CMP | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CMP[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x18 | LPTIMx_ARR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ARR[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0x1C | LPTIMx_CNT | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CNT[15:0] | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

25 独立型ウォッチドッグ (IWDG)

25.1 概要

デバイスは、内蔵ウォッチドッグペリフェラルを搭載しており、使用上、高い安全レベル、タイミングの正確さ、および柔軟性を兼ね備えています。独立型ウォッチドッグペリフェラルは、ソフトウェア障害による誤動作を検出および解決し、カウンタが与えられたタイムアウト値に達すると、システムリセットをトリガします。

独立型ウォッチドッグ (IWDG) は、独自の低速クロック (LSI) によってクロック供給されるので、メインクロックに障害があってもアクティブなままです。

IWDG は、メインアプリケーションの外部で、完全に独立したプロセスとして実行するウォッチドッグが必要な場合に最適ですが、タイミング精度が低いという制約があります。ウィンドウ型ウォッチドッグの詳細については、[セクション 26 \(568 ページ\)](#) を参照してください。

25.2 IWDG の主な機能

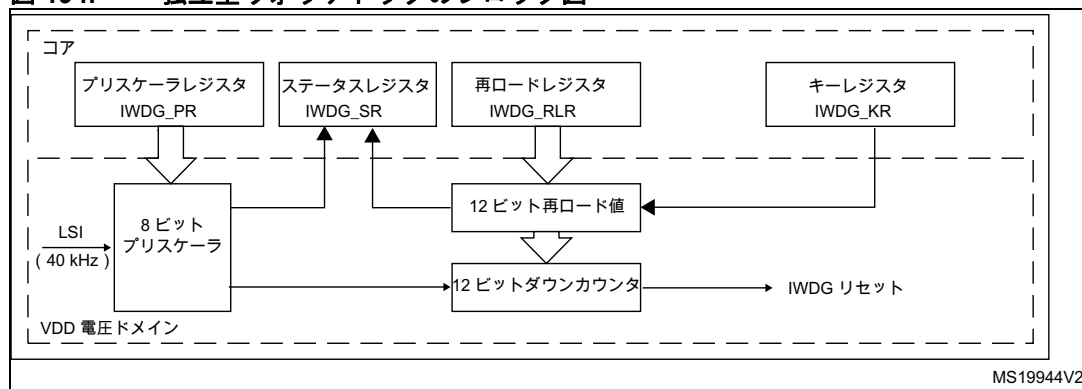
- フリーランニングダウンカウンタ
- 独立した RC オシレータからのクロック供給 (STANDBY および STOP モードで動作可能)
- 条件付きリセット
 - ダウンカウンタの値が 000 h より小さくなったときにリセット (ウォッチドッグが有効な場合)。
 - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット (ウォッチドッグが有効な場合)。

25.3 IWDG の機能説明

25.3.1 IWDG ブロック図

[図 194](#) に、独立型ウォッチドッグモジュールの機能ブロックを示します。

図 194. 独立型ウォッチドッグのブロック図



注： ウォッチドッグ機能は、STOP モードおよびSTANDBY モードでも機能する CORE 電圧ドメインに実装されています。

キーレジスタ (IWDG_KR) に値 0x0000 CCCC が書き込まれることによって独立型ウォッチドッグが開始すると、カウンタはリセット値 0xFFFF からカウントダウンを開始します。カウント値の終わり (0x000) に達すると、リセット信号が生成されます (IWDG_reset)。

IWDG_KR レジスタにキー値 0x0000 AAAA が書き込まれると、IWDG_RLR の値がカウンタに再ロードされ、ウォッチドッグのリセットが防止されます。

25.3.2 ウィンドウオプション

IWDG は、IWDG_WINR レジスタに適切なウェインドウをセットすることによって、ウィンドウ型ウォッチドッグとしても機能します。

カウンタがウィンドウレジスタ (IWDG_WINR) に格納された値より大きい間に再ロード操作が行われると、リセットが生成されます。

IWDG_WINR のデフォルト値は 0x0000 0FFF です。この値が更新されない場合は、ウィンドウオプションは無効にされます。

ウィンドウ値が変わるとすぐに再ロード操作が行われ、ダウンカウンタを IWDG_RLR 値にリセットし、次の再ロードを生成するためのサイクル数計算を容易にします。

ウィンドウオプションが有効な場合の IWDG の設定

1. IWDG_KR レジスタに 0x0000 CCCC を書き込むことによって、IWDGを有効にします。
2. IWDG_KR レジスタに 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG_PR を 0 から 7 までプログラムすることによって、IWDG プリスケールに書き込みを行います。
4. 再ロードレジスタ (IWDG_RLR) に書き込みます。
5. レジスタが更新されるのを待ちます (IWDG_SR = 0x0000 0000)。
6. ウィンドウレジスタ IWDG_WINR に書き込みます。これにより、カウンタ値 IWDG_RLR が自動的にリフレッシュされます。

注： ウィンドウ値を書き込むことで、IWDG_SR が“0x0000 0000”にセットされた時点でカウンタ値をRLRでリフレッシュすることができます。

ウィンドウオプションが無効な場合の IWDG の設定

ウィンドウオプションが使用されていない場合、IWDG は以下のように設定することができます。

1. IWDG_KR レジスタに 0x0000 CCCC を書き込むことによって、IWDGを有効にします。
2. IWDG_KR レジスタに 0x0000 5555 を書き込むことによって、レジスタのアクセスを有効にします。
3. IWDG_PR を 0 から 7 までプログラムすることによって、IWDG のプリスケールに書き込みを行います。
4. 再ロードレジスタ (IWDG_RLR) に書き込みます。
5. レジスタが更新されるのを待ちます (IWDG_SR = 0x0000 0000)。
6. カウンタ値を IWDG_RLR (IWDG_KR = 0x0000 AAAA) でリフレッシュします。

25.3.3 ハードウェアウォッチドッグ

デバイスのオプションビットを使って「ハードウェアウォッチドッグ」機能が有効化されると、ウォッチドッグは電源投入時に自動的に有効になり、カウンタがカウントの終わりに達する前にソフトウェアによってキーレジスタへ書き込まれない限り、またはダウンカウンタがウィンドウ内に再ロードされた場合は、リセットを生成します。

25.3.4 レジスタのアクセス保護

IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへの書き込みアクセスは保護されます。これらを変更するには、まず、IWDG_KR レジスタにコード 0x0000 5555 を書き込む必要があります。このレジスタに別の値を書き込むと、シーケンスがブレイクされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作 (0x0000 AAAA の書き込み) であることを意味します。ステータスレジスタは、プリスケアラの更新、あるいはダウンカウンタ再ロード値やウィンドウ値の更新が行われていることを示すために使用されます。

25.3.5 デバッグモード

マイクロコントローラがデバッグモードになると (コアは停止状態)、IWDG カウンタは、DBG モジュールの DBG_IWDG_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。

25.4 IWDG レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

25.4.1 キーレジスタ (IWDG_KR)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000 (STANDBY モードによりリセットされる)

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **KEY[15:0]** : キー値（書き込み専用、読み出しでは 0x0000）

これらのビットには、ソフトウェアによって一定間隔でキー値 0xAAAA が書き込まれなければなりません。そうしないと、カウンタが 0 に達した時点でウォッチドッグがリセットを生成します。

キー値 0x5555 を書き込むことによって、IWDG_PR、IWDG_RLR、および IWDG_WINR レジスタへのアクセスが可能になります（[セクション 25.3.4 : レジスタのアクセス保護](#)を参照）。

キー値 CCCCh を書き込むと、ウォッチドッグが開始します（ハードウェアウォッチドッグオプションが選択されている場合を除く）。

25.4.2 プリスケアラレジスタ (IWDG_PR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2:0 **PR[2:0]** : プリスケアラ分周回路

これらのビットは、書き込みアクセス保護されています（[セクション 25.3.4 : レジスタのアクセス保護](#)を参照）。カウンタクロックを供給するプリスケアラ分周回路を選択するようにソフトウェアで書き込まれます。プリスケアラ分周比を変更できるようにするには、IWDG_SR レジスタの PVU ビットをリセットする必要があります。

- 000 : 4 分周
- 001 : 8 分周
- 010 : 16 分周
- 011 : 32 分周
- 100 : 64 分周
- 101 : 128 分周
- 110 : 256 分周
- 111 : 256 分周

注 : このレジスタを読み出すと、VDD 電圧ドメインからプリスケアラ値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの PVU ビットがリセットされているときのみとなります。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

25.4.3 再ロードレジスタ (IWDG_RLR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0FFF (STANDBY モードによりリセットされる)

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **RL[11:0]** : ウォッチドッグカウンタ再ロード値

これらのビットは、書き込みアクセス保護されています ([セクション 25.3.4](#) を参照)。IWDG_KR レジスタに値 0xAAAA が書き込まれるたびにウォッチドッグカウンタにロードされる値を定義するために、ソフトウェアで書き込まれます。ウォッチドッグカウンタは、この値からカウントダウンします。タイムアウトまでの時間は、この値とクロックプリスケールによって決まります。タイムアウトに関する詳細はデータシートを参照してください。

再ロード値を変更できるようにするには、IWDG_SR レジスタの RVU ビットをリセットする必要があります。

注 : このレジスタを読み出すと、VDD 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合、この値は最新でないか、有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの RVU ビットがリセットされているときのみとなります。

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

25.4.4 ステータスレジスタ (IWDG_SR)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000 (STANDBY モードによりリセットされません)

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **WVU** : ウォッチドッグカウンタウィンドウ値の更新

このビットは、ウィンドウ値の更新が進行中であることを示すために、ハードウェアによってセットされます。 V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます (最大 5 RC 40 kHz サイクルかかる)。

ウィンドウ値は、WVU ビットがリセットされているときのみ更新できます。

このビットは、一般のウィンドウが 1 の場合のみ生成されます。

ビット 1 **RVU** : ウォッチドッグカウンタ再ロード値の更新

このビットは、再ロード値の更新が進行中であることを示すために、ハードウェアによってセットされます。 V_{DD} 電圧ドメインで再ロード値の更新操作が完了したときに、ハードウェアによってリセットされます (最大 5 RC 40 kHz サイクルかかる)。

再ロード値は、RVU ビットがリセットされているときのみ更新できます。

ビット 0 **PVU** : ウォッチドッグプリスケアラ値の更新

このビットは、プリスケアラ値の更新が進行中であることを示すために、ハードウェアによってセットされます。 V_{DD} 電圧ドメインでプリスケアラ更新操作が完了したときに、ハードウェアによってリセットされます (最大 5 RC 40 kHz サイクルかかる)。

プリスケアラ値は、PVU ビットがリセットされているときのみ更新できます。

注 : 複数の再ロード値、プリスケアラ値、またはウィンドウ値がアプリケーションで使用される場合は、それぞれ、再ロード値を変更する前に RVU ビットがリセットされるまで待つか、プリスケアラ値を変更する前に PVU ビットがリセットされるまで待つか、またはウィンドウ値を変更する前に WVU ビットがリセットされるまで待つ必要があります。ただし、プリスケアラ値、再ロード値、またはウィンドウ値を更新した後は、RVU、PVU、または WVU がリセットされるのを待たずに、コード実行を続けることができます (低電力モードに入った場合を除く)。

25.4.5 ウィンドウレジスタ (IWDG_WINR)

アドレスオフセット : 0x10

リセット値 : 0x0000 0FFF (STANDBY モードによりリセットされる)

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

ビット 31:12 予約済みであり、リセット値に保持する必要があります。

ビット 11:0 **WIN[11:0]** : ウォッチドッグカウンタウィンドウ値

これらのビットは、書き込みアクセス保護されています ([セクション 25.3.4](#) を参照)。これらのビットは、ダウンカウンタと比較されるウィンドウ値の上限を含みます。

リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x0 よりも大きい間にダウンカウンタを再ロードする必要があります。

再ロード値を変更できるようにするには、IWDG_SR レジスタの WVU ビットをリセットする必要があります。

注： このレジスタを読み出すと、V_{DD} 電圧ドメインから再ロード値が返されます。このレジスタへの書き込み操作が進行中の場合には、この値は有効でないことがあります。このため、このレジスタから読み出された値が有効なのは、IWDG_SR レジスタの WVU ビットがリセットされているときのみとなります。

25.4.6 IWDG レジスタマップ

次の表に、IWDG レジスタマップとリセット値を示します。

表 93. IWDG レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|-----------|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----------|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|-----|-----|---------|-----|-----|-----|
| 0x00 | IWDG_KR | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | KEY[15:0] | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x04 | IWDG_PR | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | PR[2:0] | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | |
| 0x08 | IWDG_RLR | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | RL[11:0] | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| 0x0C | IWDG_SR | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | WVU | RVU | PVU |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | |
| 0x10 | IWDG_WINR | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | WIN[11:0] | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

26 システムウィンドウ型ウォッチドッグ (WWDG)

26.1 概要

システムウィンドウ型ウォッチドッグ (WWDG) は、通常、外部の影響や予期しない論理条件などによって発生し、アプリケーションプログラムを正常なシーケンスから逸脱させるソフトウェア障害の発生を検出するために使用されます。ウォッチドッグ回路は、T6 ビットがクリアされる前にプログラムがダウンカウンタの内容をリフレッシュしない限り、プログラムされた時間の経過後に MCU リセットを生成します。MCU リセットは、ダウンカウンタがウィンドウレジスタ値に達する前に 7 ビットのダウンカウンタの値 (制御レジスタ内) がリフレッシュされた場合にも生成されます。このことは、限られた時間枠 (time-window) の間にカウンタがリフレッシュされなければならないことを意味します。

WWDG クロックは、APB1 クロックから分周され、また設定可能な時間枠 (time-window) があるので、これをプログラムしてアプリケーション動作の異常な進み・遅れを検出できます。

WWDG は、正確な時間枠内で反応するウォッチドッグが必要なアプリケーションに適しています。

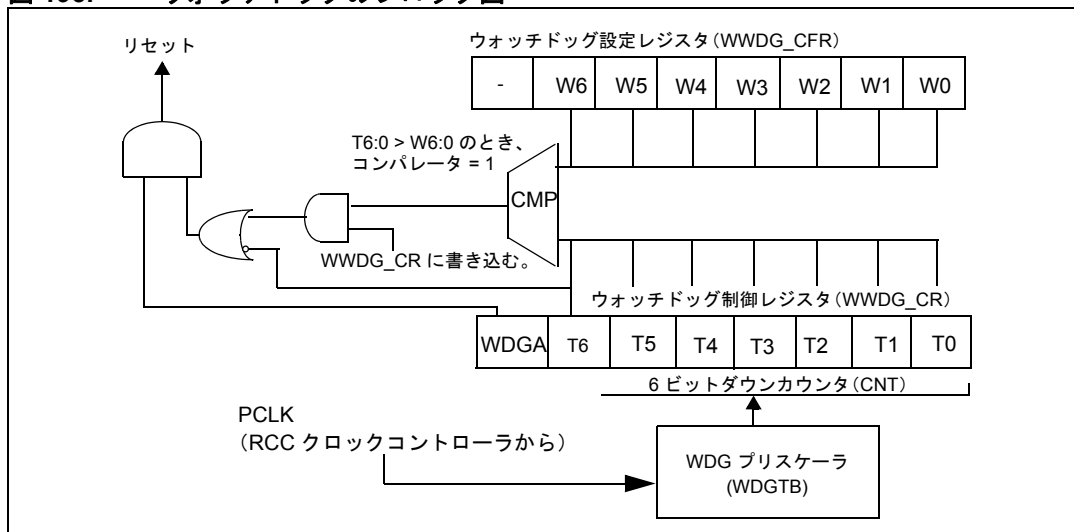
26.2 WWDG の主な機能

- プログラム可能なフリーランニングダウンカウンタ
- 条件付きリセット
 - ダウンカウンタの値が 0x40 より小さくなったときにリセット (ウォッチドッグが有効な場合)。
 - ダウンカウンタがウィンドウ外で再ロードされた場合にリセット (ウォッチドッグが有効な場合) (図 196 を参照)。
- 早期ウェイクアップ割り込み (EWI) : ダウンカウンタが 0x40 になったときにトリガ (有効であり、ウォッチドッグがアクティブな場合)

26.3 WWDG の機能説明

ウォッチドッグが有効 (WWDG_CR レジスタの WDGA ビットがセットされている) な場合、7 ビットのダウンカウンタ (T[6:0] ビット) が 0x40 に達して 0x3F に戻った (T6 がクリアされた) 時点で、リセットを開始します。カウンタがウィンドウレジスタに格納された値より大きい間にソフトウェアがカウンタを再ロードした場合にも、リセットが生成されます。

図 195. ウォッチドッグのブロック図



アプリケーションプログラムは、通常動作時には定期的に WWDG_CR レジスタへの書き込みを行って、MCU リセットを防ぐ必要があります。この操作は、カウンタの値がウィンドウレジスタの値より小さいときに限られます。WWDG_CR レジスタに格納される値は、0xFF から 0xC0 の間でなければなりません。

26.3.1 ウォッチドッグの有効化

ウォッチドッグはリセット後は常に無効です。これを有効にするには、WWDG_CR レジスタの WDGA ビットをセットします。この後は、リセット以外の方法でウォッチドッグを無効にすることはできません。

26.3.2 ダウンカウンタの制御

このダウンカウンタはフリーランニングであり、ウォッチドッグが無効状態であってもカウントダウンを続けます。ウォッチドッグを有効にするときには、T6 ビットをセットして、ただちにリセットが生成されるのを防ぐ必要があります。

T[5:0] ビットは、ウォッチドッグがリセットを生成するまでの時間遅延を表すインクリメント数を含みます。WWDG_CR レジスタへの書き込み時のプリスケアラの状態が不明なため、このタイミングは最小値から最大値の間で変化します（図 196 を参照）。設定レジスタ (WWDG_CFR) は、ウィンドウの上限値を含みます。リセットを防ぐには、カウンタの値がウィンドウレジスタの値よりも小さく、0x3F よりも大きい間にダウンカウンタを再ロードする必要があります。図 196 に、ウィンドウ型ウォッチドッグのプロセスを示します。

注： T6 ビットを使用して、ソフトウェアリセットを生成することができます (WDGA ビットはセット、T6 ビットはクリアされる)。

26.3.3 高度なウォッチドッグ割り込み機能

実際にリセットが生成される前に特定の安全処理やデータロギングを実施する必要がある場合は、早期ウェイクアップ割り込み (EWI) が使用できます。EWI 割り込みは、WWDG_CFR レジスタの EWI ビットをセットすることによって有効になります。ダウンカウンタ値が 0x40 に到達すると、EWI 割り込みが生成され、対応する割り込みサービスルーチン (ISR) を使用してデバイスをリセットする前に特定の処理（通信やデータロギングなど）をトリガすることができます。

アプリケーションによっては、EWI 割り込みを使用して、WWDG リセットを生成せずにソフトウェアのシステムチェックやシステム復旧／グレースフルデグラデーションを管理することができます。この場合、対応する割り込みサービスルーチン (ISR) で WWDG カウンタを再ロードし、WWDG リセットを回避してから必要な操作をトリガしてください。

EWI 割り込みは、WWDG_SR レジスタの EWIF ビットに“0”を書き込むことによってクリアされます。

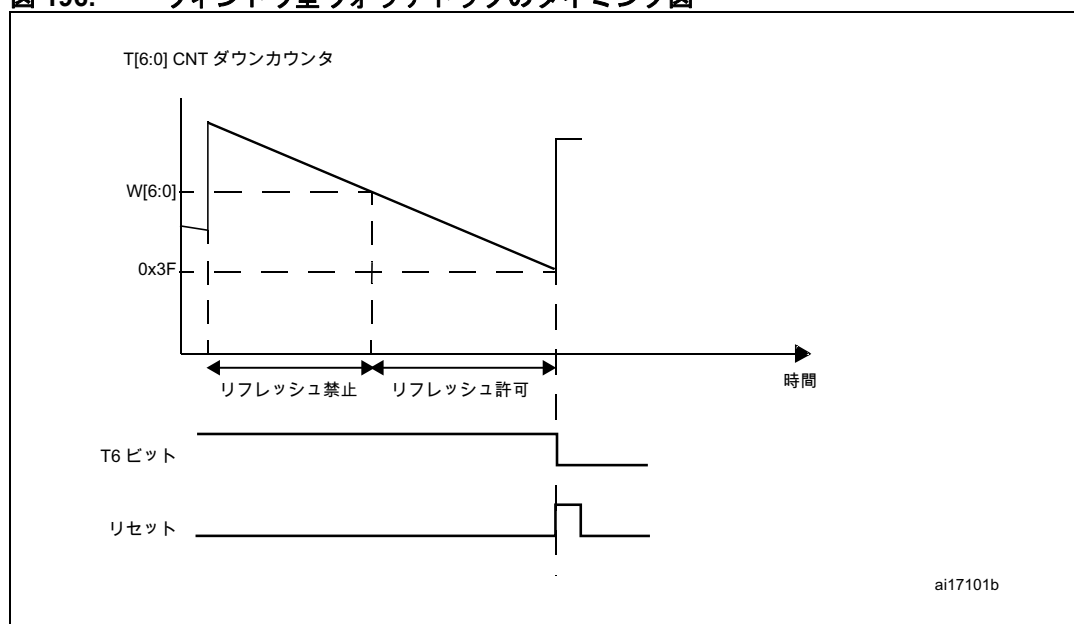
注： 優先順位の高いタスクにおけるシステムロックなどによって EWI 割り込みが使用できない場合、最終的には WWDG リセットが生成されます。

26.3.4 ウォッチドッグタイムアウトをプログラムする方法

図 196 の式を使用して、WWDG のタイムアウトを計算することができます。

警告： WWDG_CR レジスタに書き込むときには、ただちにリセットされるのを防ぐために、常に T6 ビットに 1 を書き込んでください。

図 196. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値は次の式で算出されます。

$$t_{\text{WWDG}} = t_{\text{PCLK1}} \times 4096 \times 2^{\text{WDGTB}[1:0]} \times (t[5:0] + 1) \quad (\text{ms})$$

ここで、

t_{WWDG} : WWDG タイムアウト

t_{PCLK1} : APB1 クロック周期の測定値 (ms)

T_{WWDG} の最小値と最大値については、データシートを参照してください。

26.3.5 デバッグモード

マイクロコントローラがデバッグモードになると (Cortex®-M0+ コアは停止状態)、WWDG カウンタは、DBG モジュールの DBG_WWDG_STOP 設定ビットに応じて、通常どおりに動作を続けるか、または停止します。詳細については、[セクション 33.9.2 : タイマ、ウォッチドッグ、および I2C のデバッグサポート](#)を参照してください。

26.4 WWDG レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#)を参照してください。

ペリフェラルレジスタには、ハーフワード (16 ビット) またはワード (32 ビット) 単位でアクセスする必要があります。

26.4.1 制御レジスタ (WWDG_CR)

アドレスオフセット : 0x00

リセット値 : 0x0000 007F

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

ビット 31:8 リセット値に保持する必要があります。

ビット 7 **WDGA** : 有効化ビット

このビットは、ソフトウェアでセットされ、リセット後はハードウェアによってのみクリアされます。
WDGA = 1 のとき、ウォッチドッグはリセットを生成できます。

- 0 : ウォッチドッグは無効です。
- 1 : ウォッチドッグは有効です。

ビット 6:0 **T[6:0]** : 7 ビットカウンタ (MSB から LSB まで)

これらのビットは、ウォッチドッグカウンタの値を含みます。(4096 x 2^{WDGTB[1:0]}) PCLK サイクルごとにデクリメントされます。0x40 に達して 0x3F に戻ると (T6 がクリアされると)、リセットが生成されます。

26.4.2 設定レジスタ (WWDG_CFR)

アドレスオフセット : 0x04

リセット値 : 0x0000 007F

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

ビット 31:10 リセット値に保持する必要があります。

ビット 9 **EWI** : 早期ウェイクアップ割り込み

このビットがセットされているときには、カウンタの値が 0x40 に達したときに割り込みが発生します。
この割り込みは、リセット後にハードウェアによってのみクリアされます。

ビット 8:7 **WDGTB[1:0]** : タイマーベース

プリスケアラのタイムベースは、次のように変更できます。

- 00 : CK カウンタクロック (PCLK/4096) 1 分周
- 01 : CK カウンタクロック (PCLK/4096) 2 分周
- 10 : CK カウンタクロック (PCLK/4096) 4 分周
- 11 : CK カウンタクロック (PCLK/4096) 8 分周

ビット 6:0 **W[6:0]** : 7 ビットウィンドウ値

これらのビットは、ダウンカウンタと比較されるウィンドウ値を含みます。

26.4.3 ステータスレジスタ (WWDG_SR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

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

ビット 31:1 リセット値に保持する必要があります。

ビット 0 **EWIF** : 早期ウェイクアップ割り込みフラグ

このビットは、カウンタの値が 0x40 に達したときにハードウェアによってセットされます。“0”を書き込んでソフトウェアでクリアする必要があります。“1”を書き込んでも、ビットの値は変化しません。このビットは、割り込みが有効でない場合にもセットされます。

26.4.4 WWDG レジスタマップ

次の表に、WWDG のレジスタマップとリセット値を示します。

表 94. WWDG レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | | |
|-----------|----------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|--------|--------|--------|------|------|------|------|------|------|---|--|--|--|--|--|
| 0x00 | WWDG_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WDGA | T[6:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | |
| 0x04 | WWDG_CFR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EWI | WDGTB1 | WDGTB0 | W[6:0] | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | |
| 0x08 | WWDG_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EWIF | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

27 リアルタイムクロック (RTC)

27.1 概要

RTC は、あらゆる低電力モードを管理する自動ウェイクアップ機能を提供します。

本リアルタイムクロック (RTC) は、独立した BCD タイマ／カウンタです。RTC は、プログラム可能なアラーム割り込み機能を備えた時刻クロック／カレンダーを搭載しています。

また、割り込み機能を備えたプログラム可能な周期的ウェイクアップフラグも搭載しています。

2 つの 32 ビットレジスタには、2 進化 10 進数形式 (BCD) で表現した秒、分、時 (12 時間または 24 時間形式)、曜日、日、月、年が含まれています。サブセカンドの値もバイナリ形式で利用できます。

28 日、29 日 (うるう年)、30 日、31 日の補正は、自動的に行われます。サマータイム補正も行われます。

サブセカンド、秒、分、時、曜日、日付のプログラム可能なアラームを備えた 32 ビットレジスタが追加されています。

クリスタルオシレータ精度の偏差を補正するために、デジタル較正機能が利用可能です。

RTC ドメインリセット後、すべての RTC レジスタは、起こりうる不要な書き込みアクセスから保護されます。

供給電圧が動作範囲内にある間は、デバイスのステータス (実行モード、低電力モード、またはリセット中) に関係なく、RTC が停止することはありません。

27.2 RTC の主な機能

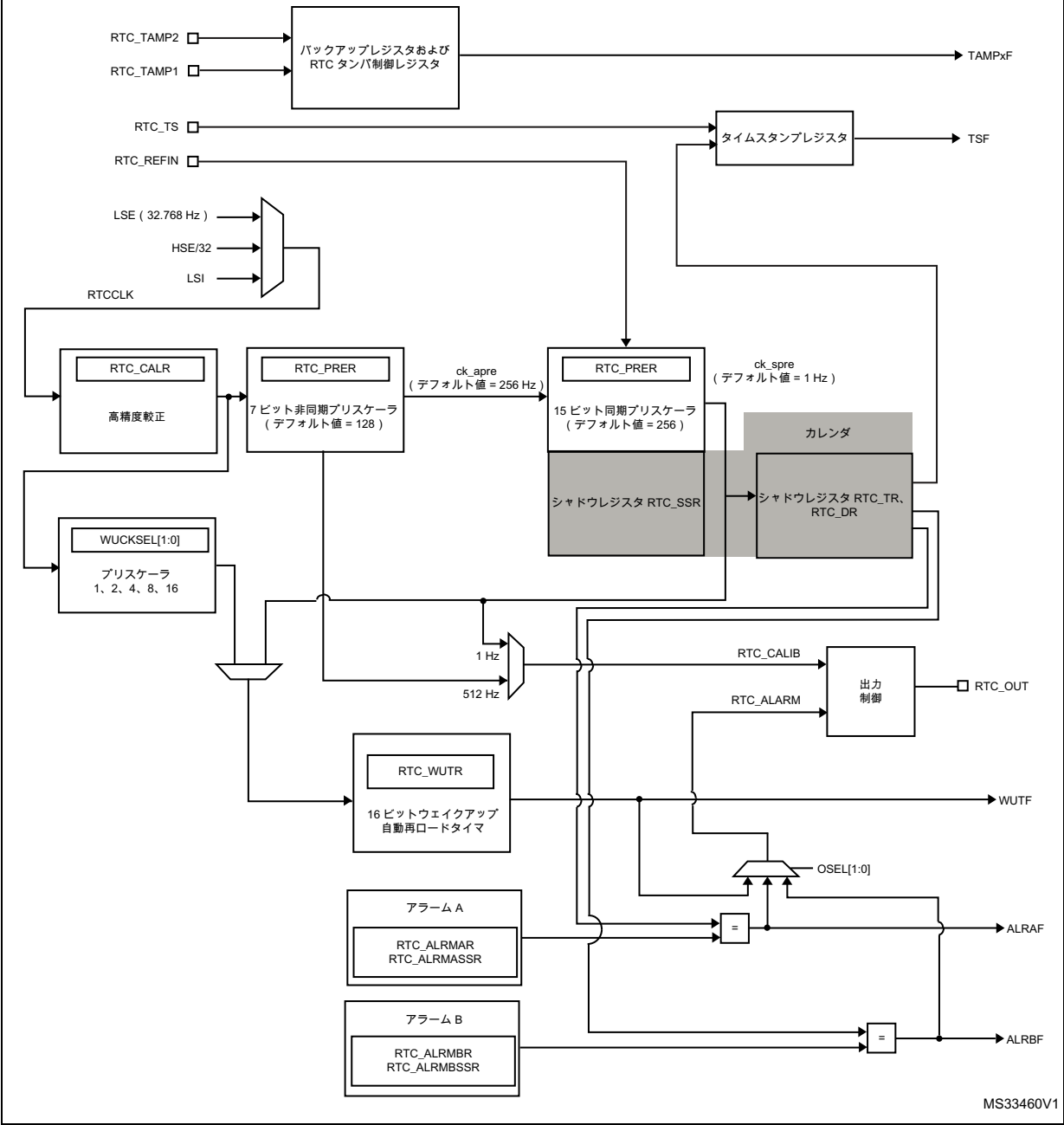
RTC ユニットの主な機能を以下に示します (図 197 : RTC ブロック図を参照)。

- サブセカンド、秒、分、時 (12 または 24 時間形式)、曜日、日、月、年に対応するカレンダー。
- ソフトウェアでプログラム可能なサマータイム補正。
- 割り込み機能を備えたプログラム可能なアラーム。アラームは、カレンダー項目のどの組み合わせでもトリガ可能。
- 自動ウェイクアップ割り込みをトリガする周期的なフラグを生成する自動ウェイクアップユニット。
- リファレンスクロック検出: より正確な秒のクロックソース (50 または 60 Hz) の使用で、カレンダーの精度を向上。
- サブセカンドシフト機能を使用する外部クロックとの正確な同期。
- デジタル較正回路 (周期的なカウンタ修正): 数秒の較正時間範囲で得られる 0.95 ppm の精度。
- イベントを保存するタイムスタンプ機能。
- 設定可能なフィルタおよび内部ブルアップのあるタンパ検出イベント。
- 以下に示すマスク可能な割り込み／イベント。
 - － アラーム A
 - － アラーム B
 - － ウェイクアップ割り込み
 - － タイムスタンプ
 - － タンパ検出
- バックアップレジスタ。

27.3 RTC の機能説明

27.3.1 RTC ブロック図

図 197. RTC ブロック図



RTC には次の要素が含まれます。

- 2 本のアラーム
- 2 つの タンパイイベント
- 5 x 32 ビットバックアップレジスタ
- オルタネート機能出力：次の 2 つの出力のうち 1 つを選択する RTC-OUT です。
 - RTC_CALIB : 512 Hz または 1 Hz のクロック出力 (LSE 周波数 32.768 kHz の場合)。この出力は、RTC_CR レジスタの COE ビットをセットして有効にします。
 - RTC_ALARM : この出力は、アラーム A、アラーム B、またはウェイクアップ出力を選択する RTC_CR レジスタの OSEL[1:0] ビットを設定することで有効にします。
- オルタネート機能入力：
 - RTC_TS : タイムスタンプイベント
 - RTC_TAMP1 : タンパ 1 イベント検出
 - RTC_TAMP2 : タンパ 2 イベント検出
 - RTC_REFIN : 50 または 60 Hz のリファレンスクロック入力

27.3.2 RTC によって制御される GPIO

RTC_OUT、RTC_TS、および RTC_TAMP1 は、同一ピン (PC13) 上に配置されます。PC13 ピンの設定は、PC13 GPIO の設定にかかわらず、RTC によって制御されます。

出力形式は表 95 に示す優先順位に従います。

表 95. RTC ピン PC13 の設定⁽¹⁾

| PC13 ピンの 設定と機能 | OSEL[1:0] ビット (RTC_ALARM 出力イネーブル) | COE ビット (RTC_CALIB 出力イネーブル) | RTC_OUT _RMP ビット | RTC_ALARM_ TYPE ビット | TAMP1E ビット (RTC_TAMP1 入力 イネーブル) | TSE ビット (RTC_TS 入力 イネーブル) |
|--|--|-----------------------------------|------------------------|---------------------------|--|------------------------------------|
| RTC_ALARM 出力 OD | 01、10、または 11 | 無視 | 0 | 0 | 無視 | 無視 |
| | | 1 | 1 | | | |
| RTC_ALARM 出力 PP | 01、10、または 11 | 無視 | 0 | 1 | 無視 | 無視 |
| | | 1 | 1 | | | |
| RTC_CALIB 出力 PP | 00 | 1 | 0 | 無視 | 無視 | 無視 |
| RTC_TAMP1 入力 フローティング | 00 | 0 | 無視 | 無視 | 1 | 0 |
| | 00 | 1 | 1 | | | |
| | 01、10、または 11 | 0 | | | | |
| RTC_TS および RTC_TAMP1 入力 フローティング | 00 | 0 | 無視 | 無視 | 1 | 1 |
| | 00 | 1 | 1 | | | |
| | 01、10、または 11 | 0 | | | | |

表 95. RTC ピン PC13 の設定⁽¹⁾ (続き)

| PC13 ピンの 設定と機能 | OSEL[1:0] ビット (RTC_ALARM 出カインーブル) | COE ビット (RTC_CALIB 出カインーブル) | RTC_OUT _RMP ビット | RTC_ALARM_ TYPE ビット | TAMP1E ビット (RTC_TAMP1 入力 インーブル) | TSE ビット (RTC_TS 入力 インーブル) |
|----------------------------|--|-----------------------------------|------------------------|---------------------------|--|------------------------------------|
| RTC_TS 入力 フローティング | 00 | 0 | 無視 | 無視 | 0 | 1 |
| | 00 | 1 | 1 | | | |
| | 01、10、または 11 | 0 | | | | |
| ウェイクアップ ピンまたは標準 GPIO | 00 | 0 | 無視 | 無視 | 0 | 0 |
| | 00 | 1 | 1 | | | |
| | 01、10、または 11 | 0 | | | | |

1. OD : オープンドレイン、PP : プッシュプル

さらに、RTC_OUT_RMP ビットにより、RTC_OUT を PB14 ピンに再配置することも可能です。この場合、PB14 GPIO レジスタを適切なタイプのオルタネート機能として設定してください。再配置機能は表 96 に示されています。

表 96. RTC_OUT の配置

| OSEL[1:0] ビット (RTC_ALARM 出力イネーブル) | COE ビット (RTC_CALIB 出力 イネーブル) | RTC_OUT_RMP ビット | PC13 上の RTC_OUT | PB14上の RTC_OUT |
|---|------------------------------------|--------------------|--------------------|-------------------|
| 00 | 0 | 0 | - | - |
| 00 | 1 | | RTC_CALIB | - |
| 01、10、または 11 | 無視 | | RTC_ALARM | - |
| 00 | 0 | 1 | - | - |
| 00 | 1 | | - | RTC_CALIB |
| 01、10、または 11 | 0 | | - | RTC_ALARM |
| 01、10、または 11 | 1 | | RTC_ALARM | RTC_CALIB |

27.3.3 クロックとプリスケアラ

RTC クロックソース (RTCCLK) は、LSE クロック、LSI オシレータクロック、HSE クロックのうちから、クロックコントローラを介して選択されます。RTC クロックソースの設定に関する詳細は、[セクション 7 : リセットおよびクロック制御 \(RCC\)](#) を参照してください。

プログラム可能なプリスケアラステージで、カレンダーの更新に使用する 1 Hz のクロックを生成します。消費電力を最少に抑えるため、プリスケアラは以下に示す 2 つのプログラム可能なプリスケアラに分割されます ([図 197 : RTC ブロック図](#)を参照)。

- RTC_PRER レジスタの PREDIV_A ビットで設定される 7 ビットの非同期プリスケアラ
- RTC_PRER レジスタの PREDIV_S ビットで設定される 15 ビットの同期プリスケアラ

注 : *両方のプリスケアラを使用する場合は、非同期プリスケアラを高い値に設定して消費を最低限に抑えることをお勧めします。*



LSE 周波数 32.768 kHz で 1 Hz (ck_spre) の内部クロック周波数を得るため、非同期プリスケアラ分周比は 128、同期プリスケアラの分周比は 256 に設定されます。

最低分周比は 1、最大分周比は 2^{22} です。

これは、約 4 MHz の最大入力周波数に相当します。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_APRE} = \frac{f_{RTCCLK}}{PREDIV_A + 1}$$

ck_apre クロックは、サブセカンドダウンカウンタであるバイナリ RTC_SSR にクロックを供給するために使用されます。値がゼロになると、RTC_SSR は、PREDIV_S の内容で再ロードされます。

f_{ck_apre} は、次の式で与えられます。

$$f_{CK_SPRE} = \frac{f_{RTCCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

ck_spre クロックは、カレンダーの更新に、または 16 ビットウェイクアップ自動再ロードタイマのタイムベースとして使用できます。短いタイムアウト期間を得るため、16 ビットウェイクアップ自動再ロードタイマを、プログラム可能な 4 ビット非同期プリスケアラで分周した RTCCLK で動作させることもできます (詳細は [セクション 27.3.6 : 周期的自動ウェイクアップ](#) を参照)。

27.3.4 リアルタイムクロックとカレンダー

RTC カレンダーの時刻および日付レジスタには、PCLK (APB クロック) と同期するシャドウレジスタからアクセスします。同期持続の待ち時間を避けるため、これらのレジスタに直接アクセスすることもできます。

- サブセカンド用 RTC_SSR
- 時刻用 RTC_TR
- 日付用 RTC_DR

RTCCLK 2 サイクルごとに現在のカレンダー値がシャドウレジスタにコピーされ、RTC_ISR レジスタの RSF ビットがセットされます ([セクション 27.6.4 : RTC 初期化とステータスレジスタ \(RTC_ISR\)](#) を参照)。STOP モードおよび STANDBY モードでは、コピーは行われません。これらのモードが終了すると、RTCCLK 2 サイクル以内にシャドウレジスタが更新されます。

アプリケーションが、カレンダーレジスタを読み出す際、実際にはシャドウレジスタの内容にアクセスします。RTC_CR レジスタの BYPSHAD 制御ビットをセットすることにより、カレンダーレジスタに直接アクセスできます。デフォルトでは、このビットはクリアされており、ユーザはシャドウレジスタにアクセスします。

RTC_SSR、RTC_TR または RTC_DR レジスタを BYPSHAD = 0 の状態で読み出す際は、APB クロックの周波数 (f_{APB}) は、RTC クロック (f_{RTCCLK}) の周波数の 7 倍以上でなければなりません。

シャドウレジスタは、システムリセットによってリセットされます。

27.3.5 プログラム可能なアラーム

RTC ユニットの、以下に示すプログラム可能なアラーム、アラーム A およびアラーム B を搭載しています。以下に示すのはアラーム A に関する説明ですが、アラーム B についても同様です。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAE ビットを通じて有効にします。ALRAF は、カレンダーのサブセカンド、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASR および RTC_ALRMAR にプログラムされている値と一致する場合は 1 にセットされます。各カレンダー項目は、RTC_ALRMAR レジスタの MSKx ビットおよび RTC_ALRMASR レジスタの MASKSSx ビットで個別に選択できます。アラームの割り込みは、RTC_CR レジスタの ALRAIE ビットを通じて有効にします。

注意： 秒の項目が選択されている (RTC_ALRMAR で MSK0 ビットがリセットされている) 場合、正しい動作を保証するため、RTC_PRER レジスタでセットされる同期プリスケアラの分周比は 3 以上でなければなりません。

アラーム A および アラーム B (RTC_CR レジスタの OSEL[0:1] ビットで有効になっている場合) は、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

27.3.6 周期的自動ウェイクアップ

周期的ウェイクアップフラグは、16 ビットのプログラム可能な自動再ロードダウンカウンタによって生成されます。ウェイクアップタイマの範囲は 17 ビットまで拡張できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットを通じて有効にします。

ウェイクアップタイマクロック入力には、次のものが使用できます。

- 2、4、8、または 16 分周した RTC クロック (RTCCLK)
RTCCLK が LSE (32.768kHz) である場合、最小分解能 61 μ s で、ウェイクアップ割り込み周期を 122 μ s から 32 s の範囲で設定できます。
- ck_spre (通常は 1 Hz の内部クロック)
ck_spre 周波数が 1 Hz の場合、1 秒の分解能でウェイクアップ時間を 1 秒 からおよそ 36 時間までの範囲で設定できます。このプログラム可能な広い時間範囲は、2 つの部分に分かれます。
 - WUCKSEL[2:1] = 10 の場合は 1 秒から 18 時間、
 - WUCKSEL[2:1] = 11 の場合は約 18 時間から 36 時間です。後者の場合、16 ビットカウンタの現在値に 216 が加算されます。初期化シーケンスが完了すると ([ウェイクアップタイマのプログラミング \(581 ページ\)](#) を参照)、タイマがカウントダウンを開始します。ウェイクアップ機能が有効な場合、低電力モードでもカウントダウンはアクティブのままとなります。さらに、カウンタがゼロに到達すると、RTC_ISR レジスタの WUTF フラグがセットされ、ウェイクアップカウンタが再ロード値 (RTC_WUTR レジスタ値) で自動的に再ロードされます。

その後、WUTF フラグはソフトウェアでクリアする必要があります。

RTC_CR2 レジスタの WUTIE ビットをセットして周期的ウェイクアップ割り込みを有効にすると、デバイスは低電力モードを終了できます。

周期的なウェイクアップフラグは、RTC_CR レジスタの OSEL[0:1] ビットを通じて有効になっている場合に限り、RTC_ALARM 出力に送ることができます。RTC_ALARM 出力の極性は、RTC_CR レジスタの POL ビットを通じて設定できます。

低電力モード (SLEEP、STOP、STANDBY) と同様に、システムリセットもウェイクアップタイマには影響しません。

27.3.7 RTC の初期化と設定

RTC レジスタアクセス

RTC レジスタは、32 ビットのレジスタです。APB インターフェイスは、RTC レジスタアクセスに 2 ウェイトステートを挿入します。ただし、BYP SHAD = 0 のときのカレンダーシャドウレジスタへの読み出しアクセスは除きます。

RTC レジスタ書き込み保護

システムリセット後、RTC レジスタは、PWR_CR レジスタの DBP ビットをクリアすることによって、不要な書き込みアクセスから保護されます（電源制御のセクションを参照）。RTC レジスタ書き込みアクセスを可能にするには、DBP ビットをセットする必要があります。

RTC ドメインリセット後、すべての RTC レジスタは書き込み保護されます。RTC レジスタへの書き込みは、書き込み保護レジスタ RTC_WPR にキーを書き込むことにより有効になります。

RTC_TAMPCR、RTC_BKPxR、RTC_OR、および RTC_ISR[13:8] を除くすべての RTC レジスタの書き込み保護を解除するには、次のステップが必要です。

1. RTC_WPR レジスタに“0xCA”を書き込みます。
2. RTC_WPR レジスタに“0x53”を書き込みます。

誤ったキーを書き込むと、書き込み保護が再度アクティブになります。

保護メカニズムは、システムリセットの影響を受けません。

カレンダーの初期化と設定

時間形式やプリスケアラ設定を含むカレンダー時刻と日付の初期値をプログラムするには、次のシーケンスが必要です。

1. RTC_ISR レジスタで INIT ビットを 1 にセットして、初期化モードに入ります。このモードでは、カレンダーカウンタが停止し、その値を更新することができます。
2. RTC_ISR レジスタの INITF ビットをポーリングします。INITF が 1 にセットされると、初期化フェーズモードに入ります。これには RTCCLK クロック約 2 サイクルを必要とします（クロック同期のため）。
3. カレンダーカウンタのための 1 Hz クロックを生成するには、RTC_PRER レジスタで両方のプリスケアラ分周比をプログラムします。
4. シャドウレジスタ (RTC_TR および RTC_DR) に時刻と日付の初期値をロードし、RTC_CR レジスタの FMT ビットを介して時間形式 (12 時間または 24 時間) を設定します。
5. INIT ビットをクリアして初期化モードを終了します。その後、カレンダーカウンタの実際の値が自動的にロードされ、4 RTCCLK クロックサイクル後にカウントが再開します。

初期化シーケンスが完了すると、カレンダーがカウントを開始します。

注： システムリセット後、アプリケーションは RTC_ISR レジスタの INITS フラグを読み出し、カレンダーが初期化されたか否かを確認できるようになります。このフラグが 0 であれば、カレンダーの年の項目が RTC ドメインリセットデフォルト値 (0x00) にセットされているため、初期化されていません。初期化後にカレンダーを読み出すには、まずソフトウェアで RTC_ISR レジスタの RSF フラグがセットされていることを確認する必要があります。

サマータイム

サマータイム管理は、RTC_CR レジスタの SUB1H ビット、ADD1H ビット、BKP ビットを介して行われます。

SUB1H または ADD1H を使用すると、ソフトウェアは初期化手順を踏まずに 1 度の操作で、カレンダーから 1 時間引いたり足したりすることができます。

さらに、ソフトウェアは BKP ビットを使用してこの操作を記憶することができます。

アラームのプログラミング

プログラム可能なアラームをプログラムまたは更新するには、同様な手順を踏む必要があります。以下に示すのはアラーム A の手順ですが、アラーム B についても同様です。

1. RTC_CR の ALRAE をクリアしてアラーム A を無効にします。
2. アラーム A レジスタ (RTC_ALRMSSR/RTC_ALRMAR) をプログラムします。
3. RTC_CR レジスタで ALRAE をセットしてアラーム A を再び有効にします。

注： *RTC_CR レジスタの各変更は、クロック同期のため RTCCLK クロック約 2 サイクル後に有効になります。*

ウェイクアップタイマのプログラミング

ウェイクアップタイマ自動再ロード値 (RTC_WUTR の WUT[15:0]) の設定または変更には、次の手順が必要です。

1. RTC_CR の WUTE をクリアしてウェイクアップタイマを無効にします。
2. RTC_ISR の WUTWF がセットされ、ウェイクアップ自動再ロードカウンタおよび WUCKSEL[2:0] ビットへのアクセスが許可されていることが確認されるまで WUTWF をポーリングします。これには RTCCLK クロック約 2 サイクルを必要とします (クロック同期のため)。
3. ウェイクアップ自動再ロード値 WUT[15:0] およびウェイクアップクロック選択 (RTC_CR の WUCKSEL[2:0] ビット) をプログラムします。RTC_CR で WUTE をセットしてタイマを再び有効にします。ウェイクアップタイマがカウントダウンを再開します。

27.3.8 カレンダーの読み出し

RTC_CR レジスタの BYPSHAD 制御ビットがクリアされている場合

RTC カレンダーレジスタ (RTC_SSR、RTC_TR、および RTC_DR) を正しく読み出すには、APB1 クロック周波数 (f_{PCLK}) が RTC クロック周波数 (f_{RTCCLK}) の 7 倍以上でなければなりません。これにより、同期メカニズムの安全な動作が保証されます。

APB1 クロック周波数が RTC クロック周波数の 7 倍未満である場合、ソフトウェアによってカレンダー時間と日付のレジスタを 2 回読み出す必要があります。RTC_TR の 2 回目の読み出しが 1 回目の読み出しと同じ結果であれば、データが正しいことが保証されます。同じでない場合は、3 回目の読み出しアクセスを行う必要があります。どの場合も、APB1 クロック周波数は必ず RTC クロック周波数以上でなければなりません。

RTC_ISR レジスタの RSF ビットは、カレンダーレジスタが RTC_SSR、RTC_TR、および RTC_DR シャドウレジスタにコピーされるたびにセットされます。コピーは、2 RTCCLK サイクルごとに行われます。3 つの値における一貫性を保証するため、RTC_SSR または RTC_TR のどちらかを読み出すと、高次カレンダーシャドウレジスタの値は RTC_DR が読み出されるまでロックされます。ソフトウェアが 2 RTCCLK サイクル未満の間隔でカレンダーの読み出しアクセスを行う場合、最初のカレンダー読み出し後に RSF をソフトウェアでクリアする必要があり、その後ソフトウェアは、RSF ビットがセッ

トされるまで待つてから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

低電力モード (STOP または STANDBY) からのウェイクアップ後は、RSF をソフトウェアでクリアする必要があります。その後、ソフトウェアは、いま一度 RSF がセットされるまで待つてから、RTC_SSR、RTC_TR、および RTC_DR レジスタを再読み出す必要があります。

RSF ビットは、ウェイクアップ後にクリアする必要がありますが、低電力モードに入る前には、その必要はありません。

システムリセット後、ソフトウェアは RSF がセットされるまで待つてから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。実際、システムリセットがかかると、シャドウレジスタはデフォルト値にリセットされます。

初期化 ([カレンダーの初期化と設定 \(580 ページ\)](#)) を参照) 後、ソフトウェアは RSF がセットされるまで待つてから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

同期 ([セクション 27.3.10 : RTC の同期](#)) を参照) 後、ソフトウェアは RSF がセットされるまで待つてから、RTC_SSR、RTC_TR、および RTC_DR レジスタを読み出す必要があります。

RTC_CR レジスタ (バイパスシャドウレジスタ) の BYPSHAD 制御ビットがセットされている場合

カレンダーレジスタを読み出すと、カレンダーカウンタの値が直接与えられるため、RSF ビットがセットされるのを待つ必要がありません。シャドウレジスタは低電力モード (STOP または STANDBY) では更新されないため、低電力モード終了後に特にこのような読み出しが有用です。

BYPSHAD ビットが 1 にセットされている場合、レジスタへの 2 回の読み出しアクセス間で RTCCLK エッジが発生した場合は、さまざまなレジスタ間で互いに不整合が起きる場合があります。さらに、読み出し操作中に RTCCLK エッジが発生した場合、レジスタの 1 つが不正な値となる場合があります。ソフトウェアはすべてのレジスタを 2 回読み出し、その結果を比較してデータに整合性があり正しいことを確認する必要があります。その代わりに、ソフトウェアはカレンダーレジスタの最下位の数値を 2 回比較するだけで構いません。

注 : ***BYPSHAD = 1 の間、カレンダーレジスタの読み出し命令が完了するには 1 APB サイクルだけ余計に必要となります。***

27.3.9 RTC のリセット

カレンダーシャドウレジスタ (RTC_SSR、RTC_TR、RTC_DR) および RTC ステータスレジスタ (RTC_ISR) の一部のビットは、利用可能なすべてのシステムリセットリソースによってデフォルト値にリセットされます。

逆に、次のレジスタは RTC ドメインリセットによってそれぞれのデフォルト値にリセットされ、システムリセットの影響は受けません。RTC の現在のカレンダーレジスタ、RTC 制御レジスタ (RTC_CR)、プリスケアラレジスタ (RTC_PRER)、RTC 較正レジスタ (RTC_CALR)、RTC シフトレジスタ (RTC_SHIFTR)、RTC タイムスタンプレジスタ (RTC_TSSSR、RTC_TSTR および RTC_TSDR)、RTC タンパおよびオルタネート機能設定レジスタ (RTC_TAMPCR)、RTC バックアップレジスタ (RTC_BKPxR)、ウェイクアップタイマレジスタ (RTC_WUTR)、アラーム A とアラーム B レジスタ (RTC_ALRMASR/RTC_ALRMAR と RTC_ALRMBSSR/RTC_ALRMBR)、およびオプションレジスタ (RTC_OR)。

さらに、RTC はリセットソースが RTC ドメインリセットと異なる場合は、システムリセット中でも動作し続けます。RTC ドメインリセットが発生すると、RTC は停止し、すべての RTC レジスタがリセット値にセットされます。

27.3.10 RTC の同期

RTC は、高精度でリモートクロックと同期できます。サブセカンド項目 (RTC_SSR または RTC_TSSSR) を読み出すと、リモートクロックによって維持されている時刻と RTC 間の正確なオフセットが計算できます。その後、RTC_SHIFTR を使用してほんの一瞬クロックを「シフト」することによって RTC を調整し、このオフセットを取り除くことができます。

RTC_SSR には、同期プリスケアラのカウンタの値が入っています。これにより、RTC によって維持されている正確な時刻を $1/(\text{PREDIV_S} + 1)$ 秒の分解能まで計算することができます。その結果、同期プリスケアラ値 (PREDIV_S[14:0]) を増加させることにより分解能を改善できます。許可されている最大分解能 (32768 Hz クロックで 30.52 μ s) は、PREDIV_S を 0x7FFF にセットすることにより得られます。

ただし、PREDIV_S を増加させるということは、同期プリスケアラの出力を 1 Hz に維持するため PREDIV_A を減らす必要があることを意味します。このように、非同期プリスケアラの出力周波数が増加すると、RTC の動的消費電力が増加する場合があります。

RTC は、RTC シフト制御レジスタ (RTC_SHIFTR) を使って微調整できます。RTC_SHIFTR に書き込むことにより、 $1/(\text{PREDIV_S} + 1)$ 秒の分解能で、クロックを最大 1 秒だけシフト (遅れ/進み) させることができます。このシフト操作の本質は、同期プリスケアラのカウンタ SS[15:0] に SUBFS[14:0] 値を加算することであり、この操作はクロックを遅らせることになります。同時に ADD1S ビットがセットされた場合、1 秒追加すると同時に秒の小数部を差し引くことになるため、クロックを進めることになります。

注意： シフト操作を始める前に、ユーザーは SS[15] = 0 であることを確認し、オーバーフローが発生しないようにする必要があります。

RTC_SHIFTR レジスタへの書き込みによってシフト操作が始まるとすぐに、シフト操作が保留中であることを示す SHPF フラグがハードウェアによってセットされます。このビットは、シフト操作が完了するとすぐに、ハードウェアによってクリアされます。

注意： この同期機能はリファレンスクロック検出機能とは両立できません。具体的には、REFCKON = 1 のときにファームウェアから RTC_SHIFTR への書き込みはできません。

27.3.11 RTC リファレンスクロック検出

RTC カレンダの更新は、リファレンスクロックである RTC_REFIN に同期させることができます。通常は商用電源 (50 または 60 Hz) です。RTC_REFIN リファレンスクロックには、32.768 kHz LSE クロックより高い精度が必要です。RTC_REFIN 検出が有効になると (RTC_CR の REFCKON ビットが 1 にセット)、カレンダは引き続き LSE クロックによって駆動されますが、RTC_REFIN を使用してカレンダ更新周波数 (1 Hz) の不正確さを補正します。

各 1 Hz クロックエッジは、一番近い RTC_REFIN クロックエッジ (所与の時間枠内に見つかった場合) と比較されます。ほとんどの場合、2 つのクロックエッジは正しく整列しています。LSE クロックが不正確なために 1 Hz のクロックがずれた場合、RTC は 1 Hz のクロックを少しシフトさせ、その後の 1 Hz のクロックエッジが整列するようにします。このメカニズムのおかげで、カレンダはリファレンスクロックと同様に正確になります。

RTC は、32.768 kHz クォーツから生成される 256 Hz クロック (ck_apre) を使用して、リファレンスクロックソースがあるかどうかを検出します。検出は各カレンダ更新 (1 秒ごと) 程度の時間枠で行われます。最初のリファレンスクロックエッジを検出する際、この時間枠は ck_apre 7 周期に等しくなります。その後のカレンダ更新では、ck_apre 3 周期より短い時間枠が使用されます。

リファレンスクロックがこの時間枠内で検出されるたびに、ck_apre クロックを出力する非同期プリスケアラは強制的に再ロードされます。プリスケアラは同時に再ロードされるので、リファレンスクロックおよび 1 Hz のクロックが整列するタイミングには影響しません。クロックが整列していない

場合、後の 1 Hz クロックエッジは、リファレンスクロックと整列するように再ロードによって少しシフトされます。

リファレンスクロックが停止した (ck_apre 3 周期の枠内でリファレンスクロックエッジが発生しない) 場合、カレンダーは LSE クロックのみを基準にして更新が継続されます。その後 RTC は ck_spre エッジを中心として ck_apre 7 周期という広い検出時間枠でリファレンスクロックを待ちます。

リファレンスクロック検出を有効にした場合、PREDIV_A および PREDIV_S を以下に示すそれぞれのデフォルト値にセットする必要があります。

- PREDIV_A = 0x007F
- PREDIV_S = 0x00FF

注： *RTC_REFIN クロック検出は、STANDBY モードでは利用できません。*

27.3.12 RTC の高精度デジタル較正

RTC 周波数の精度は、-487.1~+488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル的に較正できます。周波数の修正は、一連の微調整 (個々の RTCCLK パルスの追加や削除) によって行われます。このような調整は、短い期間で観測された場合でも RTC が十分に較正されるように、かなり広範に分散して行われます。

この高精度デジタル較正は、入力周波数が 32768 Hz の場合、RTCCLK 約 2²⁰ パルスのサイクルまたは 32 秒の間で行われます。このサイクルは、RTCCLK によって駆動される 20 ビットカウンタ、cal_cnt[19:0] によって維持されます。

高精度較正レジスタ (RTC_CALR) によって、32 秒サイクル中にマスクされる RTCCLK クロックサイクル数を指定します。

- CALM[0] ビットを 1 にセットすると、32 秒サイクルの中でちょうど 1 パルスがマスクされます。
- CALM[1] ビットを 1 にセットすると、さらに 2 サイクルがマスクされます。
- CALM[2] ビットを 1 にセットすると、さらに 4 サイクルがマスクされます。
- CALM[8] ビットを 1 にセットするまで続けると、256 クロックがマスクされます。

注： *CALM[8:0] (RTC_CALR) によって、32 秒サイクル中にマスクされる RTCCLK パルス数を指定します。CALM[0] ビットを 1 にセットすると、cal_cnt[19:0] = 0x80000 になった時点で、32 秒サイクル中でちょうど 1 パルスがマスクされます。CALM[1] = 1 では、さらに 2 サイクルがマスクされ (cal_cnt = 0x40000 および 0xC0000)、CALM[2] = 1 では、さらに 4 サイクルがマスクされ (cal_cnt = 0x20000/0x60000/0xA0000/0xE0000)、CALM[8] = 1 まで続けると、256 クロックがマスクされず (cal_cnt = 0xFF800)。*

CALM では、細かい分解能で RTC 周波数を最大 487.1 ppm 負の方向に調整することができ、CALP ビットでは周波数を 488.5 ppm 正の方向に調整することができます。CALP を 1 にセットすることにより、実質上は、RTCCLK 2¹¹ サイクルごとに、RTCCLK パルスが 1 パルス追加で挿入されます。すなわち、32 秒サイクルごとに 512 クロックが追加されることとなります。

CALM を CALP と合わせて使用すると、32 秒サイクルの間に RTCCLK -511 から +512 サイクルまでのオフセットが追加でき、これは約 0.954 ppm の分解能で較正範囲 -487.1~+488.5 ppm に換算されます。

有効較正周波数 (FCAL) を入力周波数 (FRTCCLK) に対して求める計算式は次のとおりです。

$$F_{\text{CAL}} = F_{\text{RTCCLK}} \times [1 + (\text{CALP} \times 512 - \text{CALM}) / (2^{20} + \text{CALM} - \text{CALP} \times 512)]$$

PREDIV_A < 3 の場合の較正

非同期プリスケアラ値 (RTC_PRER レジスタの PREDIV_A ビット) が 3 未満の場合、CALP ビットを 1 にセットすることはできません。CALP がすでに 1 にセットされていて、PREDIV_A ビットが 3 未満の値にセットされた場合、CALP の設定値は無視され、CALP が 0 に設定された場合と同じように較正されます。

PREDIV_A が 3 未満の状態では較正を実施するには、各秒のカウントが 8 RTCCLK クロックサイクル分早められるように同期プリスケアラ値 (PREDIV_S) を小さくする必要があります。これは 32 秒毎に 256 クロックサイクル追加することに相当します。結果として、CALM ビットのみを使用して 32 秒サイクルの間に 255~256 クロックパルス (243.3~244.1 ppm の較正範囲に相当) を追加することができます。

公称 RTCCLK 周波数が 32768 Hz で、PREDIV_A が 1 (分周比 2) の場合、PREDIV_S を 16383 ではなく 16379 (4 少ない) にセットする必要があります。また、PREDIV_A が 0 の場合、PREDIV_S を 32767 ではなく 32759 (8 少ない) にセットする必要がありますので注意して下さい。

PREDIV_S をこのように減少させた場合、入力周波数 (FRTCCLK) に対して有効較正周波数 (FCAL) を計算する式は次のようになります。

$$F_{CAL} = F_{RTCCLK} \times [1 + (256 - CALM) / (2^{20} + CALM - 256)]$$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100 (CALM 設定範囲の中間値) と等しくなるのが正しい設定です。

RTC 較正值の確認

RTC の精度は、RTCCLK の正確な周波数を測定し、正しい CALM 値および CALP 値を計算することにより保証されます。オプションの 1 Hz 出力が搭載されており、アプリケーションによって RTC 精度の測定と確認を行うことができます。

ある時間間隔で RTC の周波数を精密に測定すると、デジタル較正サイクルを測定周期とどのように合わせているかにより、測定期間中に最大 2 RTCCLK クロックサイクルの測定誤差が生じます。

ただし、この測定誤差は、測定周期が較正サイクル周期と同じ長さであれば排除できます。この場合、観測される唯一の誤差はデジタル較正の分解能による誤差となります。

- デフォルトでは、較正サイクル周期は 32 秒です。

このモードを使用して正確に 32 秒で 1 Hz 出力の精度を測定すると、その精度は 0.477 ppm (較正分解能の制限により 32 秒で 0.5 RTCCLK サイクル) 以内となることが保証されます。

- RTC_CALR レジスタの CALW16 ビットを 1 にセットして、較正サイクル周期を強制的に 16 秒にすることができます。

この場合、RTC 精度は最大誤差 0.954 ppm (16 秒で 0.5 RTCCLK サイクル) で 16 秒間で測定できます。ただし、較正分解能が下がるため、長期的な RTC 精度もまた 0.954 ppm に下がります。CALW16 が 1 にセットされると、CALM[0] ビットは 0 のままとなります。

- RTC_CALR レジスタの CALW8 ビットを 1 にセットして、較正サイクル周期を強制的に 8 秒にすることができます。

この場合、RTC 精度は最大誤差 1.907 ppm (8 秒で 0.5 RTCCLK サイクル) で 8 秒で測定できます。長期的な RTC 精度もまた 1.907 ppm に下がります。CALW8 が 1 にセットされると、CALM[1:0] ビットは 00 のままとなります。

動作中の再較正

次の処理を実施することにより、RTC_ISR/INITF = 0 の間でも、較正レジスタ (RTC_CALR) を動作中に更新することができます。

1. RTC_ISR/RECALPF (再較正保留フラグ) をポーリングします。
2. このフラグが 0 にセットされている場合は、必要に応じて新しい値を RTC_CALR に書き込みます。すると、RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書き込み動作後 ck_apre 3 サイクル以内に、新しい較正設定が有効になります。

27.3.13 タイムスタンプ機能

タイムスタンプは、RTC_CR レジスタの TSE ビットを 1 にセットすることにより有効になります。

RTC_TS ピンでタイムスタンプイベントが検出されると、タイムスタンプレジスタ (RTC_TSSSR、RTC_TSTR、RTC_TSDR) にカレンダーが保存されます。

また、タイムスタンプイベントが発生すると、RTC_ISR レジスタのタイムスタンプフラグビット (TSF) がセットされます。

RTC_CR レジスタの TSIE ビットをセットすることにより、タイムスタンプイベントが発生したときに割り込みが生成されます。

タイムスタンプフラグ (TSF) がすでにセットされている間に新しいタイムスタンプイベントが検出された場合、タイムスタンプオーバーフローフラグ (TSOVF) がセットされ、タイムスタンプレジスタ (RTC_TSTR および RTC_TSDR) は、その前のイベントの結果を維持します。

注： *同期処理のため、TSF はタイムスタンプイベント発生から ck_apre 2 サイクル後にセットされます。一方、TSOVF のセットに遅延はありません。これは、2 つのタイムスタンプイベントの発生したタイミングが近い場合、TSF がまだ“0”であっても TSOVF が“1”と検出される可能性があることを意味します。よって、TSOVF のポーリングは TSF がセットされた後に実施することをお勧めします。*

注意： TSF ビットがクリアされたと思われた直後にタイムスタンプイベントが発生した場合、TSF および TSOVF ビットの両方がセットされます。同時に発生するタイムスタンプイベントのマスキングを回避するため、TSF がすでに“1”と読み出されていない限りは、アプリケーションによる“0”クリアの処理を行ってはいけません。

オプション機能として、タンパイベントによってタイムスタンプイベントを記録することもできます。TAMPTS 制御ビットの詳細については、[セクション 27.6.14 : RTC タイムスタンプサブセカンドレジスタ \(RTC_TSSSR\)](#) を参照してください。

27.3.14 タンパ検出

RTC_TAMPx 入力イベントは、フィルタを使用してエッジ検出またはレベル検出のどちらにも設定できます。

タンパ検出は次の目的で設定することができます。

- RTC バックアップレジスタ (デフォルト設定) を消去する。
- STOP モードおよび STANDBY モードからのウェイクアップが可能な割り込みを生成する。
- 低電力タイマのためのハードウェアトリガを生成する。

RTC バックアップレジスタ

バックアップレジスタ (RTC_BKPxR) は、システムリセットや STANDBY モードからのウェイクアップではリセットされません。

バックアップレジスタ (RTC_BKPxR) は、タンパ検出イベント発生時にリセットされます ([セクション 27.6.20 : RTC バックアップレジスタ \(RTC_BKPxR\)](#) および [タンパ検出の初期化 \(587 ページ\)](#)) を参照。またはフラッシュの読み出し保護がレベル 1 からレベル 0 に変わったとき) が、TAMPxNOERASE ビットがセットされている場合や RTC_TAMPCR レジスタの TAMPxMF がセットされている場合は例外です。

タンパ検出の初期化

各入力は、RTC_TAMPCR レジスタにおいて該当する TAMPxE ビットを 1 にセットすることにより有効にできます。

各 RTC_TAMPx タンパ検出入力は、RTC_ISR レジスタの TAMPxF フラグに関連付けられています。

TAMPxMF がクリアされている場合

TAMPxF フラグは、ピン上でタンパイベントが発生した後にアサートされます。その際の遅延時間を以下に示します。

- TAMPFLT = 0x0 以外の場合、ck_apre 3 サイクル (フィルタを使ったレベル検出)
- TAMPTS = 1 の場合、ck_apre 3 サイクル (タンパイベント時のタイムスタンプ)
- TAMPFLT = 0x0 (エッジ検出) および TAMPTS = 0 の場合、遅延なし

TAMPxF がセットされている場合に、この周期中に同一ピンで発生した新たなタンパイベントを検出することはできません。

TAMPxMF がセットされている場合

同一ピンで発生した新たなタンパイベントは、上述した遅延の間とさらに ck_rtc 2.5 サイクルの間は検出できません。

RTC_TAMPCR レジスタの TAMPxIE ビットをセットすることにより、タンパ検出イベント発生時に割り込みが生成されます (TAMPxF がセットされている場合)。1 つ以上の S TAMPxMF がセットされている場合、TAMPxIE をセットすることはできません。

TAMPxIE がクリアされている場合は、各タンパピンイベント／割り込みは、RTC_TAMPCR レジスタの対応する TAMPxIE ビットをセットすることによって個別に有効にすることができます。対応する TAMPxMF がセットされている場合、TAMPxIE をセットすることはできません。

タンパイベント時のトリガ出力の生成

タンパイベント検出は、低電力タイマによるトリガ入力として使用できます。

RTC_TAMPCR レジスタの TAMPxMF ビットがクリアされている場合、TAMPxF フラグをソフトウェアでクリアして、同一ピンで新たなタンパイベントを検出できるようにする必要があります。

TAMPxMF ビットがセットされている場合、TAMPxF フラグはマスクされ、RTC_ISR レジスタでクリアされたままとなります。この設定により、TAMPxF をクリアするためにシステムをウェイクアップする必要もなく、STOP モードのまま低電力タイマを自動的にトリガすることができます。この場合、バックアップレジスタはクリアされません。

タンバイイベント時のタイムスタンプ

TAMPTS を“1”にセットすると、すべてのタンバイイベントがタイムスタンプイベントを発生させるようになります。この場合、通常のタイムスタンプイベント発生時と同様に TSF ビットまたは TSOVF ビットが RTC_ISR でセットされます。TSF または TSOVF がセットされるのと同時に、影響を受けるタンパフラグレジスタ、TAMPxF がセットされます。

タンパ入力でのエッジ検出

TAMPFLT ビットが“00”の場合、該当する TAMPxTRG ビットに応じて立ち上がりエッジまたは立ち下がりエッジが観測されると、RTC_TAMPx ピンがタンパ検出イベントを生成します。エッジ検出を選択すると、RTC_TAMPx 入力の内部プルアップ抵抗が無効になります。

注意： タンパ検出イベントを確実に検出するため、RTC_TAMPx ピンが有効になる前にタンパ検出イベントが発生した場合でも検出できるよう、エッジ検出に使用される信号は対応する TAMPxE ビットと論理積がとられます。

- TAMPxTRG = 0 の場合：タンパ検出が有効になる（TAMPxE ビットが 1 にセットされる）前に RTC_TAMPx オルタネート機能がすでにハイのとき、TAMPxE がセットされた後に RTC_TAMPx に立ち上がりエッジ入力がなくとも、RTC_TAMPx 入力が有効になるとすぐにタンバイイベントが検出されます。
- TAMPxTRG = 1 の場合：タンパ検出が有効になる前に RTC_TAMPx オルタネート機能がすでにローのとき、RTC_TAMPx 入力が有効になるとすぐに（TAMPxE がセットされた後に RTC_TAMPx に立ち下がりエッジ入力がなくとも）、タンバイイベントが検出されます。

タンバイイベントが検出されクリアされた後に、バックアップレジスタ（RTC_BKPxR）を再プログラムする場合には、事前に、RTC_TAMPx オルタネート機能を無効にしてから再度有効にする（TAMPxE を 1 にセット）必要があります。これによって、RTC_TAMPx 入力の値がタンパ検出を示している間に、アプリケーションがバックアップレジスタにデータを書き込むのを防ぎます。これは、RTC_TAMPx オルタネート機能の入力でのレベル検出に相当します。

RTC_TAMPx 入力でのフィルタを使ったレベル検出

フィルタを使ったレベル検出は、TAMPFLT を 0 以外の値にセットすることにより行われます。タンパ検出イベントは、（TAMPFLT に応じて）2、4 または 8 回のいずれかの連続したサンプルが TAMPxTRG ビットで指定するレベルで観測されたときに生成されます。

RTC_TAMPx 入力は、TAMPPUDIS が 1 にセットされて無効な状態になっていない限り、その状態がサンプリングされる前に I/O の内部プルアップ抵抗でプリチャージされています。プリチャージの継続時間は TAMPPRCH ビットによって決定され、RTC_TAMPx 入力ピンにおけるより大きな容量を持たせることができます。

タンパ検出の遅延時間と、プルアップによる電力消費との間のトレードオフは、TAMPFREQ を使用してレベル検出のサンプリング周波数を決定することにより、最適化できます。

注： プルアップ抵抗の電気的特性については、データシートを参照してください。

27.3.15 較正クロック出力

RTC_CR レジスタで COE ビットが 1 にセットされると、RTC_CALIB デバイス出力にリファレンスクロックが供給されます。

RTC_CR レジスタの COSEL ビットがリセットされ、かつ PREDIV_A = 0x7F である場合、RTC_CALIB 周波数は $f_{\text{RTCCLK}}/64$ です。これは 32.768 kHz の RTCCLK 周波数に対する 512 Hz の較正出力に相当します。立ち下がリエッジには軽いジッタがあるため、RTC_CALIB のデューティサイクルは不規則になります。したがって、立ち上がりエッジの使用が推奨されます。

COSEL がセットされ、かつ “PREDIV_S+1” がゼロ以外の 256 の倍数である場合（すなわち、PREDIV_S[7:0] = 0xFF）、RTC_CALIB 周波数は $f_{\text{RTCCLK}}/(256 * (\text{PREDIV_A}+1))$ となります。これは、RTCCLK 周波数が 32.768 kHz で、プリスケアラデフォルト値（PREDIV_A = 0x7F、PREDIV_S = 0xFF）に対する 1 Hz の較正出力に相当します。

注： *RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力オルタネート機能に設定されます。*

27.3.16 アラーム出力

RTC_CR レジスタの OSEL[1:0] 制御ビットを使用してアラームオルタネート機能出力、RTC_ALARM を有効にし、出力となる機能を選択します。これらの機能は、RTC_ISR レジスタの該当するフラグの内容を反映します。

RTC_CR の POL 制御ビットは、POL が 1 にセットされているときに選択されたフラグビットの逆が出力されるよう、出力の極性を決定します。

アラームオルタネート機能出力

RTC_ALARM ピンは、制御ビット RTC_ALARM_TYPE（RTC_OR レジスタ）を使用して、出力オープンドレインまたは出力プッシュプルに設定できます。

注： *RTC_ALARM 出力が有効になると、この設定は、RTC_CALIB（COE ビットは無視され、クリアされたまま）の設定よりも優先されます。*

RTC_CALIB または RTC_ALARM 出力が選択されると、RTC_OUT ピンは自動的に出力オルタネート機能に設定されます。

27.4 RTC 低電力モード

表 97. 低電力モードが RTC に与える影響

| モード | 説明 |
|---------|---|
| SLEEP | 影響なし。 RTC 割り込みによって、デバイスは SLEEP モードから復帰します。 |
| STOP | RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC アラーム、RTC タンパイイベント、RTC タイムスタンプイベント、RTC ウェイクアップにより、デバイスは STOP モードから復帰します。 |
| STANDBY | RTC クロックソースが LSE または LSI の場合、RTC はアクティブのままです。RTC アラーム、RTC タンパイイベント、RTC タイムスタンプイベント、RTC ウェイクアップにより、デバイスは STANDBY モードから復帰します。 |

27.5 RTC 割り込み

すべての RTC 割り込みは、EXTI コントローラに接続されています。[セクション 13.5 : EXTI レジスタを参照してください。](#)

RTC 割り込みを有効にするには、次のシーケンスが必要です。

1. 割り込みモードで RTC イベントに対応する EXTI ラインを設定して有効にし、立ち上がりエッジ感度を選択します。
2. NVIC で RTC IRQ チャンネルを設定し、有効にします。
3. RTC を、RTC 割り込みを生成するように設定します。

表 98. 割り込み制御ビット

| 割り込みイベント | イベントフラグ | イネーブル制御ビット | SLEEP モードの 終了 | STOP モードの 終了 | STANDBY モードの 終了 |
|---------------------|---------|------------|---------------------|--------------------|-----------------------|
| アラーム A | ALRAF | ALRAIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |
| アラーム B | ALRBF | ALRBIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |
| RTC_TS 入力 (タイムスタンプ) | TSF | TSIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |
| RTC_TAMP1 入力検出 | TAMP1F | TAMPIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |
| RTC_TAMP2 入力検出 | TAMP2F | TAMPIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |
| ウェイクアップタイマ割り込み | WUTF | WUTIE | あり | あり ⁽¹⁾ | あり ⁽¹⁾ |

1. STOP モードおよび STANDBY モードからのウェイクアップは、RTC クロックソースが LSE または LSI のときのみ可能です。

27.6 RTC レジスタ

レジスタの説明で使用されている略語のリストについては、リファレンスマニュアルの [セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ワード (32 ビット) 単位でアクセスすることができます。

27.6.1 RTC 時刻レジスタ (RTC_TR)

RTC_TR は、カレンダー時刻シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。[カレンダーの初期化と設定 \(580 ページ\)](#) および [カレンダーの読み出し \(581 ページ\)](#) を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x00

RTC ドメインリセット値 : 0x0000 0000

システムリセット : BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

| | | | | | | | | | | | | | | | |
|------|----------|------|------|----------|------|------|------|------|---------|---------|----|---------|---------|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PM | HT[1:0] | | | HU[3:0] | | |
| | | | | | | | | | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | MNT[2:0] | | | MNU[3:0] | | | | Res. | ST[2:0] | | | SU[3:0] | | | |
| | rW | rW | rW | rW | rW | rW | rW | | rW | rW | rW | rW | rW | rW | rW |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM** : AM/PM 表記

0 : AM または 24 時間形式

1 : PM

ビット 21:20 **HT[1:0]** : BCD 形式での時の十の位

ビット 19:16 **HU[3:0]** : BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]** : BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]** : BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]** : BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]** : BCD 形式での秒の一の位

27.6.2 RTC 日付レジスタ (RTC_DR)

RTC_DR は、カレンダー日付シャドウレジスタです。このレジスタは、必ず初期化モードで書き込む必要があります。[カレンダーの初期化と設定 \(580 ページ\)](#) および [カレンダーの読み出し \(581 ページ\)](#) を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x04

RTC ドメインリセット値 : 0x0000 2101

システムリセット : BYPSHAD = 0 の場合、0x0000 2101 です。BYPSHAD = 1 の場合、影響を受けません。

| | | | | | | | | | | | | | | | |
|----------|------|------|------|------|---------|------|------|---------|------|------|---------|---------|---------|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | YT[3:0] | | | | YU[3:0] | | | |
| | | | | | | | | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| WDU[2:0] | | | | MT | MU[3:0] | | | | Res. | Res. | DT[1:0] | | DU[3:0] | | |
| rW | rW | rW | rW | rW | rW | rW | rW | | | | rW | rW | rW | rW | rW |

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23:20 **YT[3:0]** : BCD 形式での年の十の位

ビット 19:16 **YU[3:0]** : BCD 形式での年の一の位

ビット 15:13 **WDU[2:0]** : 曜日

000 : 禁止
001 : 月曜日
~
111 : 日曜日

ビット 12 **MT** : BCD 形式での月の十の位

ビット 11:8 **MU** : BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]** : BCD 形式での日の十の位

ビット 3:0 **DU[3:0]** : BCD 形式での日の一の位

27.6.3 RTC 制御レジスタ (RTC_CR)

アドレスオフセット : 0x08

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

| | | | | | | | | | | | | | | | |
|------|-------|--------|--------|------|------|-------|-------|------|-----------|---------|---------|--------|--------------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | COE | OSEL[1:0] | | POL | COSEL | BKP | SUB1H | ADD1H |
| | | | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | w | w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TSIE | WUTIE | ALRBIE | ALRAIE | TSE | WUTE | ALRBE | ALRAE | Res. | FMT | BYPSHAD | REFCKON | TSEDGE | WUCKSEL[2:0] | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31:24 予約済みであり、リセット値に保持する必要があります。

ビット 23 **COE** : 較正出力イネーブル

このビットは、RTC_CALIB 出力を有効にします。

0 : 較正出力は無効です。

1 : 較正出力は有効です。

ビット 22:21 **OSEL[1:0]** : 出力選択

これらのビットは、RTC_ALARM 出力に送られるフラグの選択に使用します。

00 : 出力は無効です。

01 : アラーム A 出力は有効です。

10 : アラーム B 出力は有効です。

11 : ウェイクアップ出力は有効です。

ビット 20 **POL** : 出力極性

このビットは、RTC_ALARM 出力の極性の設定に使用します。

0 : ALRAF/ALRBF/WUTF がアサートされると、このピンがハイになります (OSEL[1:0] に応じて)。

1 : ALRAF/ALRBF/WUTF がアサートされると、このピンがローになります (OSEL[1:0] に応じて)。

ビット 19 **COSEL** : 較正出力選択

COE = 1 のとき、このビットによって RTC_CALIB に出力される信号を選択します。

0 : 較正出力は 512 Hz です。

1 : 較正出力は 1 Hz です。

これらの周波数は、RTCCLK が 32.768 kHz で、プリスケアラがデフォルト値 (PREDIV_A = 127 および PREDIV_S = 255) の場合に有効です。 [セクション 27.3.15 : 較正クロック出力](#) を参照してください。

ビット 18 **BKP** : バックアップ

このビットは、サマータイムの変更を実施したか否かを記憶しておくため、ユーザーが書き込むことができます。

ビット 17 **SUB1H** : 1 時間差し引き (冬時間変更)

このビットを初期化モード以外のときにセットすると、現在時刻が 0 でない場合にカレンダー時刻から 1 時間を差し引きします。このビットは常に 0 として読み出されます。

現在時間が 0 のときにこのビットをセットしても、影響はありません。

0 : 影響なし。

1 : 現在時刻から 1 時間差し引きします。これは、冬時間変更に使えます。

ビット 16 ADD1H : 1 時間加算 (サマータイム変更)

このビットを初期化モード以外のときにセットすると、カレンダー時刻に 1 時間加算します。このビットは常に 0 として読み出されます。

0 : 影響なし。

1 : 現在時刻に 1 時間加算します。これは、サマータイム変更に使えます。

ビット 15 TSIE : タイムスタンプ割り込みイネーブル

0 : タイムスタンプ割り込みは無効です。

1 : タイムスタンプ割り込みは有効です。

ビット 14 WUTIE : ウェイクアップタイマ割り込みイネーブル

0 : ウェイクアップタイマ割り込みは無効です。

1 : ウェイクアップタイマ割り込みは有効です。

ビット 13 ALRBIE : アラーム B 割り込みイネーブル

0 : アラーム B 割り込みは無効です。

1 : アラーム B 割り込みは有効です。

ビット 12 ALRAIE : アラーム A 割り込みイネーブル

0 : アラーム A 割り込みは無効です。

1 : アラーム A 割り込みは有効です。

ビット 11 TSE : タイムスタンプイネーブル

0 : タイムスタンプは無効です。

1 : タイムスタンプは有効です。

ビット 10 WUTE : ウェイクアップタイマイネーブル

0 : ウェイクアップタイマは無効です。

1 : ウェイクアップタイマは有効です。

ビット 9 ALRBE : アラーム B イネーブル

0 : アラーム B は無効です。

1 : アラーム B は有効です。

ビット 8 ALRAE : アラーム A イネーブル

0 : アラーム A は無効です。

1 : アラーム A は有効です。

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6 FMT : 時間形式

0 : 24 時間 / 日形式

1 : AM / PM 時間形式

ビット 5 BYPSHAD : シャドウレジスタをバイパスします。

0 : カレンダー値は (RTC_SSR、RTC_TR、RTC_DR から読み出す場合)、シャドウレジスタから取得され、これらは 2 RTCCLK サイクルごとに 1 回更新されます。

1 : カレンダー値は (RTC_SSR、RTC_TR、RTC_DR から読み出す場合)、カレンダーカウンタから直接取得されます。

注 : *APB1 クロックの周波数が RTCCLK の 7 倍未満である場合、BYPHAD は“1”にセットする必要があります。*

ビット 4 **REFCKON** : RTC_REFIN リファレンスクロック検出イネーブル (50 または 60 Hz)

0 : RTC_REFIN 検出は無効です。

1 : RTC_REFIN 検出は有効です。

注 : *PREDIV_S は 0x00FF である必要があります。*

ビット 3 **TSEDGE** : タイムスタンプイベントアクティブエッジ

0 : RTC_TS 入力の立ち上がりエッジによってタイムスタンプイベントを生成します。

1 : RTC_TS 入力の立ち下がりエッジによってタイムスタンプイベントを生成します。

不要な TSF 設定を回避するため、TSEDGE が変化した場合には TSE をリセットする必要があります。

ビット 2:0 **WUCKSEL[2:0]** : ウェイクアップクロック選択

000 : RTC/16 クロックが選択されます。

001 : RTC/8 クロックが選択されます。

010 : RTC/4 クロックが選択されます。

011 : RTC/2 クロックが選択されます。

10x : ck_spre (通常は 1 Hz) クロックが選択されます。

11x : ck_spre (通常は 1 Hz) クロックが選択され、 2^{16} が WUT カウンタ値に加算されます (下記注を参照)。

注 : *初期化モード (RTC_ISR/INITF = 1) の場合のみ、このレジスタのビット 7、6、4 が書き込めます。
WUT = ウェイクアップユニットカウンタ値 $WUT = (0x0000 \sim 0xFFFF) + 0x10000$ (WUCKSEL[2:1] = 11 の場合追加されます。)*

このレジスタのビット 2~0 は、RTC_CR WUTE ビット = 0 かつ RTC_ISR WUTWF ビット = 1 の場合にのみ書き込めます。

カレンダーの時間項目のインクリメント中は時間を変更しないことが推奨されます。カレンダーの時間項目のインクリメントがマスクされる可能性があるためです。

ADD1H および SUB1H の変更は、次の秒から有効になります。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

27.6.4 RTC 初期化とステータスレジスタ (RTC_ISR)

このレジスタは、書き込み保護されています (RTC_ISR[13:8] ビットを除く)。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x0C

RTC ドメインリセット値 : 0x0000 0007

システムリセット : 0 にクリアされる INIT、INITF、RSF ビット以外は影響されません。

| | | | | | | | | | | | | | | | |
|------|--------|--------|-------|-------|-------|-------|-------|------|-------|-------|-------|------|-------|------------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RECALPF |
| | | | | | | | | | | | | | | | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | TAMP2F | TAMP1F | TSOVF | TSF | WUTF | ALRBF | ALRAF | INIT | INITF | RSF | INITS | SHPF | WUTWF | ALRB WF | ALRAWF |
| | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rw | r | rc_w0 | r | r | r | r | r |

ビット 31:17 予約済みであり、リセット値に保持する必要があります。

ビット 16 **RECALPF** : 再較正保留フラグ

ソフトウェアによって RTC_CALR レジスタに書き込みが行われると、RECALPF ステータスフラグが自動的に“1”にセットされ、RTC_CALR レジスタがブロックされたことを示します。新たな較正設定が認識されると、このビットは“0”に戻ります。[動作中の再較正](#)を参照してください。

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14 **TAMP2F** : RTC_TAMP2 検出フラグ

このフラグは、RTC_TAMP2 入力にタンバ検出イベントが検出されたときに、ハードウェアによってセットされます。
ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 13 **TAMP1F** : RTC_TAMP1 検出フラグ

このフラグは、RTC_TAMP1 入力にタンバ検出イベントが検出されたときに、ハードウェアによってセットされます。
ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 12 **TSOVF** : タイムスタンプオーバーフローフラグ

このフラグは、TSF が既にセットされている間にタイムスタンプイベントが発生したときに、ハードウェアによってセットされます。
このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。TSOVF は、TSF ビットをクリアした後、チェックしてからクリアすることが推奨されます。そうしないと、TSF ビットがクリアされる直前にタイムスタンプイベントが発生した場合、オーバーフローを見逃す可能性があります。

ビット 11 **TSF** : タイムスタンプフラグ

このフラグは、タイムスタンプイベントが発生したときに、ハードウェアによってセットされます。
このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 10 **WUTF** : ウェイクアップタイマフラグ

このフラグは、ウェイクアップ自動再ロードカウンタが 0 に到達したときに、ハードウェアによってセットされます。
このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。
このフラグは、WUTF が再び 1 にセットされる前、RTCCLK 1.5 周期以上前にソフトウェアでクリアする必要があります。

ビット 9 ALRBF : アラーム B フラグ

このフラグは、時刻/日付レジスタ (RTC_TR および RTC_DR) がアラーム B レジスタ (RTC_ALRMBR) と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 8 ALRAF : アラーム A フラグ

このフラグは、時刻/日付レジスタ (RTC_TR および RTC_DR) がアラーム A レジスタ (RTC_ALRMAR) と一致したときにハードウェアによってセットされます。

このフラグは、ソフトウェアで 0 を書き込むことによってクリアされます。

ビット 7 INIT : 初期化モード

0 : フリーランニングモード

1 : 時刻と日付レジスタ (RTC_TR と RTC_DR)、およびプリスケアラレジスタ (RTC_PRER) のプログラムに使用する初期化モードです。INIT がリセットされると、カウンタは停止し、新しい値からカウントし始めます。

ビット 6 INITF : 初期化フラグ

このビットが 1 にセットされると、RTC は初期化状態となり、時刻、日付およびプリスケアラレジスタが更新できます。

0 : カレンダレジスタを更新できません。

1 : カレンダレジスタを更新できます。

ビット 5 RSF : レジスタ同期フラグ

このビットは、カレンダレジスタがシャドウレジスタ (RTC_SSRx、RTC_TRx および RTC_DRx) にコピーされるたびにハードウェアによってセットされます。このビットは、シフト操作が保留中 (SHPF = 1) に初期化モードで、またはバイパスシャドウレジスタモード (BYPSSHAD = 1) で、ハードウェアによってクリアされます。このビットは、ソフトウェアでクリアすることもできます。

初期化モードでソフトウェアまたはハードウェアによってクリアされます。

0 : カレンダシャドウレジスタはまだ同期していません。

1 : カレンダシャドウレジスタは同期しています。

ビット 4 INITS : 初期化ステータスフラグ

このビットは、カレンダの年の項目が 0 ではないとき (RTC ドメインリセット状態) にハードウェアによってセットされます。

0 : カレンダは初期化されていません。

1 : カレンダは初期化されています。

ビット 3 SHPF : シフト操作保留

0 : 保留中のシフト操作はありません。

1 : 保留中のシフト操作があります。

このフラグは、RTC_SHIFTR への書き込みによってシフト操作が開始された直後に、ハードウェアによってセットされます。該当するシフト操作が実行されると、ハードウェアによってクリアされます。SHPF ビットに書き込んでも影響はありません。

ビット 2 WUTWF : ウェイクアップタイマ書き込みフラグ

このビットは、RTC_CR で WUTE ビットが 0 にセットされた後、ウェイクアップタイマ値が変更可能な時にハードウェアによってセットされます。

0 : ウェイクアップタイマ設定は更新できません。

1 : ウェイクアップタイマ設定は更新できます。

ビット 1 ALRBWF : アラーム B 書き込みフラグ

このビットは、RTC_CR で ALRBIE ビットが 0 にセットされた後、アラーム B 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0 : アラーム B は更新できません。

1 : アラーム B は更新できます。

ビット 0 ALRAWF : アラーム A 書き込みフラグ

このビットは、RTC_CR で ALRAE ビットが 0 にセットされた後、アラーム A 値が変更可能な時にハードウェアによってセットされます。

初期化モードでハードウェアによってクリアされます。

0 : アラーム A は更新できません。

1 : アラーム A は更新できます。

注 : **ALRAF、ALRBF、WUTF、および TSF ビットは、0 にプログラムされてから 2 APB クロック サイクル後にクリアされます。**

27.6.5 RTC プリスケアラレジスタ (RTC_PRER)

このレジスタは、必ず初期化モードで書き込む必要があります。初期化は、2 回の書き込みアクセスに分けて行う必要があります。[カレンダーの初期化と設定 \(580 ページ\)](#) を参照してください。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x10

RTC ドメインリセット値 : 0x007F 00FF

システムリセット : 影響なし

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

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22:16 **PREDIV_A[6:0]** : 非同期プリスケアラ分周比
非同期分周比です。
 $ck_apre \text{ 周波数} = RTCCLK \text{ 周波数} / (PREDIV_A + 1)$

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **PREDIV_S[14:0]** : 同期プリスケアラ分周比
同期分周比です。
 $ck_spre \text{ 周波数} = ck_apre \text{ 周波数} / (PREDIV_S + 1)$

27.6.6 RTC ウェイクアップタイマレジスタ（RTC_WUTR）

このレジスタは、RTC_ISR の WUTWF が 1 にセットされているときにのみ書き込みます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット：0x14

RTC ドメインリセット値：0x0000 FFFF

システムリセット：影響なし

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **WUT[15:0]**：ウェイクアップ自動再ロード値ビット

ウェイクアップタイマが有効（WUTE が 1 にセット）なとき、ck_wut の（WUT[15:0] + 1）サイクル毎に WUTF フラグがセットされます。ck_wut の周期は、RTC_CR レジスタの WUCKSEL[2:0] ビットで選択します。

WUCKSEL[2] = 1 のとき、ウェイクアップタイマは 17 ビットとなり、WUCKSEL[1] が事実上タイマに再ロードされる最上位ビットである WUT[16] となります。

WUTF の最初のアサートは、WUTE がセットされてからck_wut の（WUT+1）サイクル後に発生します。WUCKSEL[2:0] = 011（RTCCLK/2）のときに WUT[15:0] を 0x0000 にセットすることはできません。



27.6.7 RTC アラーム A レジスタ (RTC_ALRMAR)

このレジスタは、RTC_ISR の ALRAWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込みめます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x1C

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

| | | | | | | | | | | | | | | | |
|------|----------|---------|-----|----------|-----|-----|-----|------|---------|---------|-----|---------|-----|-----|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| MSK4 | WDSEL | DT[1:0] | | DU[3:0] | | | | MSK3 | PM | HT[1:0] | | HU[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MSK2 | MNT[2:0] | | | MNU[3:0] | | | | MSK1 | ST[2:0] | | | SU[3:0] | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 31 **MSK4** : アラーム A 日付マスク
0 : 日付／曜日が一致すると、アラーム A がセットされます。
1 : アラーム A の比較では日付／曜日を無視します。

ビット 30 **WDSEL** : 曜日選択
0 : DU[3:0] は日付の一の位を表します。
1 : DU[3:0] は曜日を表します。DT[1:0] は無視されます。

ビット 29:28 **DT[1:0]** : BCD 形式での日の十の位

ビット 27:24 **DU[3:0]** : BCD 形式での日の一の位または曜日

ビット 23 **MSK3** : アラーム A 時マスク
0 : 時が一致すると、アラーム A がセットされます。
1 : アラーム A の比較では時を無視します。

ビット 22 **PM** : AM/PM 表記
0 : AM または 24 時間形式
1 : PM

ビット 21:20 **HT[1:0]** : BCD 形式での時の十の位

ビット 19:16 **HU[3:0]** : BCD 形式での時の一の位

ビット 15 **MSK2** : アラーム A 分マスク
0 : 分が一致すると、アラーム A がセットされます。
1 : アラーム A の比較では分を無視します。

ビット 14:12 **MNT[2:0]** : BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]** : BCD 形式での分の一の位

ビット 7 **MSK1** : アラーム A 秒マスク
0 : 秒が一致すると、アラーム A がセットされます。
1 : アラーム A の比較では秒を無視します。

ビット 6:4 **ST[2:0]** : BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]** : BCD 形式での秒の一の位



27.6.8 RTC アラーム B レジスタ（RTC_ALRMBR）

このレジスタは、RTC_ISR の ALRBWF が 1 にセットされた場合、または初期化モードの場合にのみ書き込みめます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット：0x20

RTC ドメインリセット値：0x0000 0000

システムリセット：影響なし

| | | | | | | | | | | | | | | | |
|------|----------|---------|----|----------|----|----|----|------|---------|---------|----|---------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| MSK4 | WDSEL | DT[1:0] | | DU[3:0] | | | | MSK3 | PM | HT[1:0] | | HU[3:0] | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| MSK2 | MNT[2:0] | | | MNU[3:0] | | | | MSK1 | ST[2:0] | | | SU[3:0] | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

- ビット 31 **MSK4**：アラーム B 日付マスク
- 0：日付／曜日が一致すると、アラーム B がセットされます。
 - 1：アラーム B の比較では日付／曜日を無視します。

- ビット 30 **WDSEL**：曜日選択
- 0：DU[3:0] は日付の一の位を表します。
 - 1：DU[3:0] は曜日を表します。DT[1:0] は無視されます。

- ビット 29:28 **DT[1:0]**：BCD 形式での日の十の位
- ビット 27:24 **DU[3:0]**：BCD 形式での日の一の位または曜日

- ビット 23 **MSK3**：アラーム B 時マスク
- 0：時が一致すると、アラーム B がセットされます。
 - 1：アラーム B の比較では時を無視します。

- ビット 22 **PM**：AM/PM 表記
- 0：AM または 24 時間形式
 - 1：PM

- ビット 21:20 **HT[1:0]**：BCD 形式での時の十の位
- ビット 19:16 **HU[3:0]**：BCD 形式での時の一の位

- ビット 15 **MSK2**：アラーム B 分マスク
- 0：分が一致すると、アラーム B がセットされます。
 - 1：アラーム B の比較では分を無視します。

- ビット 14:12 **MNT[2:0]**：BCD 形式での分の十の位
- ビット 11:8 **MNU[3:0]**：BCD 形式での分の一の位

- ビット 7 **MSK1**：アラーム 秒マスク
- 0：秒が一致すると、アラーム B がセットされます。
 - 1：アラーム B の比較では秒を無視します。

- ビット 6:4 **ST[2:0]**：BCD 形式での秒の十の位
- ビット 3:0 **SU[3:0]**：BCD 形式での秒の一の位



27.6.9 RTC 書き込み保護レジスタ (RTC_WPR)

アドレスオフセット : 0x24

リセット値 : 0x0000 0000

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

ビット 31:8 予約済みであり、リセット値に保持する必要があります。

ビット 7:0 **KEY** : 書き込み保護キー

このバイトはソフトウェアで書き込まれます。

このバイトを読み出すと常に 0x00 が返されます。

RTC レジスタの書き込み保護解除方法については、[RTC レジスタ書き込み保護](#)を参照してください。

27.6.10 RTC サブセカンドレジスタ (RTC_SSR)

アドレスオフセット : 0x28

RTC ドメインリセット値 : 0x0000 0000

システムリセット : BYPSHAD = 0 の場合、0x0000 0000 です。BYPSHAD = 1 の場合、影響を受けません。

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS** : サブセカンド値

SS[15:0] は、同期プリスケアラのカウンタ内の値です。秒の小数部は、下の式によって与えられます。

秒の小数部 = (PREDIV_S - SS) / (PREDIV_S + 1)

注 : **SS** は、シフト操作後に限り、PREDIV_S より大きな値となる場合があります。この場合、正確な時刻/日付は、RTC_TR/RTC_DR で示される値よりも 1 秒少ない値となります。

27.6.11 RTC シフト制御レジスタ (RTC_SHIFTR)

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x2C

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

| | | | | | | | | | | | | | | | |
|-------|-------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| ADD1S | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| w | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | SUBFS[14:0] | | | | | | | | | | | | | | |
| | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

ビット 31 **ADD1S** : 1 秒加算

0 : 影響なし。

1 : 時計／カレンダーに 1 秒加算します。

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中 (RTC_ISR で SHPF = 1) の場合、このビットに書き込んでも影響はありません。

この機能は、SUBFS (下記説明を参照) と共に使用されることを想定しており、不可分操作で、効果的に時計に秒の小数部を加算することを目的としています。

ビット 30:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SUBFS** : 秒の小数部差し引き

このビットは書き込み専用であり、常に 0 として読み出されます。シフト操作が保留中 (RTC_ISR で SHPF = 1) の場合、このビットに書き込んでも影響はありません。

SUBFS に書き込まれた値は、同期プリスケアラのカウンタに加算されます。このカウンタはカウントダウンしていくので、この操作によって、次の式で求める値が効果的にクロックから差し引き(遅延)されます。

遅れ (秒) = SUBFS / (PREDIV_S + 1)

ADD1S 機能が SUBFS と共に用いられた場合、秒の小数部を効果的にクロックに加算する (クロックを進める) ことができ、実際のクロックの進みは次の式のとおりとなります。

進み (秒) = (1 - (SUBFS / (PREDIV_S + 1)))

注 : **SUBFS に書き込むことにより RSF はクリアされます。その後、ソフトウェアが RSF = 1 まで待つことにより、シャドウレジスタがシフトされた時刻で更新されていることが確実にになります。**

27.6.12 RTC タイムスタンプ時刻レジスタ（RTC_TSTR）

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x30

RTC ドメインリセット値：0x0000 0000

システムリセット：影響なし

| | | | | | | | | | | | | | | | |
|------|----------|------|------|----------|------|------|------|------|---------|---------|----|---------|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PM | HT[1:0] | | HU[3:0] | | | |
| | | | | | | | | | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | MNT[2:0] | | | MNU[3:0] | | | | Res. | ST[2:0] | | | SU[3:0] | | | |
| | r | r | r | r | r | r | r | | r | r | r | r | r | r | r |

ビット 31:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **PM** : AM/PM 表記

0 : AM または 24 時間形式
1 : PM

ビット 21:20 **HT[1:0]** : BCD 形式での時の十の位

ビット 19:16 **HU[3:0]** : BCD 形式での時の一の位

ビット 15 予約済みであり、リセット値に保持する必要があります。

ビット 14:12 **MNT[2:0]** : BCD 形式での分の十の位

ビット 11:8 **MNU[3:0]** : BCD 形式での分の一の位

ビット 7 予約済みであり、リセット値に保持する必要があります。

ビット 6:4 **ST[2:0]** : BCD 形式での秒の十の位

ビット 3:0 **SU[3:0]** : BCD 形式での秒の一の位

27.6.13 RTC タイムスタンプ日付レジスタ (RTC_TSDR)

このレジスタの内容は、RTC_ISR で TSF が 1 にセットされている場合にのみ有効です。また、TSF ビットがリセットされるとクリアされます。

アドレスオフセット : 0x34

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:13 **WDU[1:0]** : 曜日

ビット 12 **MT** : BCD 形式での月の十の位

ビット 11:8 **MU[3:0]** : BCD 形式での月の一の位

ビット 7:6 予約済みであり、リセット値に保持する必要があります。

ビット 5:4 **DT[1:0]** : BCD 形式での日の十の位

ビット 3:0 **DU[3:0]** : BCD 形式での日の一の位

27.6.14 RTC タイムスタンプサブセカンドレジスタ（RTC_TSSSR）

このレジスタの内容は、RTC_ISR/TSF がセットされている場合にのみ有効です。また、RTC_ISR/TSF ビットがリセットされるとクリアされます。

アドレスオフセット：0x38

RTC ドメインリセット値：0x0000 0000

システムリセット：影響なし

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

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15:0 **SS**：サブセカンド値

SS[15:0] は、タイムスタンプイベントが発生したときの同期プリスケアラのカウンタの値です。



27.6.15 RTC 較正レジスタ (RTC_CALR)

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[RTC レジスタ書き込み保護 \(580 ページ\)](#) を参照してください。

アドレスオフセット : 0x3C

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

| | | | | | | | | | | | | | | | |
|------|-------|--------|------|------|------|------|-----------|------|------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CALP | CALW8 | CALW16 | Res. | Res. | Res. | Res. | CALM[8:0] | | | | | | | | |
| rW | rW | rW | | | | | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:16 予約済みであり、リセット値に保持する必要があります。

ビット 15 **CALP** : RTC 周波数を 488.5 ppm 増加

0 : RTCCLK パルスは加えられません。

1 : RTCCLK の 2^{11} パルスごとに 1 パルス効果的に挿入されます (周波数が 488.5 ppm 増加)。

この機能は、CALM と共に使用されることを想定しており、カレンダーの周波数を高分解能で下げることができます。入力周波数が 32768 Hz の場合、32 秒枠の間に追加される RTCCLK パルスの数は次のように算出されます : $(512 * \text{CALP}) - \text{CALM}$

[セクション 27.3.12 : RTC の高精度デジタル較正](#) を参照してください。

ビット 14 **CALW8** : 8 秒較正サイクル周期の使用

CALW8 が“1”にセットされると、8 秒較正サイクル周期が選択されます。

注 : **CALW8 = 1 の場合、CALM[1:0] は“00”に固定されます。**[セクション 27.3.12 : RTC の高精度デジタル較正](#) を参照してください。

ビット 13 **CALW16** : 16 秒較正サイクル周期の使用

CALW16 が“1”にセットされると、16 秒較正サイクル周期が選択されます。CALW8 = 1 の場合、このビットを“1”にセットすることはできません。

注 : **CALW16 = 1 の場合、CALM[0] は“0”に固定されます。**[セクション 27.3.12 : RTC の高精度デジタル較正](#) を参照してください。

ビット 12:9 予約済みであり、リセット値に保持する必要があります。

ビット 8:0 **CALM[8:0]** : 較正マイナス

RTCCLK 2^{20} パルス (入力周波数が 32768 Hz の場合 32 秒) 内の CALM をマスクすることによって、カレンダーの周波数が下げられます。この方法により、カレンダーの周波数を 0.9537 ppm の分解能で下げることができます。

カレンダーの周波数を上げるには、この機能を CALP と共に使用する必要があります。[セクション 27.3.12 : RTC の高精度デジタル較正 \(584 ページ\)](#) を参照してください。

27.6.16 RTC タンパ設定レジスタ (RTC_TAMPCR)

アドレスオフセット : 0x40

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

| | | | | | | | | | | | | | | | |
|-----------|---------------|------|--------------|------|---------------|------|----|--------|------|---------|--------------|---------|---------|--------------|---------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | | | | TAMP2MF | TAMP2NOERASE | TAMP2IE | TAMP1MF | TAMP1NOERASE | TAMP1IE |
| | | | | | | | | rw | rw | rw | rw | rw | rw | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| TAMPPUDIS | TAMPPRCH[1:0] | | TAMPFLT[1:0] | | TAMPFREQ[2:0] | | | TAMPTS | Res. | Res. | TAMP2T RG | TAMP2E | TAMPIE | TAMP1T RG | TAMP1E |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | | | rw | rw | rw | rw | rw |

ビット 31:25 予約済みであり、リセット値に保持する必要があります。

ビット 24:22 予約済みであり、リセット値に保持する必要があります。

ビット 21 TAMP2MF : タンパ 2 マスクフラグ

0 : タンパ 2 イベントによってトリガイベントが生成され、TAMP2F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1 : タンパ 2 イベントによってトリガイベントが生成されます。TAMP2F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタは消去されません。

注 : TAMP2MF がセットされている場合、タンパ 2 割り込みを有効にすることはできません。

ビット 20 TAMP2NOERASE : タンパ 2 消去なし

0 : タンパ 2 イベントでバックアップレジスタは消去されます。

1 : タンパ 2 イベントでバックアップレジスタは消去されません。

ビット 19 TAMP2IE : タンパ 2 割り込みイネーブル

0 : TAMPIE = 0 の場合、タンパ 2 割り込みは無効です。

1 : タンパ 2 割り込みは有効です。

ビット 18 TAMP1MF : タンパ 1 マスクフラグ

0 : タンパ 1 イベントによってトリガイベントが生成され、TAMP1F はソフトウェアでクリアして、次のタンパイベントを検出できるようにする必要があります。

1 : タンパ 1 イベントによってトリガイベントが生成されます。TAMP1F はマスクされ、ハードウェアによって内部でクリアされます。バックアップレジスタは消去されません。

注 : TAMP1MF がセットされている場合、タンパ 1 割り込みを有効にすることはできません。

ビット 17 TAMP1NOERASE : タンパ 1 消去なし

0 : タンパ 1 イベントでバックアップレジスタは消去されます。

1 : タンパ 1 イベントでバックアップレジスタは消去されません。

ビット 16 TAMP1IE : タンパ 1 割り込みイネーブル

0 : TAMPIE = 0 の場合、タンパ 1 割り込みは無効です。

1 : タンパ 1 割り込みは有効です。

ビット 15 TAMPPUDIS : RTC_TAMPx プルアップディセーブル

このビットにより、毎回のサンプリング前に各 RTC_TAMPx ピンをプリチャージするかどうか決定します。

0 : サンプリング前に RTC_TAMPx ピンをプリチャージします (内部プルアップを有効化)。

1 : RTC_TAMPx ピンのプリチャージを無効化します。

ビット 14:13 **TAMPPRCH[1:0]** : RTC_TAMPx プリチャージ時間

これらのビットにより、各サンプリングの前にプルアップを有効化している時間を決定します。TAMPPRCH は、各 RTC_TAMPx 入力に対して有効です。

0x0 : 1 RTCCLK サイクル

0x1 : 2 RTCCLK サイクル

0x2 : 4 RTCCLK サイクル

0x3 : 8 RTCCLK サイクル

ビット 12:11 **TAMPFLT[1:0]** : RTC_TAMPx フィルタカウント

これらのビットにより、タンパイイベントをアクティブにするのに必要な指定のレベル (TAMP*TRG) での連続サンプリングの数を決定します。TAMPFLT は、各 RTC_TAMPx 入力に対して有効です。

0x0 : RTC_TAMPx 入力がアクティブレベル (RTC_TAMPx 入力における内部プルアップ無し) に遷移するときのエッジで、タンパイイベントがアクティブになります。

0x1 : アクティブレベルでの連続した 2 回のサンプリングの後、タンパイイベントがアクティブになります。

0x2 : アクティブレベルでの連続した 4 回のサンプリングの後、タンパイイベントがアクティブになります。

0x3 : アクティブレベルでの連続した 8 回のサンプリングの後、タンパイイベントがアクティブになります。

ビット 10:8 **TAMPFREQ[2:0]** : タンパサンプリング周波数

各 RTC_TAMPx 入力がサンプリングされる周波数を決定します。

0x0 : RTCCLK / 32768 (RTCCLK = 32768 Hz の場合 1 Hz)

0x1 : RTCCLK / 16384 (RTCCLK = 32768 Hz の場合 2 Hz)

0x2 : RTCCLK / 8192 (RTCCLK = 32768 Hz の場合 4 Hz)

0x3 : RTCCLK / 4096 (RTCCLK = 32768 Hz の場合 8 Hz)

0x4 : RTCCLK / 2048 (RTCCLK = 32768 Hz の場合 16 Hz)

0x5 : RTCCLK / 1024 (RTCCLK = 32768 Hz の場合 32 Hz)

0x6 : RTCCLK / 512 (RTCCLK = 32768 Hz の場合 64 Hz)

0x7 : RTCCLK / 256 (RTCCLK = 32768 Hz の場合 128 Hz)

ビット 7 **TAMPTS** : タンパ検出イベント時のタイムスタンプの有効化

0 : タンパ検出イベントがあっても、タイムスタンプは保存されません。

1 : タンパ検出イベント時、タイムスタンプが保存されます。

RTC_CR レジスタで TSE = 0 であっても TAMPTS は有効です。

ビット 6:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TAMP2TRG** : RTC_TAMP2 入力のアクティブレベル

TAMPFLT != 00 の場合

0 : RTC_TAMP2 入力がローのままのとき、タンパ検出イベントがトリガされます。

1 : RTC_TAMP2 入力がハイのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0 : RTC_TAMP2 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1 : RTC_TAMP2 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 3 **TAMP2E** : RTC_TAMP2 入力検出イネーブル

0 : RTC_TAMP2 検出は無効です。

1 : RTC_TAMP2 検出は有効です。

ビット 2 **TAMPIE** : タンパ割り込みイネーブル

0 : タンパ割り込みは無効です。

1 : タンパ割り込みは有効です。

注 : このビットは、**TAMPxIE** レベルにかかわらず、すべてのタンパピンイベントに対する割り込みを有効にします。このビットがクリアされた場合、**TAMPxIE** をセットすることによって、各タンパイベントの割り込みを個別に有効にすることができます。

ビット 1 **TAMP1TRG** : RTC_TAMP1 入力のアクティブレベル

TAMPFLT != 00 の場合

0 : RTC_TAMP1 入力がローのままのとき、タンパ検出イベントがトリガされます。

1 : RTC_TAMP1 入力が高いのままのとき、タンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合

0 : RTC_TAMP1 入力の立ち上がりエッジでタンパ検出イベントがトリガされます。

1 : RTC_TAMP1 入力の立ち下がりエッジでタンパ検出イベントがトリガされます。

ビット 0 **TAMP1E** : RTC_TAMP1 入力検出イネーブル

0 : RTC_TAMP1 検出は無効です。

1 : RTC_TAMP1 検出は有効です。

注意 : TAMPFLT = 0 の場合、TAMP1E ビットは、TAMP1TRG が変更されたらリセットし、TAMP1F の不要なセットを回避する必要があります。

27.6.17 RTC アラーム A サブセカンドレジスタ (RTC_ALRMASR)

このレジスタは、RTC_CR レジスタの ALRAIE がリセットされた場合、または初期化モードの場合にのみ書き込み可能です。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、RTC レジスタ書き込み保護を参照してください。[RTC レジスタ書き込み保護 \(580 ページ\)](#)

アドレスオフセット : 0x44

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 MASKSS[3:0] : このビットから始まる最上位ビットのマスク

- 0 : アラーム A に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます (他の項目が一致していることを前提として)。
- 1 : アラーム A の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
- 2 : アラーム A の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
- 3 : アラーム A の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
- ...
- 12 : アラーム A の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
- 13 : アラーム A の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
- 14 : アラーム A の比較では SS[14] を無視します。SS[13:0] が比較されます。
- 15 : アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。同期カウンタのオーバーフロービット (ビット 15) が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 SS[14:0] : サブセカンド値

この値が同期プリスケアラのカウンタの内容と比較され、アラーム A をアクティブ化するかどうかを決定します。0~MASKSS-1 のビットだけが比較されます。

27.6.18 RTC アラーム B サブセカンドレジスタ（RTC_ALRMBSSR）

このレジスタは、RTC_CR レジスタの ALRBE がリセットされた場合、または初期化モードの場合にのみ書き込みめます。

このレジスタは書き込み保護されています。書き込みアクセスの手順は、[セクション : RTC レジスタ書き込み保護](#)を参照してください。

アドレスオフセット : 0x48

RTC ドメインリセット値 : 0x0000 0000

システムリセット : 影響なし

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

ビット 31:28 予約済みであり、リセット値に保持する必要があります。

ビット 27:24 **MASKSS[3:0]** : このビットから始まる最上位ビットのマスク

- 0x0 : アラーム B に対してサブセカンドを比較しません。このアラームは秒の位がインクリメントされたときにセットされます（他の項目が一致していることを前提として）。
- 0x1 : アラーム B の比較では SS[14:1] を無視します。SS[0] のみ比較されます。
- 0x2 : アラーム B の比較では SS[14:2] を無視します。SS[1:0] のみ比較されます。
- 0x3 : アラーム B の比較では SS[14:3] を無視します。SS[2:0] のみ比較されます。
- ...
- 0xC : アラーム B の比較では SS[14:12] を無視します。SS[11:0] が比較されます。
- 0xD : アラーム B の比較では SS[14:13] を無視します。SS[12:0] が比較されます。
- 0xE : アラーム B の比較では SS[14] を無視します。SS[13:0] が比較されます。
- 0xF : アラームをアクティブにするには、15 の全ての SS ビットを比較し一致する必要があります。同期カウンタのオーバーフロービット（ビット 15）が比較されることはありません。このビットは、シフト操作後に限り、0 でなくなる場合があります。

ビット 23:15 予約済みであり、リセット値に保持する必要があります。

ビット 14:0 **SS[14:0]** : サブセカンド値

この値が同期プリスケアラのカウンタの内容と比較され、アラーム B をアクティブ化するかどうかを決定します。0~MASKSS-1 のビットだけが比較されます。



27.6.19 RTC オプションレジスタ (RTC_OR)

アドレスオフセット : 0x4C
パワーオンリセット値 : 0x0000 0000
システムリセット : 影響なし

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

- ビット 31:2 予約済みであり、リセット値に保持する必要があります。
- ビット 1 **RTC_OUT_RMP** : RTC_OUT 再配置
- このビットを設定することで、以下のように、PB14 の RTC 出力を再配置することが可能になります。
- RTC_OUT_RMP = 0 の場合**
- OSEL が"00"以外するとき、PC13 に RTC_ALARM が出力されます。
- OSEL が"00"かつ COE が"1"のとき、PC13 に RTC_CALIB が出力されます。
- RTC_OUT_RMP = 1 の場合**
- OSEL が"00"以外かつ COE が"0"のとき、PB14に RTC_ALARM が出力されます。
- OSEL が"00"かつ COE が"1"のとき、PB14に RTC_CALIB が出力されます。
- OSEL が"00"以外かつ COE が"1"のとき、PB14 に RTC_CALIB が、PC13 に RTC_ALARM が出力されます。
- ビット 0 **RTC_ALARM_TYPE** : PC13 への RTC_ALARM の出力タイプ
- 0 : RTC_ALARM は、PC13 に配置される場合は、オープンドレイン出力です。
- 1 : RTC_ALARM は、PC13 に配置される場合は、プッシュプル出力です。

27.6.20 RTC バックアップレジスタ (RTC_BKPxR)

アドレスオフセット : 0x50~0x60
RTC ドメインリセット値 : 0x0000 0000
システムリセット : 影響なし

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

ビット 31:0 BKP[31:0]
アプリケーションはこれらのレジスタに対してデータの読み書きをすることができます。
レジスタの内容は、デバイスが低電力モードで動作する場合、有効なまま保持されます。
このレジスタは、TAMPxP = 1 である限り、あるいはフラッシュ読み出し保護が無効なときは、タンパ検出イベント時にリセットされます。

27.6.21 RTC レジスタマップ

表 99. RTC レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | |
|-----------|------------|-------|-------|-------------|----|---------|----|----|------|---------|---------------|-------------|---------|---------|-------|-------|----------------|-----------|-------------|--------|---------|----------|---------|-------|-----------|-------------|-------------|---------|---------|--------|------------------|--------|---|---|--|--|--|--|
| 0x00 | RTC_TR | Res | | | | | | | | | PM | HT [1:0] | HU[3:0] | | | | Res | MNT[2:0] | | | | MNU[3:0] | | | | Res | ST[2:0] | | | | SU[3:0] | | | | | | | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x04 | RTC_DR | Res | | | | | | | | YT[3:0] | | | | YU[3:0] | | | | WDU[2:0] | | MT | MU[3:0] | | | | Res | Res | DT [1:0] | | | | DU[3:0] | | | | | | | |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | | | 0 | 0 | 0 | 0 | 0 | 1 | | | | |
| 0x08 | RTC_CR | Res | | | | | | | | COE | OSEL [1:0] | POL | COSEL | BKP | SUB1H | ADD1H | TSIE | WUTIE | ALRBIE | ALRAIE | TSE | WUTE | ALRBE | ALRAE | Res | Res | FMT | BYPHAD | REFCKON | TSEDGE | WUCKSE L[2:0] | | | | | | | |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x0C | RTC_ISR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | TAMP2F | TAMP1F | TSOVF | TSF | WUTF | ALRBF | ALRAF | INIT | INTF | RSF | INTS | SHPF | WUT WF | ALRBWF | ALRAWF | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | | | | |
| 0x10 | RTC_PRER | Res | | | | | | | | Res | PREDIV_A[6:0] | | | | | | PREDIV_S[14:0] | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | |
| 0x14 | RTC_WUTR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | WUT[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | | | | | |
| 0x1C | RTC_ALRMAR | MSK4 | WDSEL | DT [1:0] | | DU[3:0] | | | MSK3 | PM | HT [1:0] | HU[3:0] | | | | MSK2 | MNT[2:0] | | MNU[3:0] | | | MSK1 | ST[2:0] | | SU[3:0] | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x20 | RTC_ALRMBR | MSK4 | WDSEL | DT [1:0] | | DU[3:0] | | | MSK3 | PM | HT [1:0] | HU[3:0] | | | | MSK2 | MNT[2:0] | | MNU[3:0] | | | MSK2 | ST[2:0] | | SU[3:0] | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x24 | RTC_WPR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | Res | KEY | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x28 | RTC_SSR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | SS[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | |
| 0x2C | RTC_SHIFTR | ADD1S | Res | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | Res | SUBFS[14:0] | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x30 | RTC_TSTR | Res | | | | | | | | PM | HT [1:0] | HU[3:0] | | | | Res | MNT[2:0] | | MNU[3:0] | | | Res | ST[2:0] | | SU[3:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x34 | RTC_TSDR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | WDU[1:0] | | MT | MU[3:0] | | | Res | Res | DT [1:0] | | DU[3:0] | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x38 | RTC_TSSSR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | SS[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x3C | RTC_CALR | Res | | | | | | | | Res | Res | Res | Res | Res | Res | Res | Res | CALP | CALW8 | CALW16 | Res | Res | Res | Res | CALM[8:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |

表 99. RTC レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------------|--------------|-----------|------|------|------|-----------------|------|------|------|------|------|---------|--------------|---------|---------|--------------|---------|-----------|---------------|------|--------------|------|---------------|------|------|--------|------|------|-----------|--------|--------|--------------------------------|--------|
| 0x40 | RTC_TAMPCR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TAMP2MF | TAMP2NOERASE | TAMP2IE | TAMP1MF | TAMP1NOERASE | TAMP1IE | TAMPPUDIS | TAMPPRCH[1:0] | | TAMPFLT[1:0] | | TAMPFREQ[2:0] | | | TAMPTS | Res. | Res. | TAMP2-TRG | TAMP2E | TAMP1E | TAMP1TRG | TAMP1E |
| | リセット値 | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 |
| 0x44 | RTC_ALRMASSR | Res. | Res. | Res. | Res. | MASKSS [3:0] | | | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SS[14:0] | | | | | | | | | | | | | | |
| | リセット値 | | | | | 0 | 0 | 0 | 0 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x48 | RTC_ALRMASSR | Res. | Res. | Res. | Res. | MASKSS [3:0] | | | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | SS[14:0] | | | | | | | | | | | | | | |
| | リセット値 | | | | | 0 | 0 | 0 | 0 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x4C | RTC_OR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RTC_OUT_RMP. RTC_ALARM_TYPE | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x50 to 0x60 | RTC_BKP0R | BKP[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | RTC_BKP4R へ | BKP[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

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

28.1 概要

I²C (Inter-integrated circuit) バスインタフェースは、マイクロコントローラとシリアル I²C バス間の通信を処理します。マルチマスタ機能を備え、すべての I²C バス固有のシーケンシング、プロトコル、アービトレーション、およびタイミングを制御します。標準モード (Sm)、高速モード (Fm)、および高速モードプラス (Fm+) をサポートします。

また、SMBus (System Management Bus) および PMBus (Power Management Bus) と互換性があります。

DMA を使用して、CPU の負荷を軽減できます。

28.2 I2C の主な機能

- I²C バス仕様 rev03 との互換性 :
 - スレーブおよびマスタモード
 - マルチマスタ機能
 - 標準モード (最大 100 kHz)
 - 高速モード (最大 400 kHz)
 - 高速モードプラス (最大 1 MHz)
 - 7ビットおよび 10 ビットアドレッシングモード
 - 複数の 7 ビットスレーブアドレス (2 つのアドレス、1 つは設定可能なマスク付き)
 - すべての 7 ビットアドレス確認応答モード
 - 同報 (General call) コール
 - プログラム可能なセットアップおよびホールド時間
 - 使いやすいイベント管理
 - クロックストレッチオプション
 - ソフトウェアリセット

- DMA 機能付きの 1 バイトバッファ
- プログラム可能なアナログおよびデジタルノイズフィルタ

製品の実装によっては、次の追加機能も使用できます ([セクション 28.3 : I2C の実装](#) を参照) :

- SMBus 仕様 rev 2.0 との互換性 :
 - ハードウェア PEC (Packet Error Checking) の生成と ACK 制御による確認
 - コマンドおよびデータ確認応答制御
 - アドレス解決プロトコル (ARP) サポート
 - ホストおよびデバイスのサポート
 - SMBus アラート
 - タイムアウトおよびアイドル条件の検出
- PMBus rev 1.1 標準との互換性
- 独立したクロック : 独立したクロックソースの選択により、I2C の 通信速度は PCLK の再プログラミングから独立
- アドレス一致時に STOP モードからウェイクアップ



28.3 I2C の実装

このマニュアルでは、I2C1 で実装されている詳細な機能セットについて説明します。I2C2 は、より少ない機能セットをサポートしますが、それ以外はI2C1 と同じです。違いを次の表に示します。

表 100. STM32L0x3 I2C の機能

| I2C の機能 ⁽¹⁾ | I2C1 | I2C2 |
|------------------------|------|------------------|
| 独立クロック | X | |
| SMBus | X | |
| STOP モードからのウェイクアップ | X | |
| Fm+ モードのための 20 mA 出力駆動 | X | X ⁽²⁾ |

- 1. X = サポート。
- 2. この機能をサポートする I/O のリストについては、データシートを参照してください。

28.4 I2C 機能の詳細

データの送受信に加えて、このインタフェースは、データをシリアル形式からパラレル形式（およびその逆）に変換します。割り込みは、ソフトウェアによって有効または無効にできます。このインタフェースは、データピン（SDA）とクロックピン（SCL）によって I2C バスに接続されます。標準（最大 100 kHz）、高速モード（最大 400 kHz）、または高速モードプラス（最大 1 MHz）の I²C バスで接続できます。

このインタフェースは、データピン（SDA）とクロックピン（SCL）によって SMBus に接続することもできます。

SMBus 機能がサポートされる場合、追加の SMBus アラートピン（SMBA）オプションも使用できます。

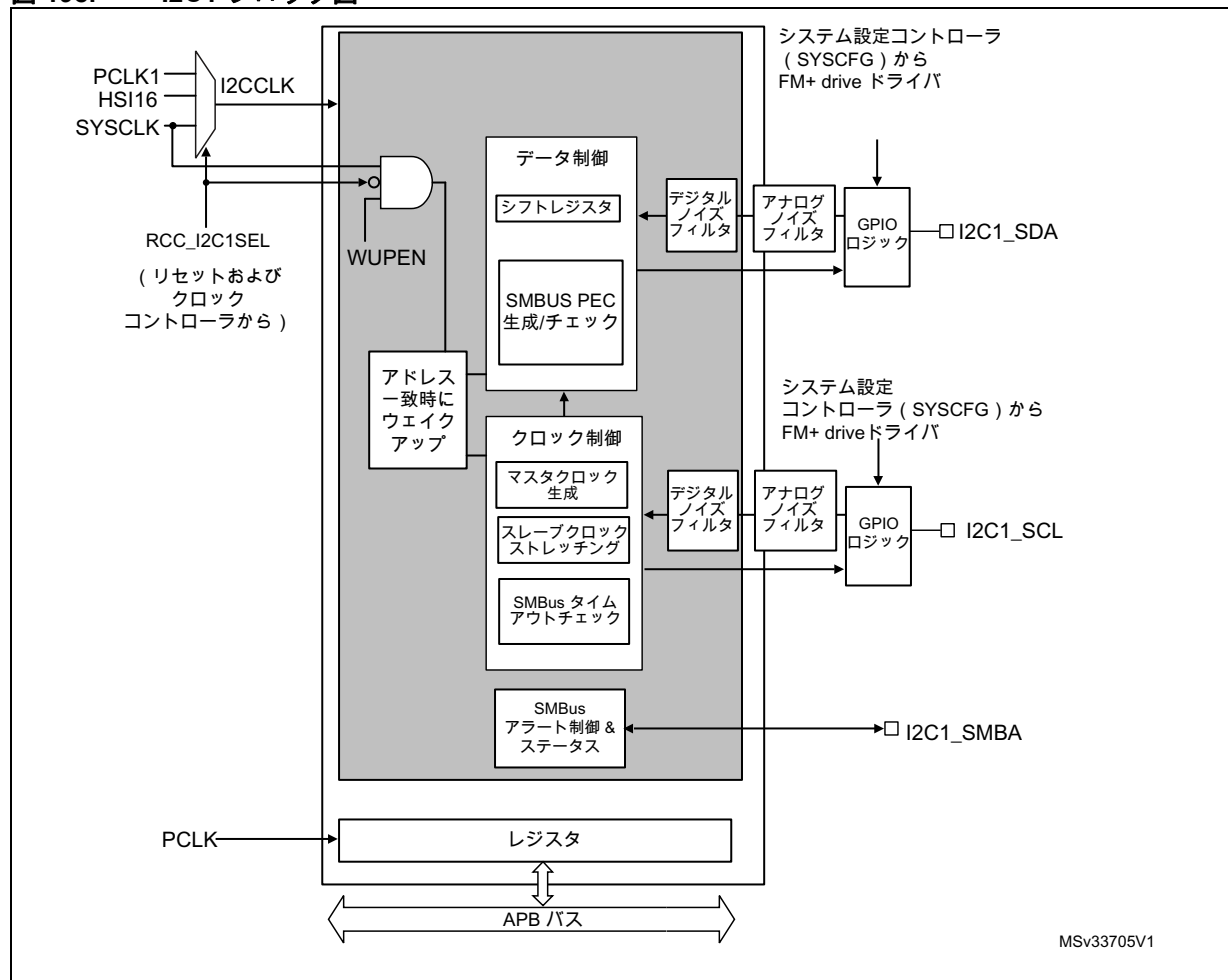


この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

28.4.1 I2C1 ブロック図

I2C インタフェースのブロック図を [図 198](#) に示します。

図 198. I2C1 ブロック図



I2C は、独立したクロックソースによってクロック供給されるため、I2C は PCLK 周波数から独立して動作できます。

この独立したクロックソースは、次の 3 つのクロックソースから選択できます：

- PCLK1 : APB1 クロック (デフォルト値)
- HSI16 : 内部 16MHz RC オシレータ
- SYSCLK : システムクロック

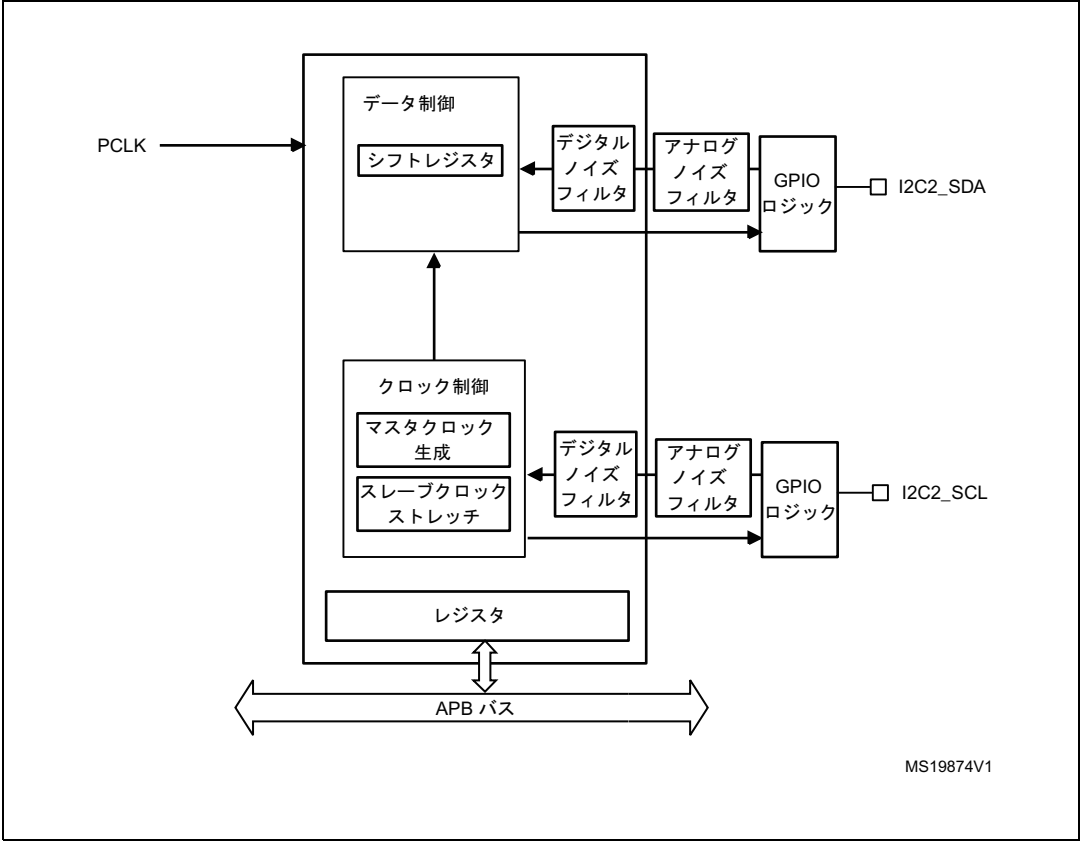
詳細については、[セクション 7 : リセットおよびクロック制御 \(RCC\)](#) を参照してください。

I2C I/O は、高速モードプラス動作のための 20 mA 出力電流駆動をサポートします。これを有効にするには、[セクション 10.2.2: SYSCFG ペリフェラルモード設定レジスタ \(SYSCFG_CFGR2\)](#) の SCL および SDA の駆動機能制御ビットをセットします。

28.4.2 I2C2 ブロック図

I2C2 インタフェースのブロック図を [図 199](#) に示します。

図 199. I2C2 ブロック図



28.4.3 I2C クロックの要件

I2C カーネルは I2CCLK によってクロック供給されます。

I2CCLK の周期 t_{I2CCLK} は、次の条件を満たす必要があります。

$$t_{I2CCLK} < (t_{LOW} - t_{filters}) / 4 \text{ and } t_{I2CCLK} < t_{HIGH}$$

ここで：

t_{LOW} : SCL ロー時間、および t_{HIGH} : SCL ハイ時間

$t_{filters}$: 有効なときには、アナログフィルタとデジタルフィルタによる遅延の合計。

アナログフィルタの遅延は、最大 260 ns です。デジタルフィルタの遅延は、 $DNF \times t_{I2CCLK}$ です。

PCLK の周期 t_{PCLK} は、次の条件を満たす必要があります。

$$t_{PCLK} < 4/3 t_{SCL}$$

t_{SCL} : SCL 周期

注意： I2C カーネルが PCLK によってクロック供給されるとき、PCLK は t_{I2CCLK} の条件を満たす必要があります。

28.4.4 モード選択

このインタフェースは、次の 4 つのモードのいずれかで動作できます：

- スレーブトランスミッタ
- スレーブレシーバ
- マスタトランスミッタ
- マスタレシーバ

デフォルトでは、スレーブモードで動作します。このインタフェースは、START コンディションを生成したときにはスレーブからマスタへ、アービトレーションの喪失または STOP 生成が発生したときにはマスタからスレーブへ自動的に切り替わるため、マルチマスタ機能を使用できます。

通信の流れ

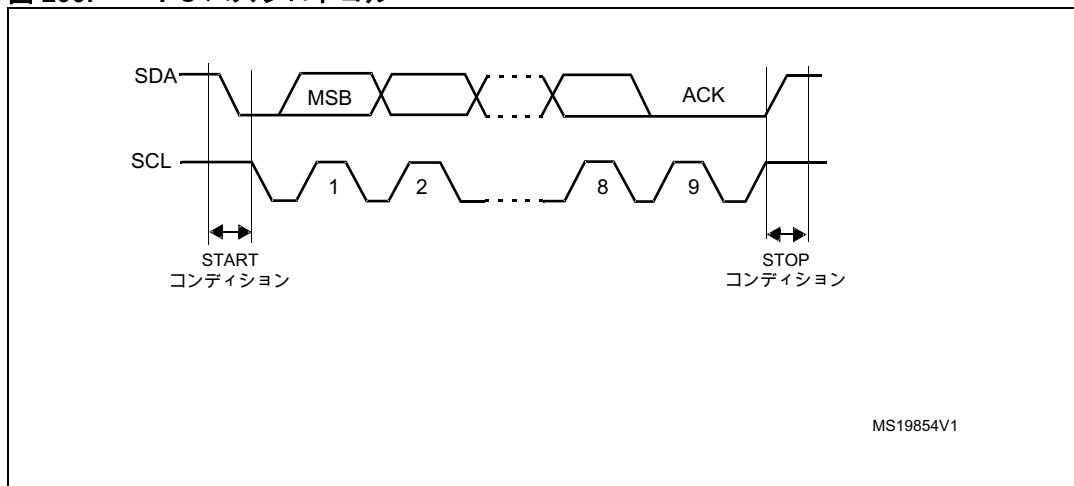
マスタモードでは、I2C インタフェースは、データ転送を開始し、クロック信号を生成します。シリアルデータ転送は、常に START コンディションで開始され、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

スレーブモードでは、このインタフェースは、自己アドレス (7 または 10 ビット) と同報アドレスを認識できます。同報アドレスの検出は、ソフトウェアによって有効または無効にできます。予約済みの SMBus アドレスもソフトウェアによって有効にできます。

データとアドレスは、MSB ファーストの 8 ビットバイトとして転送されます。START コンディションの後に続く最初のバイト (7 ビットモードでは 1 バイト、10 ビットモードでは 2 バイト) にアドレスが含まれています。アドレスは、常にマスタモードで送信されます。

8 クロックサイクルのバイト転送の後には 9 番目のクロックパルスが続きます。その間に、レシーバはトランスミッタに確認応答ビットを送信する必要があります。次の図を参照してください。

図 200. I²C バスプロトコル



確認応答 (Acknowledge) は、ソフトウェアによって有効または無効にできます。I2C インタフェースのアドレスは、ソフトウェアによって選択できます。

28.4.5 I2C の初期化

ペリフェラルの有効化と無効化

I2C ペリフェラルクロックは、クロックコントローラで設定し、有効にする必要があります ([セクション 7: リセットおよびクロック制御 \(RCC\)](#) を参照)。

そして、I2Cx_CR1 レジスタの PE ビットをセットすることによって、I2C を有効にできます。

I2C が無効なときには (PE=0)、I²C はソフトウェアリセットを実行します。詳細については、[セクション 28.4.6: ソフトウェアリセット](#) を参照してください。

ノイズフィルタ

I2Cx_CR1 レジスタの PE ビットをセットすることによって I2C ペリフェラルを有効にする前に、必要な場合は、ノイズフィルタを設定する必要があります。デフォルトでは、SDA および SCL 入力にアナログノイズフィルタがあります。このアナログフィルタは I²C 仕様に準拠しており、高速モードおよび高速モードプラスで最大 50 ns のパルス幅を持つスパイクを抑制します。ANFOFF ビットをセットすることによって、このアナログフィルタを無効にし、I2Cx_CR1 レジスタの DNF[3:0] ビットを設定することによってデジタルフィルタを選択することができます。

デジタルフィルタが有効なときには、SCL または SDA ラインのレベルは、DNF x I2CCLK 周期より長く安定していた場合のみ、内部で変更されます。これにより、プログラム可能な 1 ~ 15 I2CCLK 周期の長さを持つスパイクを抑制できます。

表 101. アナログフィルタとデジタルフィルタの比較

| | アナログフィルタ | デジタルフィルタ |
|--------------------|---------------------|---|
| 抑制される スパイクのパルス幅 | ≥ 50 ns | 長さを 1 ~ 15 I2C ペリフェラルクロックにプログラム可能 |
| 利点 | STOP モードで使用可能 | – プログラム可能な長さ: 追加のフィルタリング機能対標準要件 – 安定した長さ |
| 欠点 | 温度、電圧、 プロセスのばらつき | デジタルフィルタが有効なときには、アドレス一致時の STOP モードからのウェイクアップは使用できない |

注意 : I2C が有効なときには、フィルタ構成の変更はできません。

マスタおよびスレーブモードで正しいデータホールドおよびセットアップ時間が使用されるのを保証するためには、タイミングを設定する必要があります。これを行うには、I2Cx_TIMINGR レジスタの PRESC[3:0]、SCLDEL[3:0]、および SDADEL[3:0] ビットをプログラムします。

データホールド時間

SCL 立ち下がりエッジ内部検出

t_{SYNC1} SDA_{DEL}

SDA 出力遅延

SCL

SDA

$t_{\text{HD,DAT}}$

データホールド時間

データセットアップ時間

SCL_{DEL}

SCL スレーブトランスミッタによるSCLのストレッチされたローにストレッチ

SCL

SDA

$t_{\text{SU,DAT}}$

データセットアップ時間

- SCL 立ち下がりエッジが内部で検出されると、SDA 出力を送信する前に遅延が挿入されます。この遅延は、 $t_{SDADEL} = SDADEL \times t_{PRESC} + t_{I2CCLK}$ であり $t_{PRESC} = (PRESC+1) \times t_{I2CCLK}$ です。 T_{SDADEL} はホールド時間 $t_{HD:DAT}$ に影響を与えます。

SDA 出力遅延の合計は、次のとおりです：

$$t_{\text{SYNC1}} + \{ [\text{SDADEL} \times (\text{PRESC} + 1) + 1] \times t_{\text{I2CCLK}} \}$$

t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- アナログフィルタが有効なときの入力遅延: $t_{\text{AF(min)}} < t_{\text{AF}} < t_{\text{AF(max)}} \text{ ns}$
- デジタルフィルタが有効なときの入力遅延: $t_{\text{DNF}} = \text{DNF} \times t_{\text{I2CCLK}}$
- SCL と I2CCLK クロックの同期による遅延 (2 ~ 3 I2CCLK 周期)

SCL 立ち下がりエッジの未定義の領域をブリッジするためには、SDADEL を次のようにプログラムする必要があります:

$$\{t_{\text{f(max)}} + t_{\text{HD;DAT(min)}} - t_{\text{AF(min)}} - [(DNF + 3) \times t_{\text{I2CCLK}}]\} / \{(PRESC + 1) \times t_{\text{I2CCLK}}\} \leq \text{SDADEL}$$

$$\text{SDADEL} \leq \{t_{\text{HD;DAT(max)}} - t_{\text{AF(max)}} - [(DNF + 4) \times t_{\text{I2CCLK}}]\} / \{(PRESC + 1) \times t_{\text{I2CCLK}}\}$$

注: $t_{\text{AF(min)}} / t_{\text{AF(max)}}$ は、アナログフィルタが有効なときのみ、等式に含まれます。 t_{AF} の値については、デバイスのデータシートを参照してください。

最大 $t_{\text{HD;DAT}}$ は、標準モード、高速モード、および高速モードプラスで 3.45 μs 、0.9 μs 、および 0.45 μs ですが、遷移時間による $t_{\text{VD;DAT}}$ の最大値より短い必要があります。この最大値を満たす必要があるのは、デバイスが SCL 信号の LOW 周期 (t_{LOW}) をストレッチしない場合だけです。クロックが SCL をストレッチする場合、クロックをリリースする前に、データがセットアップ時間まで有効である必要があります。

SDA 立ち上がりエッジは、通常、最悪ケースであり、この場合、前の等式は次のようになります:

$$\text{SDADEL} \leq \{t_{\text{VD;DAT(max)}} - t_{\text{r(max)}} - 260 \text{ ns} - [(DNF + 4) \times t_{\text{I2CCLK}}]\} / \{(PRESC + 1) \times t_{\text{I2CCLK}}\}$$

注: **NOSTRETCH=0 のときには、SCLDEL の値に従って、デバイスはセットアップ時間を保証するために SCL ローをストレッチするので、この条件に違反することがあります。**

t_{f} 、 t_{r} 、 $t_{\text{HD;DAT}}$ 、および $t_{\text{VD;DAT}}$ の標準値については、表 102: I2C-SMBUS 仕様のデータのセットアップおよびホールド時間を参照してください。

- SDA 出力の送信後、SCL ラインはセットアップ時間中、ローレベルに保たれます。このセットアップ時間は、 $t_{\text{SCLDEL}} = (\text{SCLDEL} + 1) \times t_{\text{PRESC}}$ であり $t_{\text{PRESC}} = (\text{PRESC} + 1) \times t_{\text{I2CCLK}}$ です。

t_{SCLDEL} は、セットアップ時間 $t_{\text{SU;DAT}}$ に影響を与えます。

SDA 遷移 (立ち上がりエッジは通常、最悪のケース) の未定義の領域をブリッジするためには、SCLDEL を次のようにプログラムする必要があります:

$$\{[t_{\text{r(max)}} + t_{\text{SU;DAT(min)}}] / [(\text{PRESC} + 1)] \times t_{\text{I2CCLK}}\} - 1 \leq \text{SCLDEL}$$

t_{r} および $t_{\text{SU;DAT}}$ の標準値については、表 102: I2C-SMBUS 仕様のデータのセットアップおよびホールド時間を参照してください。

使用される SDA および SCL 遷移時間の値は、アプリケーションの値です。標準から最大値を使用すると、SDADEL と SCLDEL の計算の制約が増えますが、アプリケーションにかかわらず、この機能を使用できます。

表 102. I2C-SMBUS 仕様のデータのセットアップおよびホールド時間

| 記号 | パラメータ | 標準モード | | 高速モード | | 高速トモードプラス | | SMBUS | | 単位 |
|--------------|------------------------|-------|------|-------|-----|-----------|------|-------|------|---------|
| | | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | |
| $t_{HD;DAT}$ | データホールド時間 | 0 | | 0 | | 0 | | 0.3 | | μs |
| $t_{VD;DAT}$ | データ有効時間 | | 3.45 | | 0.9 | | 0.45 | | | |
| $t_{SU;DAT}$ | データセットアップ時間 | 250 | | 100 | | 50 | | 250 | | ns |
| t_r | SDA および SCL 信号の立ち上がり時間 | | 1000 | | 300 | | 120 | | 1000 | |
| t_f | SDA および SCL 信号の立ち下がり時間 | | 300 | | 300 | | 120 | | 300 | |

また、マスタモードでは、I2Cx_TIMINGR レジスタの PRESC[3:0]、SCLH[7:0]、および SCLL[7:0] ビットをプログラムすることによって、SCL クロックのハイおよびローレベルを設定する必要があります。

- SCL 立ち下がりエッジが内部で検出されると、SCL 出力をリリースする前に遅延が挿入されます。この遅延は、 $t_{SCLL} = (SCLL+1) \times t_{PRESC}$ であり、 $t_{PRESC} = (PRESC+1) \times t_{I2CCLK}$ です。
 t_{SCLL} は、SCL ロー時間 t_{LOW} に影響を与えます。
- SCL 立ち上がりエッジが内部で検出されると、SCL 出力を強制的にローレベルにする前に遅延が挿入されます。この遅延は、 $t_{SCLH} = (SCLH+1) \times t_{PRESC}$ であり、 $t_{PRESC} = (PRESC+1) \times t_{I2CCLK}$ です。 t_{SCLH} は、SCLハイ時間 t_{HIGH} に影響を与えます。

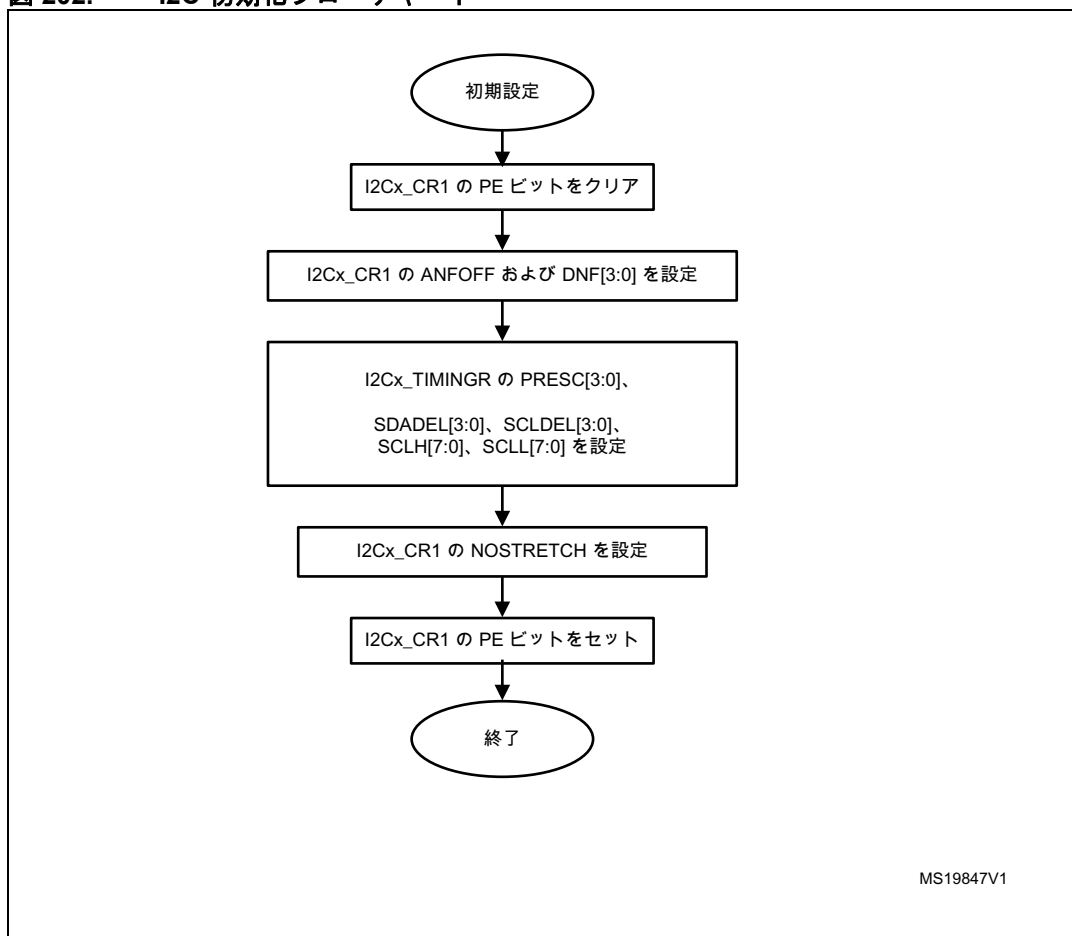
詳細については、セクション [I2C マスタ初期化](#) を参照してください。

注意： I2C が有効なときには、タイミング構成の設定はできません。

ペリフェラルを有効にする前に、I2C スレーブ NOSTRETCH モードも設定する必要があります。詳細については、[I2C スレーブ初期化](#) を参照してください。

注意： I2C が有効なときには、NOSTRETCH 構成の変更はできません。

図 202. I2C 初期化フローチャート



28.4.6 ソフトウェアリセット

ソフトウェアリセットを行うには、I2Cx_CR1 レジスタの PE ビットをクリアします。その場合、I2C のライン SCL および SDA がリリースされます。内部状態マシンがリセットされ、通信制御ビットとステータスビットがリセット値に戻ります。構成レジスタは影響を受けません。

影響を受けるレジスタのビットは、以下のとおりです：

1. I2Cx_CR2 レジスタ：START、STOP、NACK
2. I2Cx_ISR レジスタ：BUSY、TXE、TXIS、RXNE、ADDR、NACKF、TCR、TC、STOPF、BERR、ARLO、OVR

SMBus 機能がサポートされるときには、以下も影響を受けます：

1. I2Cx_CR2 レジスタ：PECBYTE
2. I2Cx_ISR レジスタ：PECERR、TIMEOUT、ALERT

ソフトウェアリセットを実行するためには、PE は少なくとも 3 APB クロックサイクルの間、ローに保たなければなりません。このためには、次のソフトウェアシーケンスを書き込みます：- PE=0 を書き込む - PE=0 を確認する - PE=1 を書き込む

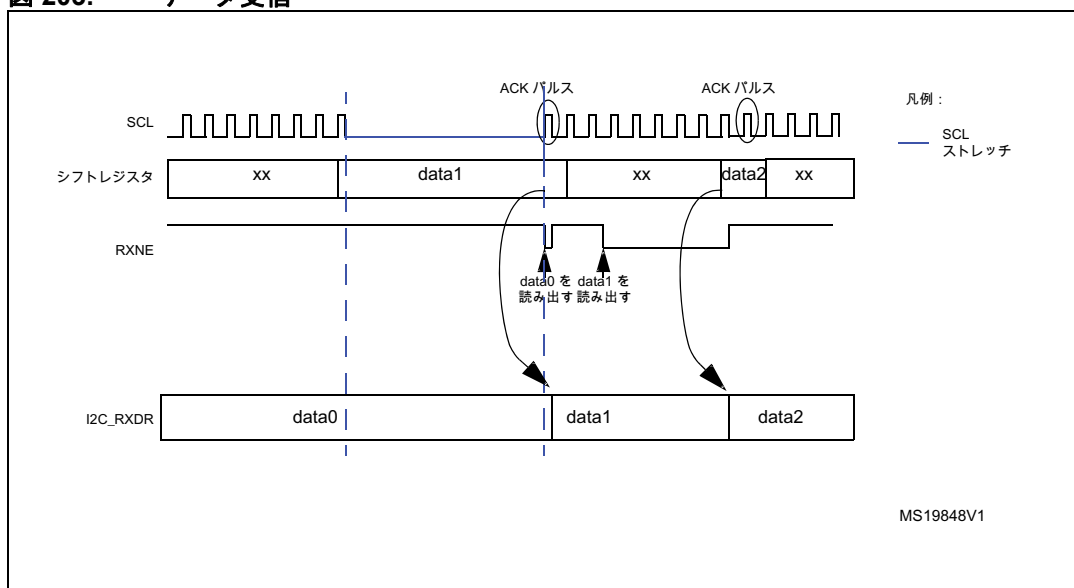
28.4.7 データ転送

データ転送は、送受信データレジスタとシフトレジスタを通じて管理されます。

受信

SDA 入力はシフトレジスタに送られます。8 番目の SCL パルスの後 (完全なデータバイトの受信後)、シフトレジスタは、I2Cx_RXDR レジスタが空の場合 (RXNE=0)、このレジスタにコピーされます。RXNE=1 の場合、すなわち、前に受信されたデータバイトがまだ読み出されていなかった場合、SCL ラインは I2Cx_RXDR が読み出されるまでストレッチされます。ストレッチは、8 番目と 9 番目の SCL パルスの間 (確認応答パルスの前) に挿入されます。

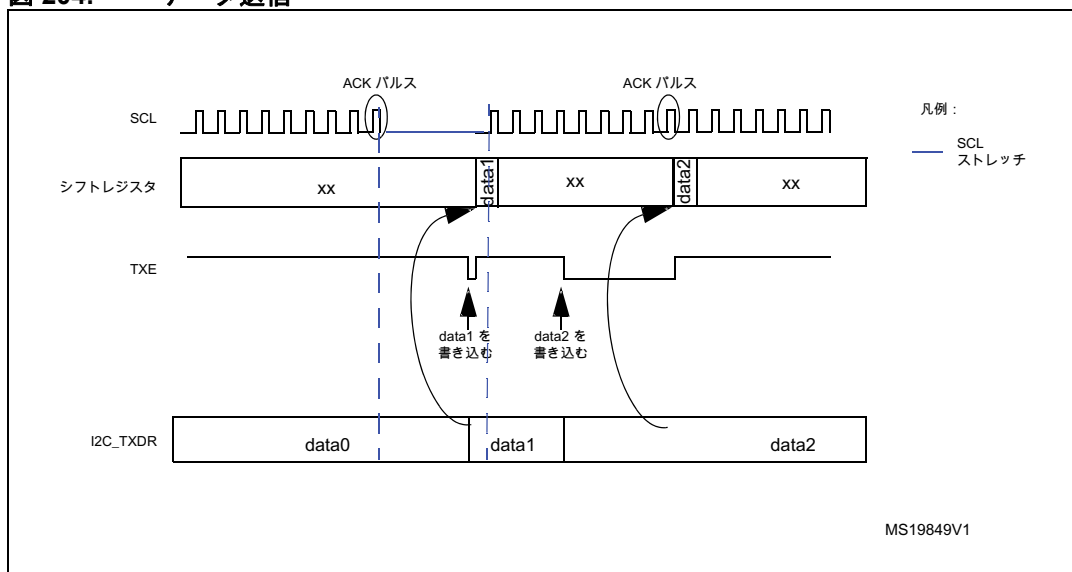
図 203. データ受信



送信

I2Cx_TXDR レジスタが空 (TXE=0) でない場合、その内容が 9 番目の SCL パルス (確認応答パルス) の後、シフトレジスタにコピーされます。次に、シフトレジスタの内容が SDA ラインにシフトアウトされます。TXE=1 の場合、すなわち、I2Cx_TXDR にデータがまだ書き込まれていない場合、SCL ラインは I2Cx_TXDR に書き込まれるまでストレッチされます。ストレッチは、9 番目の SCL パルスの後で行われます。

図 204. データ送信



ハードウェア転送管理

次のようにさまざまなモードでバイト転送を管理し、通信をクローズするために、I2C にはハードウェアにバイトカウンタが組み込まれています：

- マスタモードでの NACK、STOP、および ReSTART 生成
- スレーブレシーバモードでの ACK 制御
- SMBus 機能がサポートされているときの PEC 生成／確認

バイトカウンタは、マスタモードでは常に使用されます。デフォルトでは、スレーブモードでは無効ですが、I2Cx_CR2 レジスタの SBC (スレーブバイト制御) ビットをセットすることによって、ソフトウェアにより有効にできます。

転送されるバイト数は、I2Cx_CR2 レジスタの NBYTES[7:0] ビットフィールドでプログラムされます。転送バイト数 (NBYTES) が 255 より大きい場合、またはレシーバが受信データバイトの確認応答値を制御したい場合には、I2Cx_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。このモードでは、NBYTES でプログラムされたバイト数が転送されると、TCR フラグがセットされ、TCIE がセットされている場合は割り込みが生成されます。SCL は、TCR フラグがセットされている間、ストレッチされます。TCR は、NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

NBYTES カウンタに最後のバイト数が再ロードされたときには、RELOAD ビットがクリアされる必要があります。

マスタモードで RELOAD=0 のときには、カウンタは 2 つのモードで使用できます：

- **自動終了モード** (I2Cx_CR2 レジスタの AUTOEND = 1)。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、マスタは STOP コンディションを自動的に送信します。
- **ソフトウェア終了モード** (I2Cx_CR2 レジスタの AUTOEND = 0)。このモードでは、NBYTES[7:0] ビットフィールドでプログラムされたバイト数が転送されると、ソフトウェアアクションが求められます。TC フラグがセットされ、TCIE ビットがセットされている場合は割り込みが生成されます。SCL 信号は、TC フラグがセットされている間、ストレッチされます。TC フラグは、I2Cx_CR2 レジスタの START または STOP ビットがセットされたときに、ソフトウェアによってクリアされます。マスタが RESTART コンディションを送信したいときには、このモードを使用する必要があります。

注意 : AUTOEND ビットは、RELOAD ビットがセットされているときには効果がありません。

表 103. I2C 設定表

| 機能 | SBC ビット | RELOAD ビット | AUTOEND ビット |
|------------------------------|---------|------------|-------------|
| マスタ Tx/Rx NBYTES + STOP | x | 0 | 1 |
| マスタ Tx/Rx + NBYTES + RESTART | x | 0 | 0 |
| スレーブ Tx/Rx すべての受信バイトに ACK | 0 | x | x |
| スレーブ Rx および ACK 制御 | 1 | 1 | x |

28.4.8 I2C スレーブモード

I2C スレーブ初期化

スレーブモードで動作するには、少なくとも 1 つのスレーブアドレスを有効にする必要があります。2 つのレジスタ I2Cx_OAR1 と I2Cx_OAR2 を使用して、スレーブ専用アドレス OA1 および OA2 をプログラムできます。

- OA1 は、I2Cx_OAR1 レジスタの OA1MODE ビットをセットすることによって、7 ビットモード（デフォルト）または 10 ビットアドレッシングモードに設定できます。
OA1 を有効にするには、I2Cx_OAR1 レジスタの OA1EN ビットをセットします。
- 追加のスレーブアドレスが必要な場合は、2 番目のスレーブアドレス OA2 を設定できます。I2Cx_OAR2 レジスタの OA2MSK[2:0] ビットを設定することによって、最大 7 つの OA2 LSB をマスクできます。したがって、OA2MSK が 1 から 6 まです設定された場合、OA2[7:2]、OA2[7:3]、OA2[7:4]、OA2[7:5]、OA2[7:6]、または OA2[7] のみが受信アドレスと比較されます。OA2MSK が 0 に等しくなくなるとすぐに、OA2 のアドレスコンパレータは、確認応答されない I2C 予約済みアドレス（0000 XXX および 1111 XXX）を除外します。OA2MSK=7 の場合、受信されたすべてのアドレスが確認応答されます（予約済みアドレスを除く）。OA2 は常に 7 ビットアドレスです。
これらの予約済みアドレスは、特定のイネーブルビットによって有効化された場合、I2Cx_OAR1 または I2Cx_OAR2 レジスタが OA2MSK=0 でプログラムされた場合、確認応答できます。
OA2 を有効にするには、I2Cx_OAR2 レジスタの OA2EN ビットをセットします。
- 同報アドレスは、I2Cx_CR1 レジスタの GCEN ビットをセットすることで有効になります。

I2C が有効アドレスの 1 つによって選択されると、ADDR 割り込みステータスフラグがセットされ、ADDRIE ビットがセットされている場合は割り込みが生成されます。

デフォルトでは、スレーブはクロックストレッチ機能を使用し、必要なときには、ソフトウェアアクションを実行するために、SCL 信号をローレベルでストレッチすることを意味します。マスタがクロックストレッチをサポートしない場合、I2Cx_CR1 レジスタの NOSTRETCH=1 で I2C を設定する必要があります。

ADDR 割り込みの受信後、いくつかのアドレスが有効な場合は、I2Cx_ISR レジスタの ADDCODE[6:0] ビットを読み出して、一致するアドレスを確認する必要があります。転送方向を知るために、DIR フラグも確認する必要があります。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

スレーブクロックストレッチ (NOSTRETCH = 0)

デフォルトモードでは、I2C スレーブは次の状況で SCL クロックをストレッチします：

- ADDR フラグがセットされると：受信アドレスは有効なスレーブアドレスの 1 つと一致します。このストレッチは、ADDRCF ビットをセットすることによりソフトウェアによって ADDR フラグがクリアされたときにリリースされます。
- 送信時、前のデータ送信が完了し、新しいデータが I2Cx_TXDR レジスタに書き込まれなかった場合、または ADDR フラグがクリアされたときに (TXE=1)、最初のデータバイトが書き込まれていなかった場合。このストレッチは、データが I2Cx_TXDR レジスタに書き込まれたときにリリースされます。
- 受信時、I2Cx_RXDR レジスタがまだ読み出されていず、新しいデータ受信が完了したとき。このストレッチは、I2Cx_RXDR が読み出されたときにリリースされます。
- スレーブバイト制御モードおよび再ロードモード (SBC=1 および RELOAD=1) で TCR = 1 のとき、すなわち、最後データバイトが転送されたとき。このストレッチは、NBYTES[7:0] フィールドにゼロ以外の値を書き込むことによって TCR がクリアされたときにリリースされます。
- SCL 立ち下がりエッジの検出後、I2C は、 $[(SDADEL+SCLDEL+1) \times (PRESC+1) + 1] \times t_{I2CCLK}$ の間、SCL ローをストレッチします。

クロックストレッチなしのスレーブ (NOSTRETCH = 1)

I2Cx_CR1 レジスタの NOSTRETCH = 1 のとき、I2C スレーブは SCL 信号をストレッチしません。

- ADDR フラグがセットされている間、SCL クロックはストレッチされません。
- 送信時、転送に対応する最初の SCL パルスが発生する前に、I2Cx_TXDR レジスタにデータが書き込まれる必要があります。そうでない場合、アンダーランが発生し、I2Cx_ISR レジスタで OVR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。OVR フラグは、最初のデータ送信が開始し、STOPF ビットがまだセットされている（クリアされていない）ときにもセットされます。したがって、次の転送で送信される最初のデータを書き込んだ後でのみ、前に転送の STOPF フラグをクリアすることによって、送信される最初のデータについても、OVR ステータスが提供されることを確実にできます。
- 受信時、次のデータバイトの 9 番目の SCL パルス (ACK パルス) が発生する前に、I2Cx_RXDR レジスタからデータが読み出される必要があります。そうでない場合、オーバーランが発生し、I2Cx_ISR レジスタの OVR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

スレーブバイト制御モード

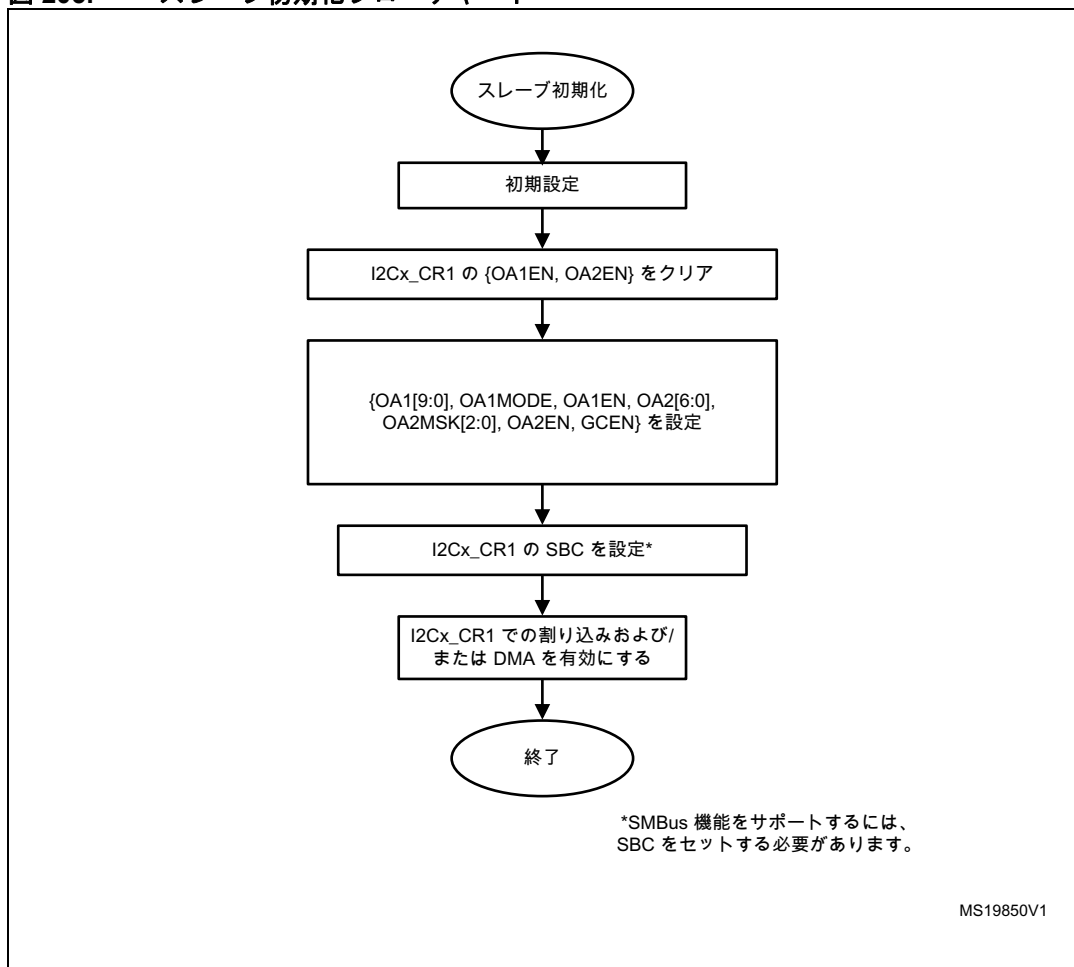
スレーブ受信モードでバイト ACK 制御を可能にするためには、I2Cx_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。これは、SMBus 標準に準拠する必要があります。

スレーブ受信モードでバイト ACK 制御を可能にするためには、再ロードモードを選択する必要があります (RELOAD=1)。各バイトの制御を得るには、ADDR 割り込みサブルーチンで NBYTES を 0x1 に初期化し、各受信バイト後に 0x1 に再ロードする必要があります。バイトが受信されると、TCR ビットがセットされ、8 番目と 9 番目の SCL パルスの間で、SCL 信号ローをストレッチします。I2Cx_RXDR レジスタからデータを読み出すことができ、その後、I2Cx_CR2 レジスタの ACK ビットを設定することによって、確認応答するかどうかを決定できます。SCL ストレッチは、NBYTES をゼロ以外の値にプログラムすることによってリリースされ、確認応答または非確認応答が送信され、次のバイトを受信できます。

NBYTES に 0x1 より大きい値をロードでき、この場合、受信フローは NBYTES データ受信、継続します。

- 注：** SBC ビットは、I2C が無効なとき、またはスレーブがアドレス指定されていないとき、または ADDR=1 のときに設定する必要があります。
- RELOAD ビットの値は、ADDR=1 のとき、または TCR=1 のときに変更できます。**
- 注意：** スレーブバイト制御モードは、NOSTRETCH モードと互換性がありません。NOSTRETCH=1 のときに SBC をセットすることはできません。

図 205. スレーブ初期化フローチャート



スレーブトランスミッタ

I2Cx_TXDR レジスタが空になると、送信割り込みステータス (TXIS) が生成されます。I2Cx_CR1 レジスタの TXIE ビットがセットされている場合は、割り込みが生成されます。

TXIS ビットは、I2Cx_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

NACK が受信されると、I2Cx_ISR レジスタの NACKF ビットがセットされ、I2Cx_CR1 レジスタの NACKIE ビットがセットされていた場合は割り込みが生成されます。マスタが STOP または RESTART コンディションを実行できるように、スレーブは SCL および SDA ラインを自動的にリリースします。TXIS ビットは、NACK 受信時にはセットされません。

STOP が受信され、I2Cx_CR1 レジスタの STOPIE ビットがセットされると、I2Cx_ISR レジスタの STOPF フラグがセットされ、割り込みが生成されます。ほとんどのアプリケーションでは、SBC は通常、0 にプログラムされます。この場合、スレーブアドレスが受信されたときに (ADDR=1)、TXE = 0 であった場合、I2Cx_TXDR レジスタの内容を最初のデータバイトとして送信するか、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2Cx_TXDR レジスタをフラッシュするかを選択できます。

スレーブバイト制御モード (SBC=1) では、送信バイト数をアドレス一致割り込みサブルーチンの NBYTES でプログラムする必要があります (ADDR=1)。この場合、転送中の TXIS イベントの数は、NBYTES でプログラムされた値に対応します。

注意：

NOSTRETCH=1 のとき、SCL クロックは ADDR フラグがセットされている間はストレッチされないため、最初のデータバイトをプログラムするために ADDR サブルーチンで I2Cx_TXDR レジスタの内容をフラッシュすることはできません。最初に送信されるデータバイトは、I2Cx_TXDR レジスタで前もってプログラムされている必要があります：

- このデータは、前の送信メッセージの最後の TXIS イベントで書き込まれたデータでもかまいません。
- このデータバイトが送信データバイトでない場合、新しいデータバイトをプログラムするために TXE ビットをセットすることによって I2Cx_TXDR レジスタをフラッシュできます。アドレスの確認応答に続いて、最初のデータ送信が開始する前にこれらが実行されることを保証するためには、STOPF ビットのクリアは、これらのアクションの後でのみ行う必要があります。

最初のデータ送信が開始したときに STOPF がまだセットされていた場合、アンダーランエラーが生成されます (OVR フラグがセットされます)。

TXIS イベントが必要な場合 (送信割り込みまたは送信 DMA リクエスト)、TXIS イベントを生成するためには、TXE ビットに加えて TXIS ビットもセットする必要があります。

図 206. I2C スレーブトランスミッタの転送シーケンスフローチャート (NOSTRETCH=0)

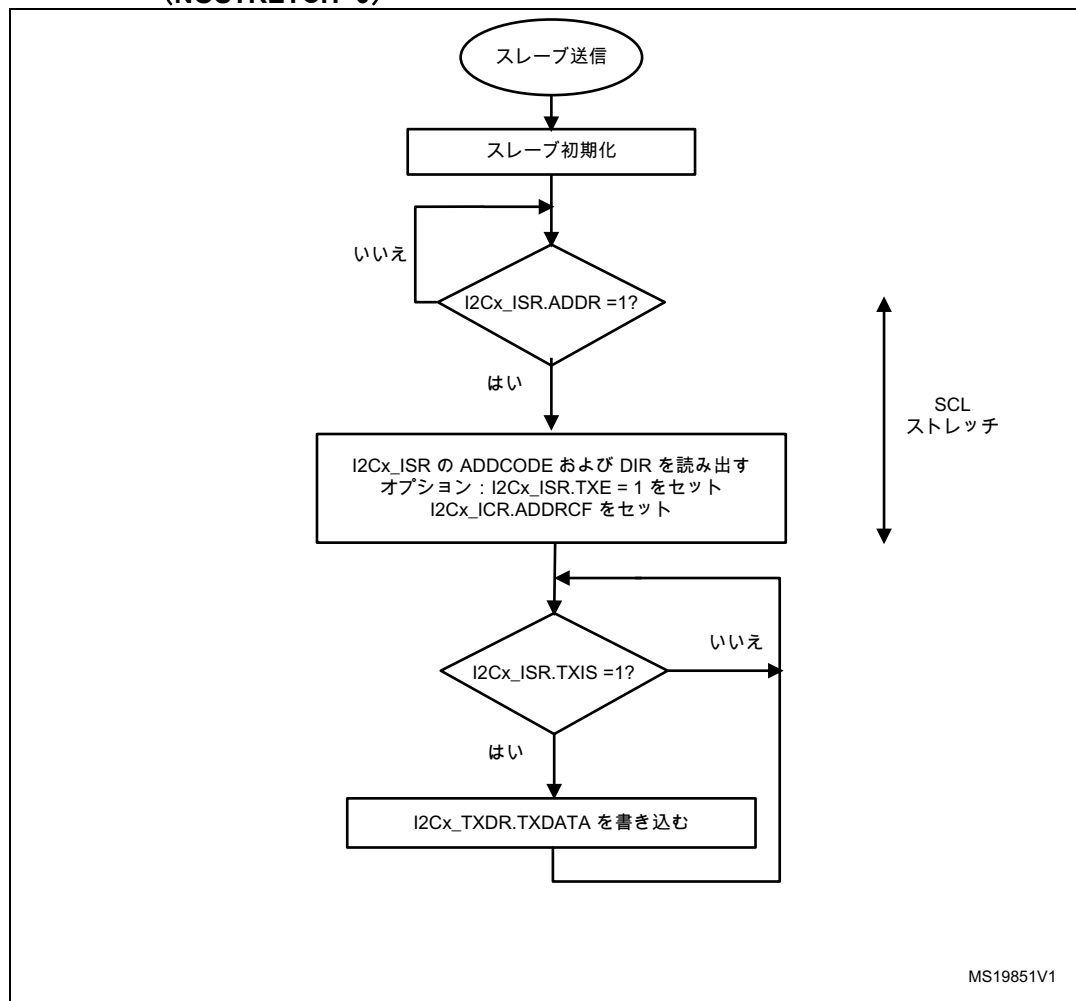


図 207. I2C スレーブトランスマッタの転送シーケンスフローチャート (NOSTRETCH=1)

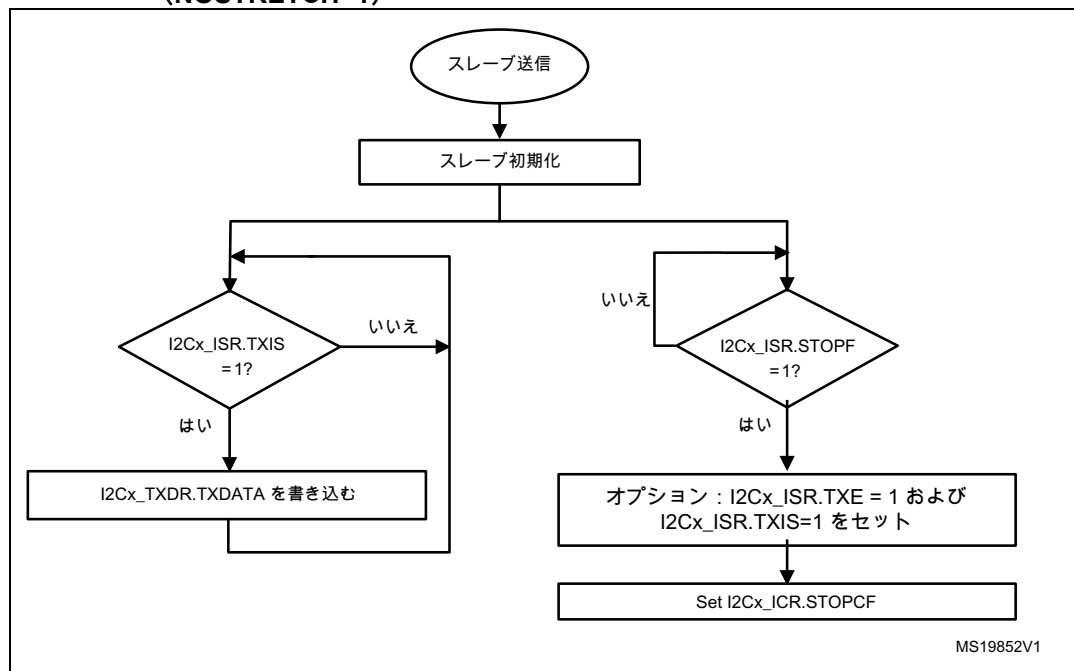
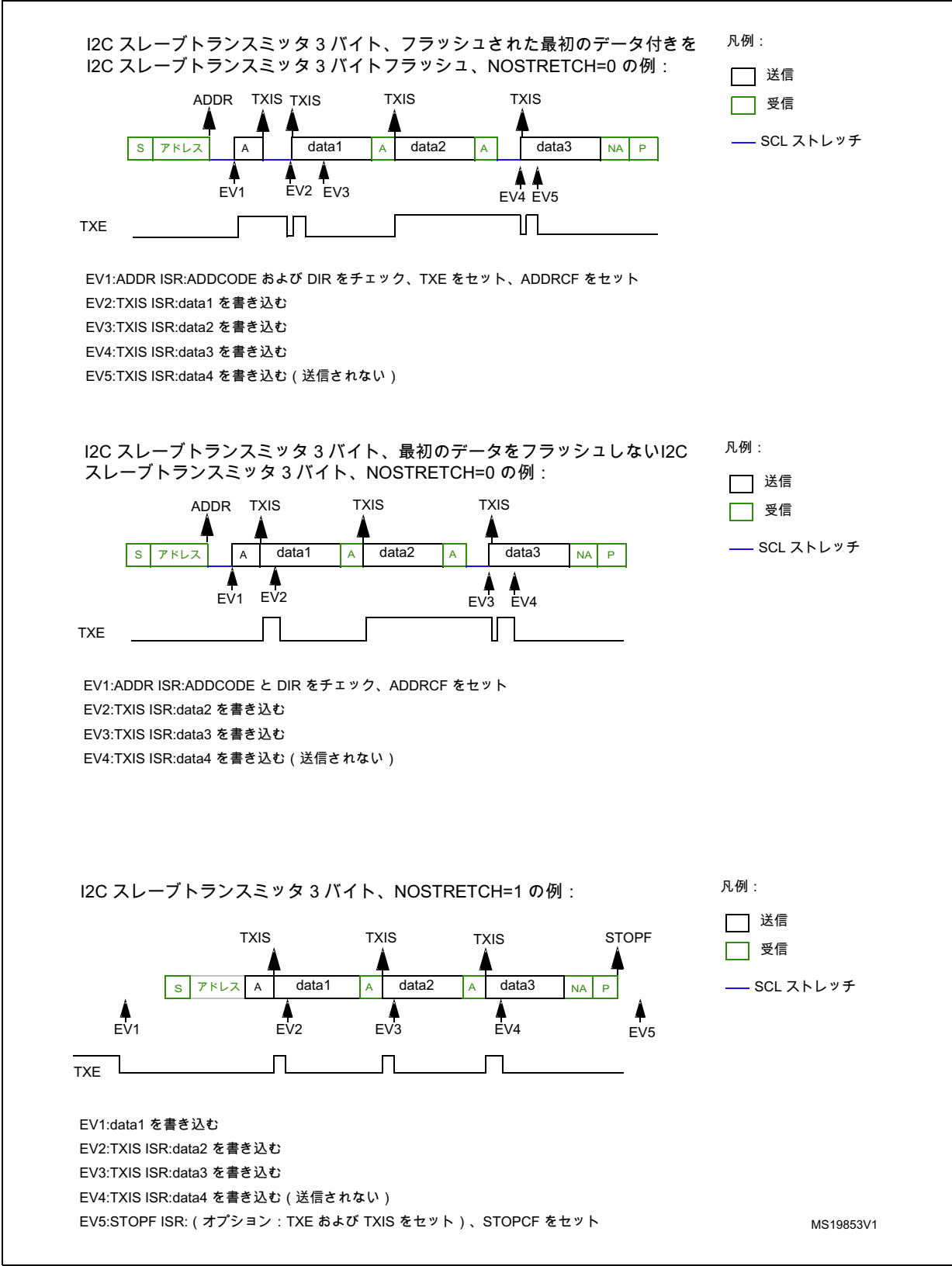


図 208. I2C スレーブトランスミッタの転送バス図



スレーブレシーバ

I2Cx_RXDR がフルのときには、I2Cx_ISR の RXNE がセットされ、I2Cx_CR1 の RXIE がセットされている場合は割り込みが生成されます。RXNE は、I2Cx_RXDR が読み出されたときにクリアされます。

STOP が受信され、I2Cx_CR1 レジスタの STOPIE ビットがセットされると、I2Cx_ISR の STOPF がセットされ、割り込みが生成されます。

図 209. スレーブレシーバの転送シーケンスフローチャート (NOSTRETCH=0)

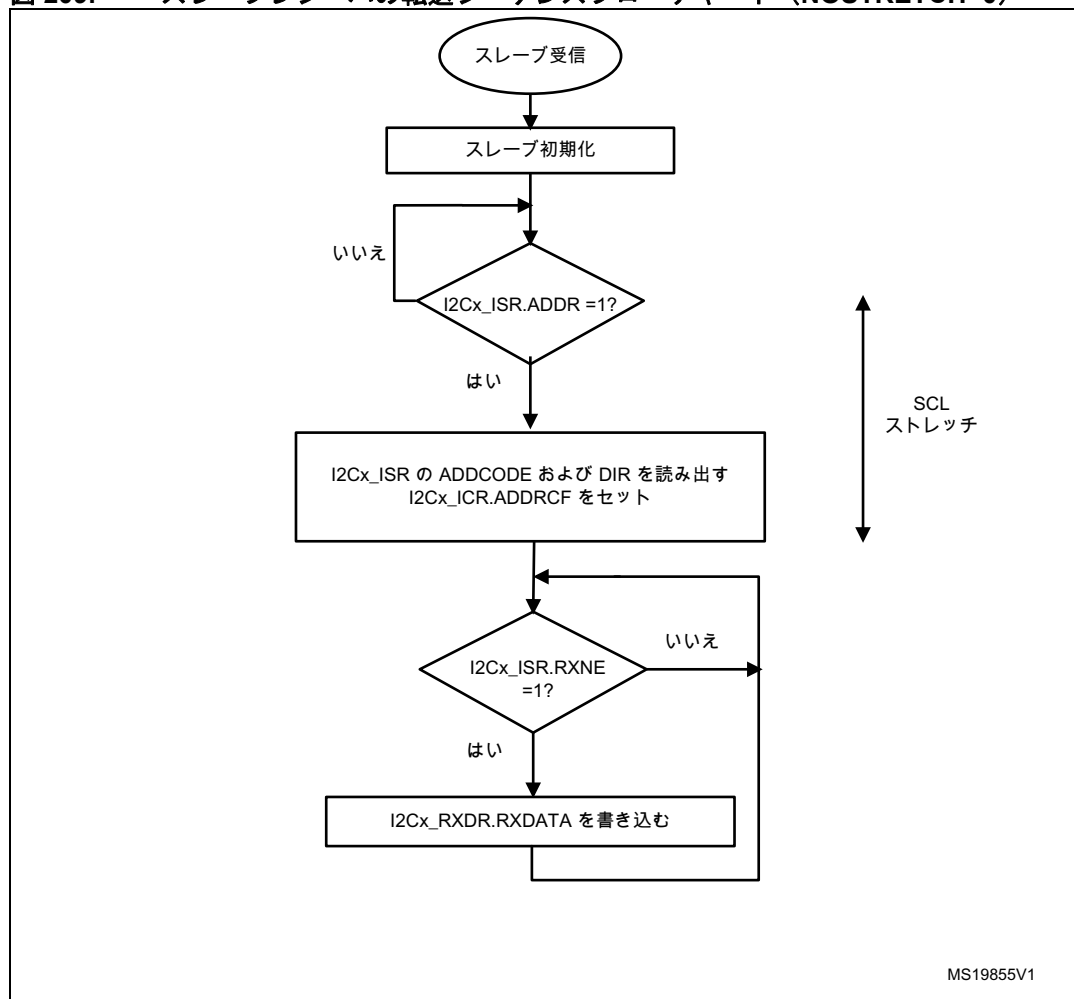


図 210. スレーブレシーバの転送シーケンスフローチャート (NOSTRETCH=1)

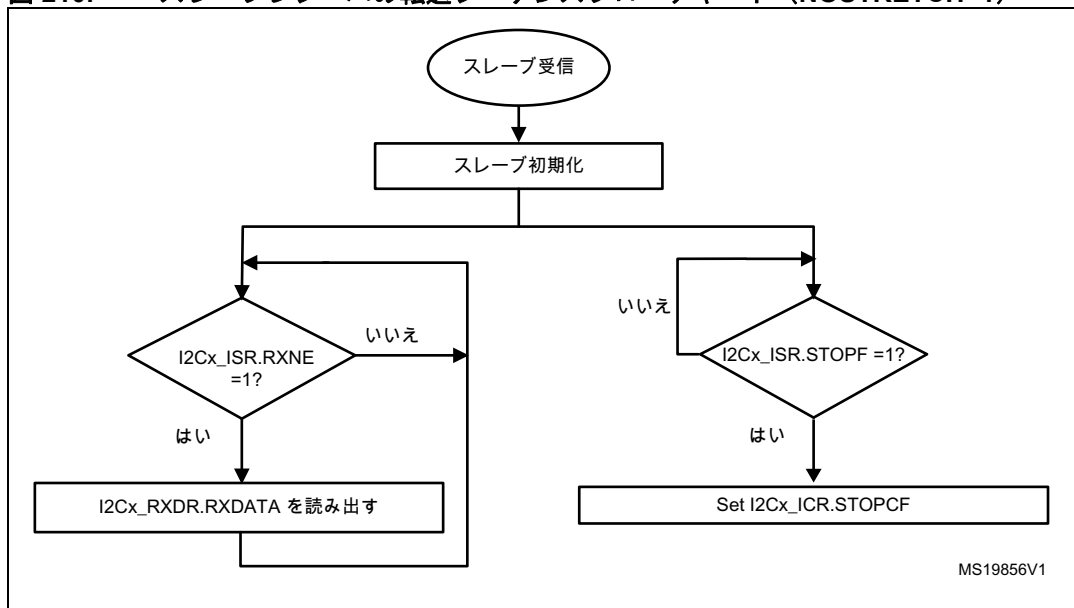
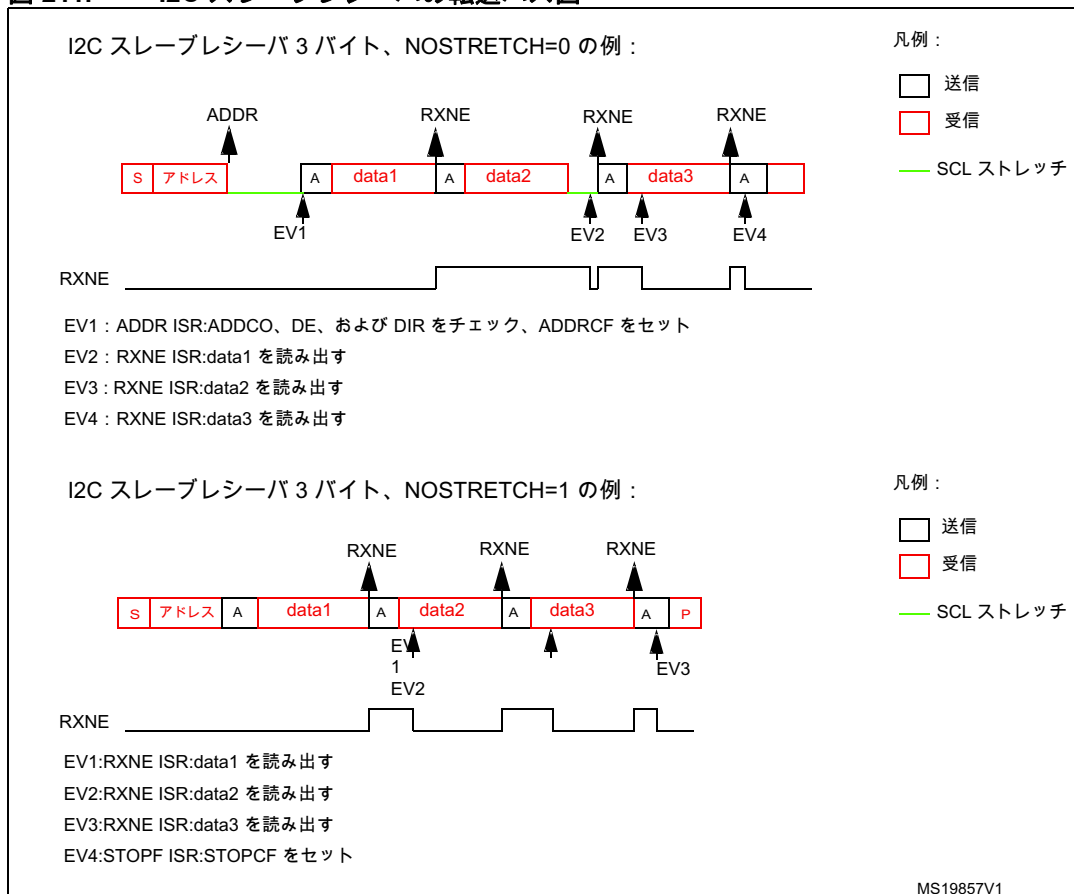


図 211. I2C スレーブレシーバの転送バス図



28.4.9 I2C マスタモード

I2C マスタ初期化

ペリフェラルを有効にする前に、I2Cx_TIMINGR レジスタの SCLH および SCLL ビットをセットすることによって、I2C マスタクロックを設定する必要があります。

マルチマスタ環境とスレーブクロックストレッチをサポートするために、クロック同期メカニズムが実装されています。

クロック同期を可能にするために：

- クロックのローレベルは SCLL カウンタを使用してカウントされ、SCL ローレベル内部検出から開始されます。
- クロックのハイレベルは SCLH カウンタを使用してカウントされ、SCL ハイレベル内部検出から開始されます。

I2C は、SCL 立ち下がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC1} の後に SCL ローレベルを検出します。SCLL カウンタが I2Cx_TIMINGR レジスタの SCLL[7:0] ビットでプログラムされた値に達すると、I2C は SCL をハイレベルにリリースします。

I2C は、SCL 立ち上がりエッジ、SCL 入力ノイズフィルタ（アナログ + デジタル）、および I2CxCLK クロックとの SCL 同期に応じた遅延 t_{SYNC2} の後に SCL ハイレベルを検出します。SCLH カウンタが I2Cx_TIMINGR レジスタの SCLH[7:0] ビットでプログラムされた値に達すると、I2C は SCL をローレベルにします。

結果として、マスタクロック周期は次のとおりです：

$$t_{\text{SCL}} = t_{\text{SYNC1}} + t_{\text{SYNC2}} + \{ [(SCLH+1) + (SCLL+1)] \times (\text{PRESC}+1) \times t_{\text{I2CCLK}} \}$$

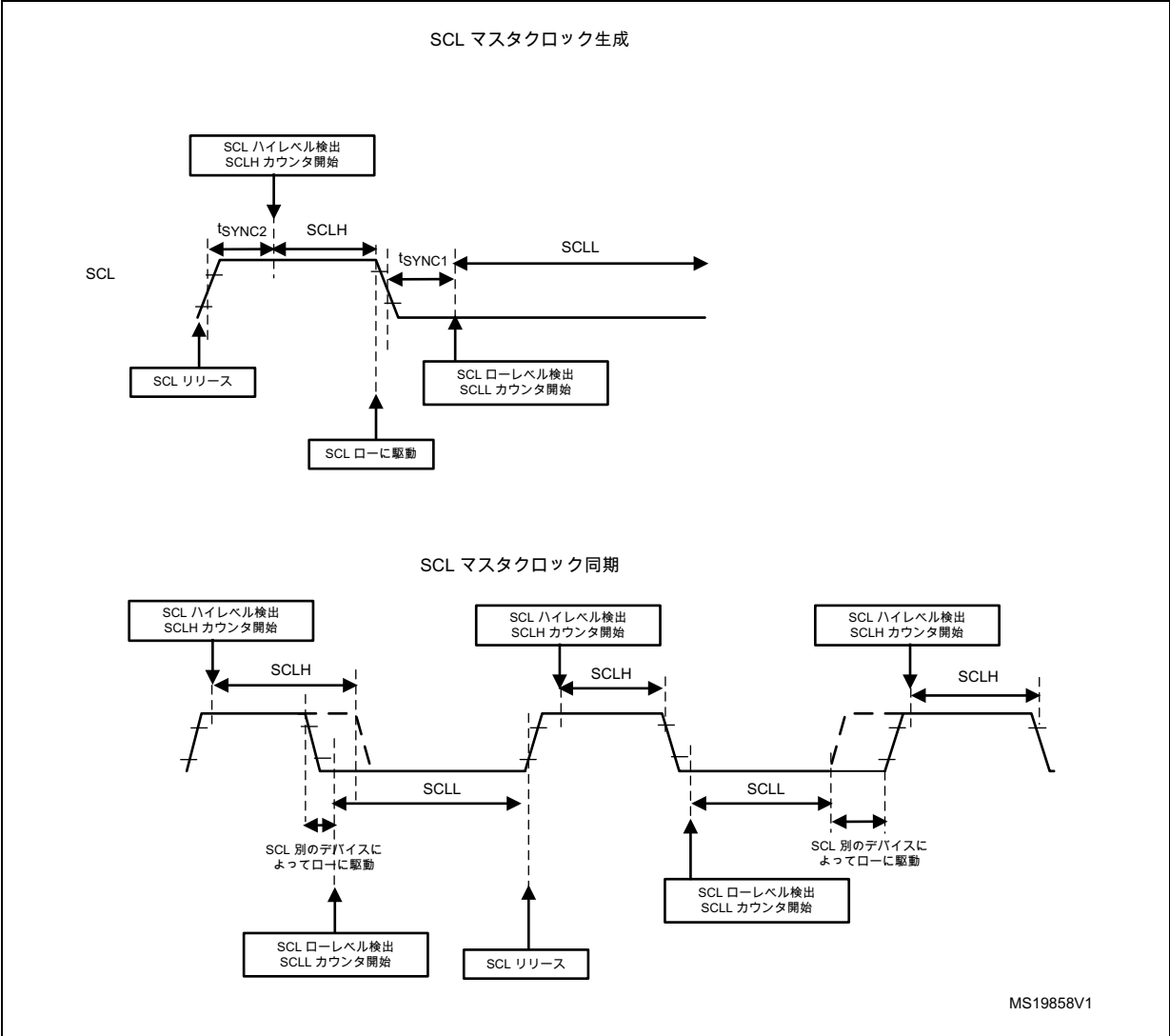
t_{SYNC1} の長さは、次のパラメータに依存します。

- SCL 立ち下がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF $\times t_{\text{I2CCLK}}$
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

t_{SYNC2} の長さは、次のパラメータに依存します：

- SCL 立ち上がり傾斜
- 有効な場合、アナログフィルタによる入力遅延。
- 有効な場合、デジタルフィルタによる入力遅延：DNF $\times t_{\text{I2CCLK}}$
- I2CCLK クロックとの SCL 同期による遅延（2 ～ 3 I2CCLK 周期）

図 212. マスタクロック生成



注意： I²C または SMBus 準拠のためには、マスタクロックは次のタイミングを満たす必要があります：

表 104. I2C-SMBUS 仕様のクロックタイミング

| 記号 | パラメータ | 標準モード | | 高速モード | | 高速モード プラス | | SMBUS | | 単位 |
|---------------------|-------------------------------|-------|-----|-------|-----|--------------|------|-------|-----|-----|
| | | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | |
| f _{SCL} | SCL クロック周波数 | | 100 | | 400 | | 1000 | | 100 | kHz |
| t _{HD:STA} | (反復) START コンディションの ホールド時間 | 4.0 | | 0.6 | | 0.26 | | 4.0 | | μs |
| t _{SU:STA} | 反復 START コンディションの セットアップ時間 | 4.7 | | 0.6 | | 0.26 | | 4.7 | | μs |
| t _{SU:STO} | STOP コンディションのセットアップ 時間 | 4.0 | | 0.6 | | 0.26 | | 4.0 | | μs |



表 104. I2C-SMBUS 仕様のクロックタイミング (続き)

| 記号 | パラメータ | 標準モード | | 高速モード | | 高速モード プラス | | SMBUS | | 単位 |
|------------|--|-------|------|-------|-----|--------------|-----|-------|------|---------|
| | | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | 最小値 | 最大値 | |
| t_{BUF} | STOP コンディションと START コンディションの間の バスフリー時間 | 4.7 | | 1.3 | | 0.5 | | 4.7 | | μs |
| t_{LOW} | SCL クロックのロー周期 | 4.7 | | 1.3 | | 0.5 | | 4.7 | | μs |
| t_{HIGH} | SCL クロックの周期 | 4.0 | | 0.6 | | 0.26 | | 4.0 | 50 | μs |
| t_r | SDA および SCL 信号の立ち上がり 時間 | | 1000 | | 300 | | 120 | | 1000 | ns |
| t_f | SDA および SCL 信号の立ち下がり 時間 | | 300 | | 300 | | 120 | | 300 | ns |

注 : **SCLL は、 t_{BUF} および $t_{SU:STA}$ タイミングの生成にも使用されます。**
SCLH は、 $t_{HD:STA}$ および $t_{SU:STO}$ タイミングの生成にも使用されます。
I2Cx_TIMINGR 設定と I2CCLK 周波数の例については、[セクション 28.4.10 : I2Cx_TIMINGR レジスタの設定例](#) を参照してください。

マスタ通信の初期化 (アドレスフェーズ)

通信を初期化するためには、I2Cx_CR2 レジスタでアドレス指定されたスレーブについて次のパラメータをプログラムする必要があります :

- アドレッシングモード (7 ビットまたは 10 ビット) : ADD10
- 送信されるスレーブアドレス : SADD[9:0]
- 転送方向 : RD_WRN
- 10 ビットアドレスが読み出される場合 : HEAD10R ビット。HEAD10R を設定して、完全なアドレスシーケンスが送信されなければならないか、ヘッダのみ (方向の変更の場合) かを示す必要があります。
- 転送されるバイト数 : NBYTES[7:0]。バイト数が 255 バイト以上の場合は、NBYTES[7:0] に 0xFF を書き込む必要があります。

次に、I2Cx_CR2 レジスタの START ビットをセットする必要があります。START ビットがセットされているとき、上記のすべてのビットを変更することはできません。

その場合、マスタは、バスがフリーである (BUSY = 0) ことを検出すると、 t_{BUF} の遅延後に、自動的に START コンディションとスレーブアドレスを送信します。

アービトレーション喪失の場合、マスタはスレーブモードに自動的に切り替えて、スレーブとしてアドレス指定された場合は専用アドレスを確認応答できます。

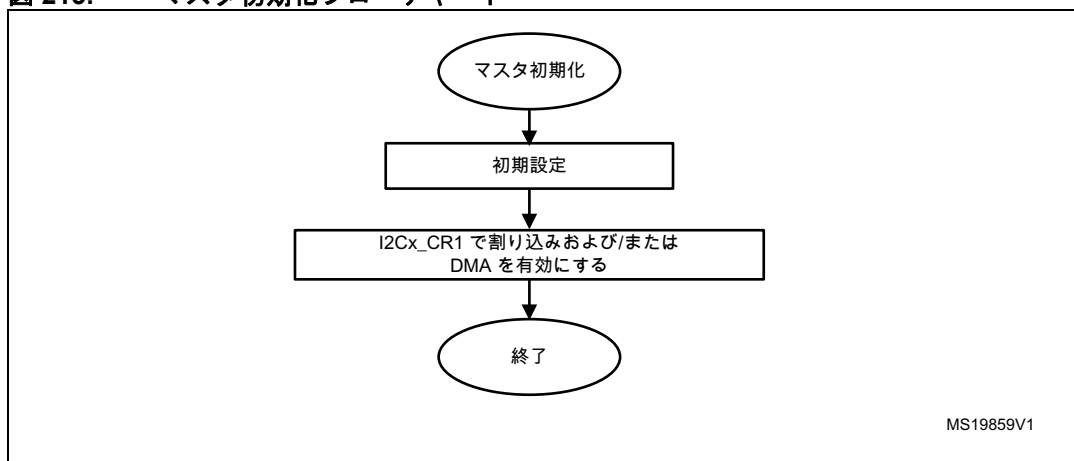
注 : **START ビットは、スレーブアドレスがバスに送信されたとき、受信した確認応答値にかかわらず、ハードウェアによってリセットされます。START ビットは、アービトレーション喪失が発生した場合にも、ハードウェアによってリセットされます。START ビットがセットされているときに、I2C がスレーブとしてアドレス指定された場合 (ADDR=1)、I2C はスレーブモードに切り替わり、ADDRCF ビットがセットされたときに START ビットがクリアされます。**

注 : **反復スタートコンディションにも同じ手順が適用されます。この場合、BUSY=1 です。**



この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

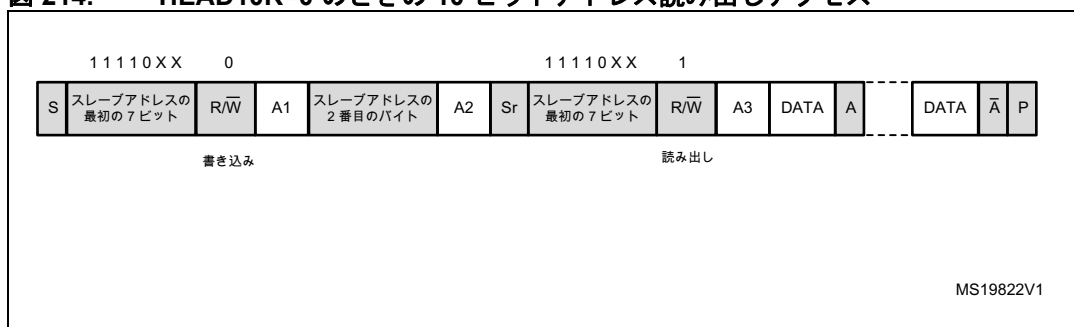
図 213. マスタ初期化フローチャート



10 ビットアドレススレーブをアドレス指定するマスタレシーバの初期化

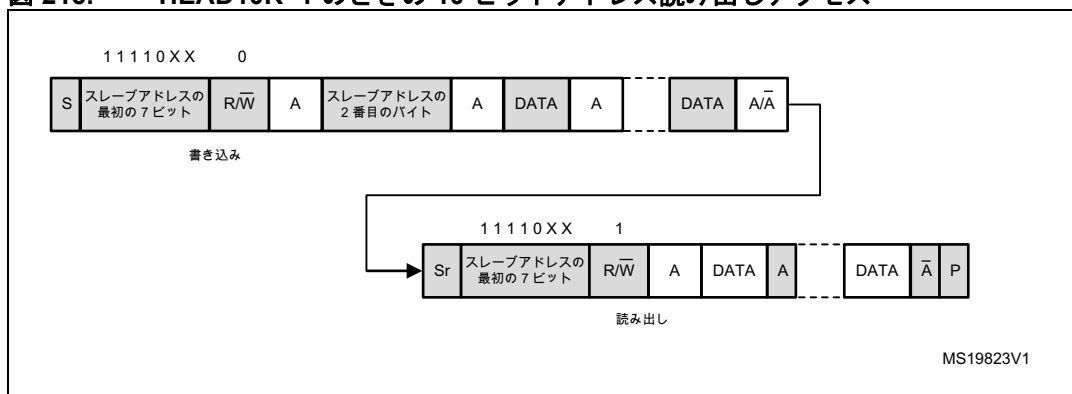
- スレーブアドレスが 10 ビット形式の場合、I2Cx_CR2 レジスタの HEAD10R ビットをクリアすることによって、完全な読み出しシーケンスを送信することができます。この場合、マスタは、START ビットがセットされた後、次のような完全なシーケンスを自動的に送信します：(Re) START + スレーブアドレス 10 ビットヘッダ書き込み + スレーブアドレスの 2 番目のバイト + RStart + スレーブアドレス 10 ビットヘッダ読み出し。

図 214. HEAD10R=0 のときの 10 ビットアドレス読み出しアクセス



- マスタが 10 ビットアドレススレーブをアドレス指定して、このスレーブアドレスにデータを送信した後、同じスレーブからデータを読み出す場合には、まず、マスタ送信フローが行われる必要があります。その場合、反復開始が、HEAD10R=1 で設定された 10 ビットスレーブアドレスでセットされます。この場合、マスタは次のシーケンスを送信します：ReStart + スレーブアドレス 10 ビットヘッダ読み出し

図 215. HEAD10R=1 のときの 10 ビットアドレス読み出しアクセス



マスタトランスマッタ

書き込み転送の場合、ACK が受信されたときの9 番目の SCL パルス後、各バイトの送信後にTXIS フラグがセットされます。

I2Cx_CR1 レジスタの TXIE ビットがセットされている場合、TXIS イベント時に割り込みが生成されます。このフラグは、I2Cx_TXDR レジスタに次に送信されるデータバイトが書き込まれると、クリアされます。

転送中の TXIS イベントの数は、NBYTES[7:0] でプログラムされた値に対応します。送信されるデータバイト数の合計が 255 より大きい場合、I2Cx_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

TXIS フラグは、NACK 受信時にはセットされません。

- RELOAD=0 でNBYTES データが転送されたとき：
 - － 自動終了モード (AUTOEND=1) では、STOP が自動的に送信されます。
 - － ソフトウェア終了モード (AUTOEND=0) では、TC フラグがセットされ、ソフトウェアアクションを実行するために SCL ラインがローでストレッチされます：

正しいスレーブアドレス設定と転送バイト数で I2Cx_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションがバスに送信されます。

I2Cx_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。
- NACK が受信された場合：TXIS フラグはセットされず、NACK 受信後、自動的に STOP コンディションが送信されます。I2Cx_ISR レジスタの NACKF フラグがセットされ、NACKIE ビットがセットされていた場合は割り込みが生成されます。

図 216. $N \leq 255$ バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート

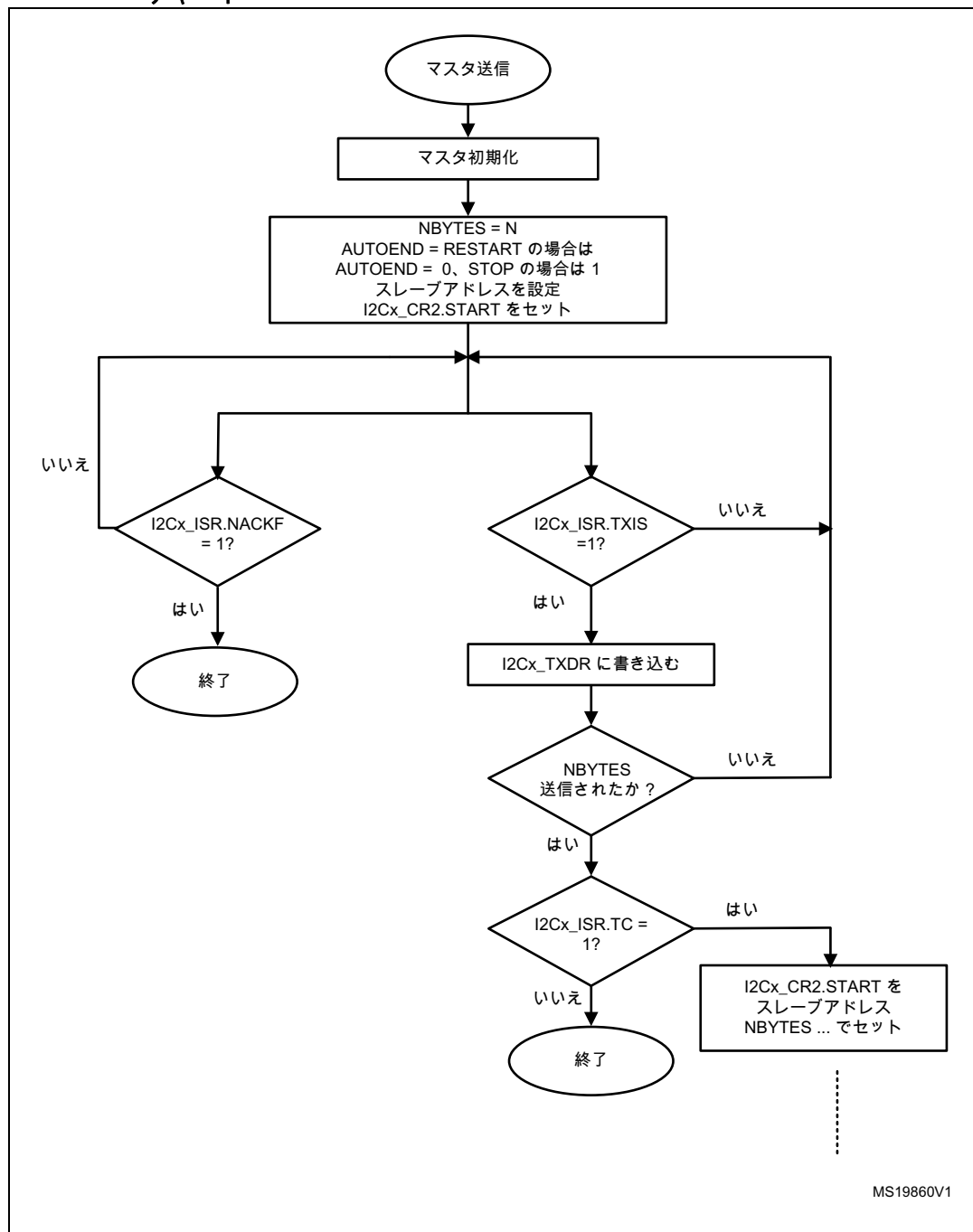


図 217. N>255 バイトの場合の I2C マスタトランスミッタの転送シーケンスフローチャート

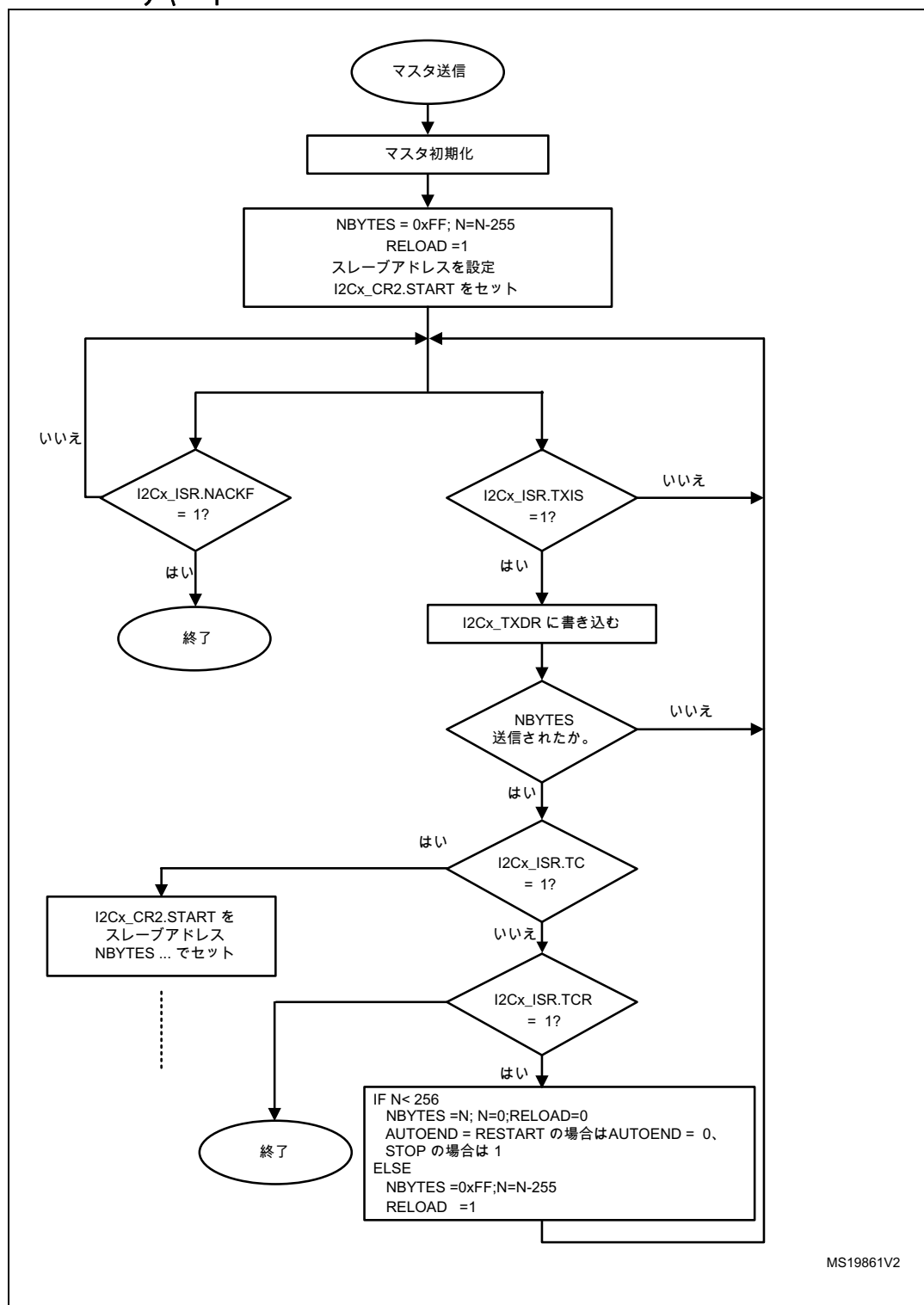
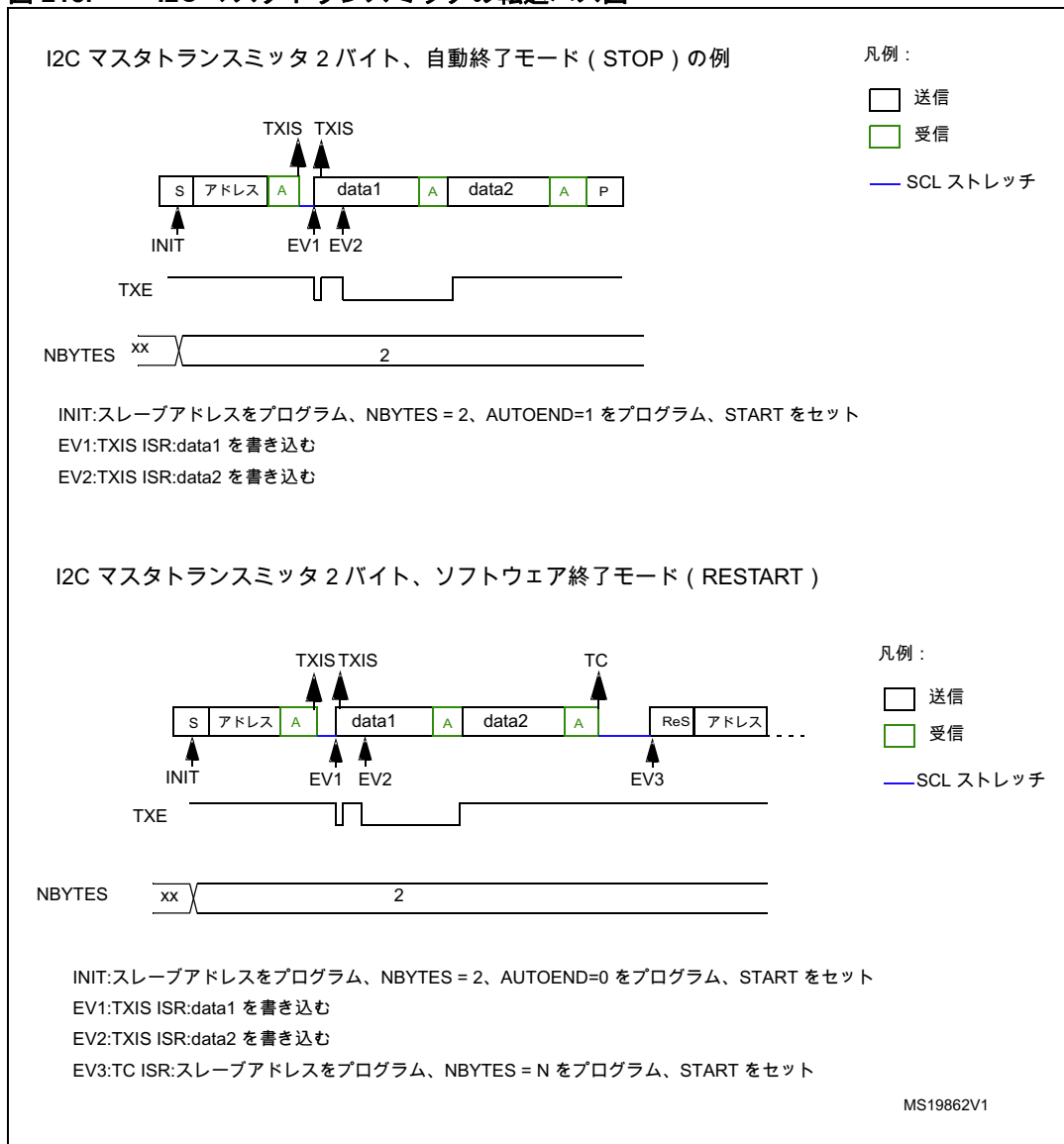


図 218. I2C マスタトランスミッタの転送バス図



マスタレシーバ

読み出し転送の場合、各バイトの受信後や8番目の SCL パルス後に RXNE フラグがセットされます。I2Cx_CR1 レジスタの RXIE ビットがセットされている場合、RXNE イベント時に割り込みが生成されます。このフラグは、I2Cx_RXDR が読み出されたときにクリアされます。

受信されるデータバイト数の合計が 255 より大きい場合、I2Cx_CR2 レジスタの RELOAD ビットをセットすることによって、再ロードモードを選択する必要があります。この場合、NBYTES[7:0] データが転送されると、TCR フラグがセットされ、NBYTES[7:0] にゼロ以外の値が書き込まれるまで、SCL ラインはローでストレッチされます。

- RELOAD=0 で NBYTES[7:0] データが転送されたとき：
 - － 自動終了モード (AUTOEND=1) では、最後の受信バイト後に NACK および STOP が自動的に送信されます。
 - － ソフトウェア終了モード (AUTOEND=0) では、最後の受信バイト後に NACK が自動的に送信され、TC フラグがセットされ、ソフトウェアアクションを実行できるように、SCL ラインがローでストレッチされます。

正しいスレーブアドレス設定と転送バイト数で I2Cx_CR2 レジスタの START ビットをセットすることによって、RESTART コンディションをリクエストできます。START ビットをセットすると、TC フラグがクリアされ、START コンディションとスレーブアドレスがバスに送信されます。

I2Cx_CR2 レジスタの STOP ビットをセットすることによって、STOP コンディションをリクエストできます。STOP ビットをセットすると、TC フラグがクリアされ、STOP コンディションがバスに送信されます。

図 219. $N \leq 255$ バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート

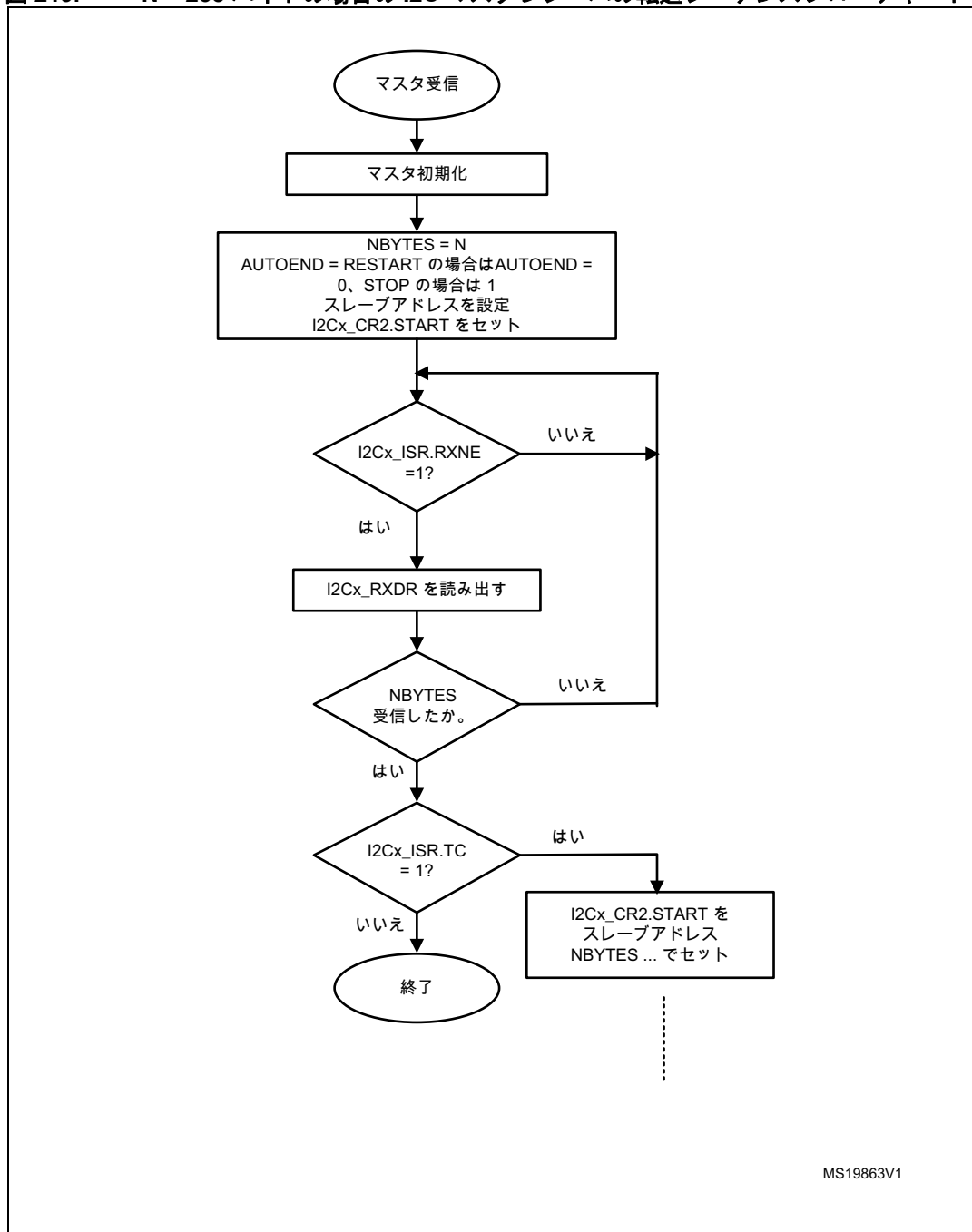


図 220. N>255 バイトの場合の I2C マスタレシーバの転送シーケンスフローチャート

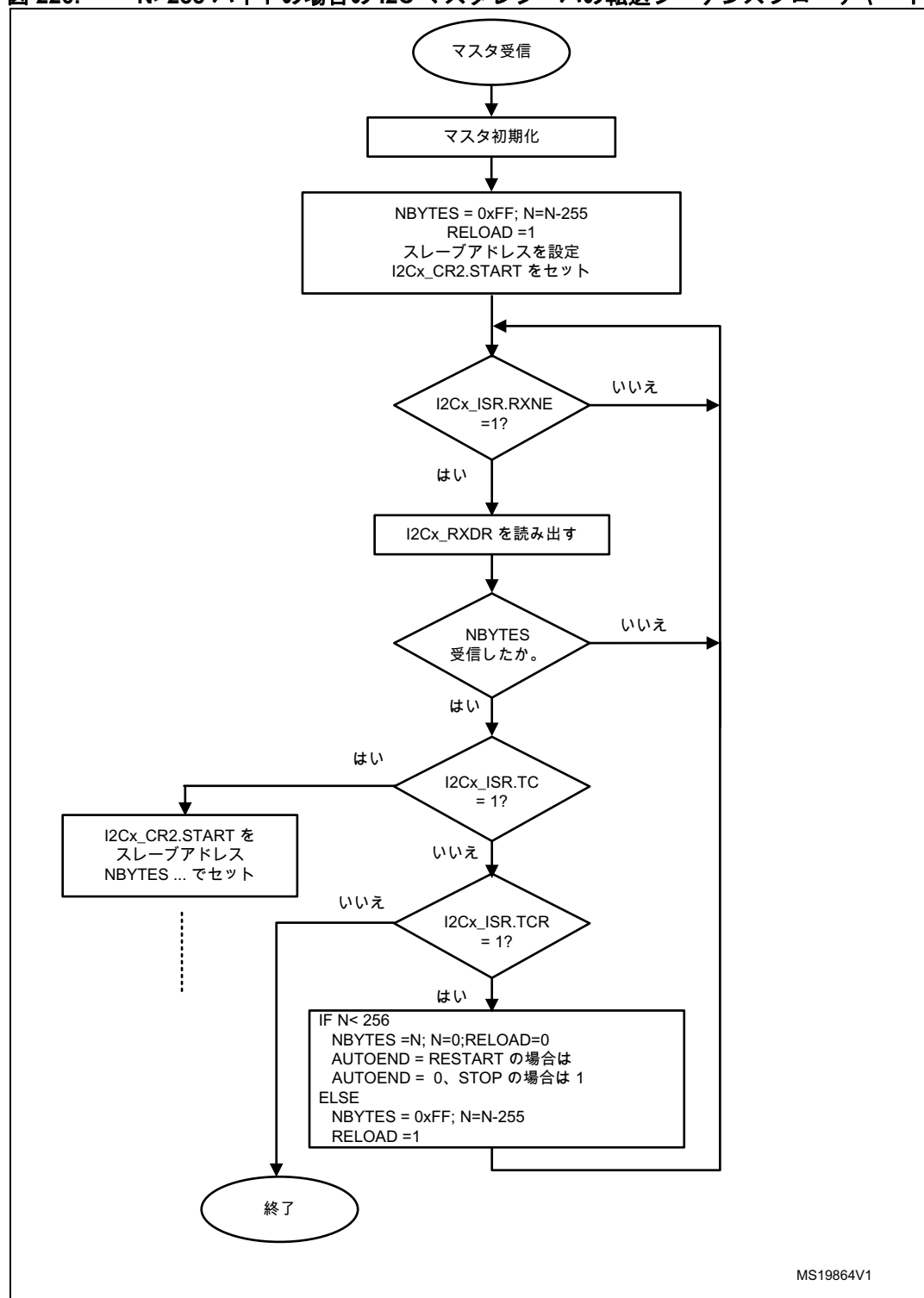
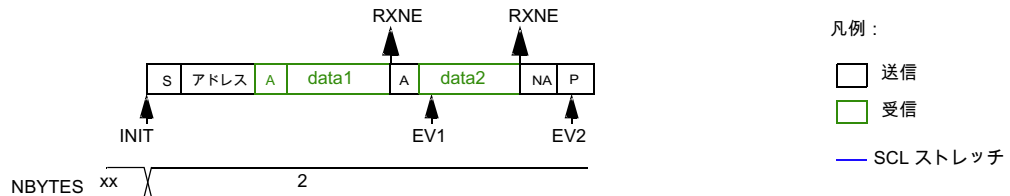


図 221. I2C マスタレシーバの転送バス図

I2C マスタレシーバ 2 バイト、自動終了モード (STOP) の例

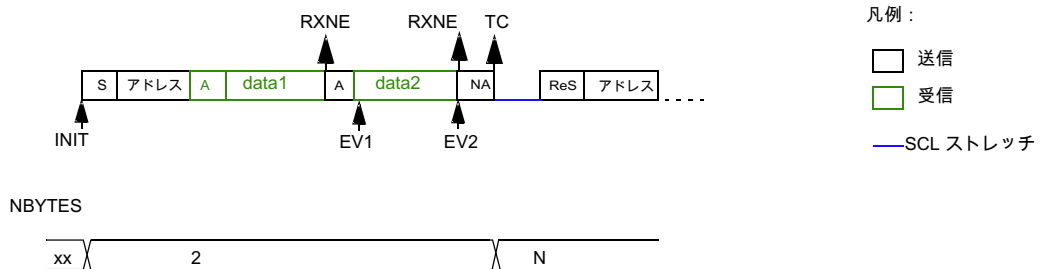


INIT:スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=1 をプログラム、START をセット

EV1:RXNE ISR:data1 を読み出す

EV2:RXNE ISR:data2 を読み出す

I2C マスタレシーバ 2 バイト、ソフトウェア終了モード (RESTART) の例



INIT:スレーブアドレスをプログラム、NBYTES = 2、AUTOEND=0 をプログラム、START をセット

EV1:RXNE ISR:data1 を読み出す

EV2:RXNE ISR:data2 を読み出す

EV3:TC ISR:スレーブアドレスをプログラム、NBYTES = N をプログラム、START をセット

MS19865V1

28.4.10 I2Cx_TIMINGR レジスタの設定例

下の表に、I²C 仕様 に準拠したタイミングを得るための I2Cx_TIMINGR をプログラムする方法の例を示します。より正確な設定値を得るには、アプリケーションノート AN4235 I²C タイミング設定ツールおよび該当するソフトウェア STSW-STM32126 を参照してください。

表 105. f_{I2CCLK} = 8 MHz でのタイミング設定の例

| パラメータ | 標準モード | | 高速モード | 高速モードプラス |
|---------------------------------|------------------------|-----------------------|-------------------------|-------------------------|
| | 10 kHz | 100 kHz | 400 kHz | 500 kHz |
| PRESC | 1 | 1 | 0 | 0 |
| SCLL | 0xC7 | 0x13 | 0x9 | 0x6 |
| t _{SCLL} | 200x250 ns = 50 μs | 20x250 ns = 5.0 μs | 10x125 ns = 1250 ns | 7x125 ns = 875 ns |
| SCLH | 0xC3 | 0xF | 0x3 | 0x3 |
| t _{SCLH} | 196x250 ns = 49 μs | 16x250 ns = 4.0 μs | 4x125ns = 500ns | 4x125 ns = 500 ns |
| t _{SCL} ⁽¹⁾ | ~100 μs ⁽²⁾ | ~10 μs ⁽²⁾ | ~2500 ns ⁽³⁾ | ~2000 ns ⁽⁴⁾ |
| SDADEL | 0x2 | 0x2 | 0x1 | 0x0 |
| t _{SDADEL} | 2x250 ns = 500 ns | 2x250 ns = 500 ns | 1x125 ns = 125 ns | 0 ns |
| SCLDEL | 0x4 | 0x4 | 0x3 | 0x1 |
| t _{SCLDEL} | 5x250 ns = 1250 ns | 5x250 ns = 1250 ns | 4x125 ns = 500 ns | 2x125 ns = 250 ns |

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 500 ns です。例：t_{SYNC1} + t_{SYNC2} = 655 ns

表 106. f_{I2CCLK} = 16 MHz でのタイミング設定の例

| パラメータ | 標準モード | | 高速モード | 高速モードプラス |
|---------------------------------|------------------------|-----------------------|-------------------------|-------------------------|
| | 10 kHz | 100 kHz | 400 kHz | 1000 kHz |
| PRESC | 3 | 3 | 1 | 0 |
| SCLL | 0xC7 | 0x13 | 0x9 | 0x4 |
| t _{SCLL} | 200 x 250 ns = 50 μs | 20 x 250 ns = 5.0 μs | 10 x 125 ns = 1250 ns | 5 x 62.5 ns = 312.5 ns |
| SCLH | 0xC3 | 0xF | 0x3 | 0x2 |
| t _{SCLH} | 196 x 250 ns = 49 μs | 16 x 250 ns = 4.0 μs | 4 x 125ns = 500 ns | 3 x 62.5 ns = 187.5 ns |
| t _{SCL} ⁽¹⁾ | ~100 μs ⁽²⁾ | ~10 μs ⁽²⁾ | ~2500 ns ⁽³⁾ | ~1000 ns ⁽⁴⁾ |
| SDADEL | 0x2 | 0x2 | 0x2 | 0x0 |
| t _{SDADEL} | 2 x 250 ns = 500 ns | 2 x 250 ns = 500 ns | 2 x 125 ns = 250 ns | 0 ns |
| SCLDEL | 0x4 | 0x4 | 0x3 | 0x2 |
| t _{SCLDEL} | 5 x 250 ns = 1250 ns | 5 x 250 ns = 1250 ns | 4 x 125 ns = 500 ns | 3 x 62.5 ns = 187.5 ns |

1. SCL 周期 t_{SCL} は、SCL 内部検出遅延のため、t_{SCLL} + t_{SCLH} より大きくなります。t_{SCL} として示されている値は例にすぎません。
2. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 1000 ns
3. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 750 ns
4. t_{SYNC1} + t_{SYNC2} 最小値は、4 × t_{I2CCLK} = 250 ns です。例：t_{SYNC1} + t_{SYNC2} = 500 ns

28.4.11 SMBus 固有の機能

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

概要

システム管理バス (SMBus) は、さまざまなデバイスが互いに通信したり、残りのシステム部分と通信したりできる 2 線インタフェースです。I²C の動作原理に基づきます。SMBus により、システムおよびパワーマネジメント関連のタスク向けの制御バスが実現できます。

このペリフェラルは、SMBUS 仕様 rev 2.0 (<http://smbus.org>) と互換性があります。

システム管理バス仕様では、3 種類のデバイスを規定しています。

- スレーブとは、コマンドを受信したり、コマンドに応答したりするデバイスです。
- マスタとは、コマンドを発行し、クロックを生成し、転送を終了させるデバイスです。
- ホストとは、システムの CPU にメインインタフェースを提供する特殊なマスタです。ホストは、マスタ/スレーブとすることができ、SMBus ホスト通知プロトコルをサポートする必要があります。システム内では、ただ 1 つのホストが許容されます。

このペリフェラルは、マスタまたはスレーブデバイスとして、また、ホストとしても設定できます。

SMBUS は、I²C 仕様 rev 2.1 に基づきます。

バスプロトコル

特定のデバイスについて、11 の可能なコマンドプロトコルがあります。デバイスは、11 のプロトコルの一部または全部を使用して通信できます。プロトコルは、Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write、および Block Write-Block Read Process Call です。これらのプロトコルは、ユーザのソフトウェアによって実装してください。

これらのプロトコルの詳細については、SMBus 仕様 ver. 2.0 (<http://smbus.org>) を参照してください。

アドレス解決プロトコル (ARP)

SMBus スレーブアドレスの競合は、各スレーブデバイスに新しいユニークなアドレスを動的に割り当てることによって解決できます。アドレス割り当てを目的とする各デバイスを分離する仕組みを提供するために、各デバイスは一意デバイス識別子 (UDID) を実装する必要があります。128 ビットの番号がソフトウェアによって実装されます。

このペリフェラルは、アドレス解決プロトコル (ARP) をサポートします。SMBus デバイスのデフォルトアドレス (0b1100 001) は、I2Cx_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。ARP コマンドは、ユーザのソフトウェアによって実装してください。

ARP サポートのために、スレーブモードでアービトレーションも行われます。

SMBus アドレス解決プロトコルの詳細については、SMBus 仕様 ver. 2.0 (<http://smbus.org>) を参照してください。

受信コマンドおよびデータ確認応答制御

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2Cx_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、セクション [スレーブバイト制御モード \(632 ページ\)](#) を参照してください。

Host Notify プロトコル

このペリフェラルは、I2Cx_CR1 レジスタの SMBHEN ビットをセットすることによって、Host Notify (ホスト通知) プロトコルをサポートします。この場合、ホストは SMBus ホストアドレス (0b0001 000) を確認応答します。

このプロトコルが使用されると、デバイスはマスタとして動作し、ホストはスレーブとして動作します。

SMBus アラート

SMBus ALERT オプション信号がサポートされます。スレーブ専用デバイスは、通信したいホストの SMBALERT# ピンを通じてホストに信号を送信します。ホストは、割り込みを処理し、アラート応答アドレス (0b0001 100) を通じて全 SMBALERT# デバイスに同時にアクセスします。SMBALERT# をローに引き下げたデバイスのみが、アラート応答アドレスを確認応答します。

スレーブデバイスとして設定されたとき (SMBHEN=0)、I2Cx_CR1 レジスタの ALERTEN ビットをセットすることによって、SMBA ピンはローに引き下げられます。同時に、アラート応答アドレスが有効になります。

ホストとして設定されたとき (SMBHEN=1)、SMBA ピンで立ち下がりエッジが検出され、ALERTEN=1 のとき、I2Cx_ISR レジスタの ALERT フラグがセットされます。I2Cx_CR1 レジスタの ERRIE ビットがセットされている場合は、割り込みが生成されます。ALERTEN=0 のときには、外部 SMBA ピンがローの場合でも、ALERT ラインはハイとみなされます。

SMBus ALERT ピンが不要な場合には、ALERTEN=0 の場合、SMBA ピンを標準 GPIO として使用できます。

パケットエラーチェック

信頼性と通信の堅牢性を向上させるために、SMBus 仕様にパケットエラーチェックメカニズムが導入されました。パケットエラーチェックは、各メッセージ転送の終わりにパケットエラーコード (PEC) を付加することによって実装されます。PEC は、すべてのメッセージバイト (アドレスと読み出し/書き込みビットを含む) に対して $C(x) = x_8 + x^2 + x + 1$ CRC-8 多項式を使用して計算されます。

ペリフェラルはハードウェア PEC 計算機が組み込まれ、受信バイトがハードウェアによって計算された PEC に一致しないときには自動的に非確認応答を送信できます。

タイムアウト

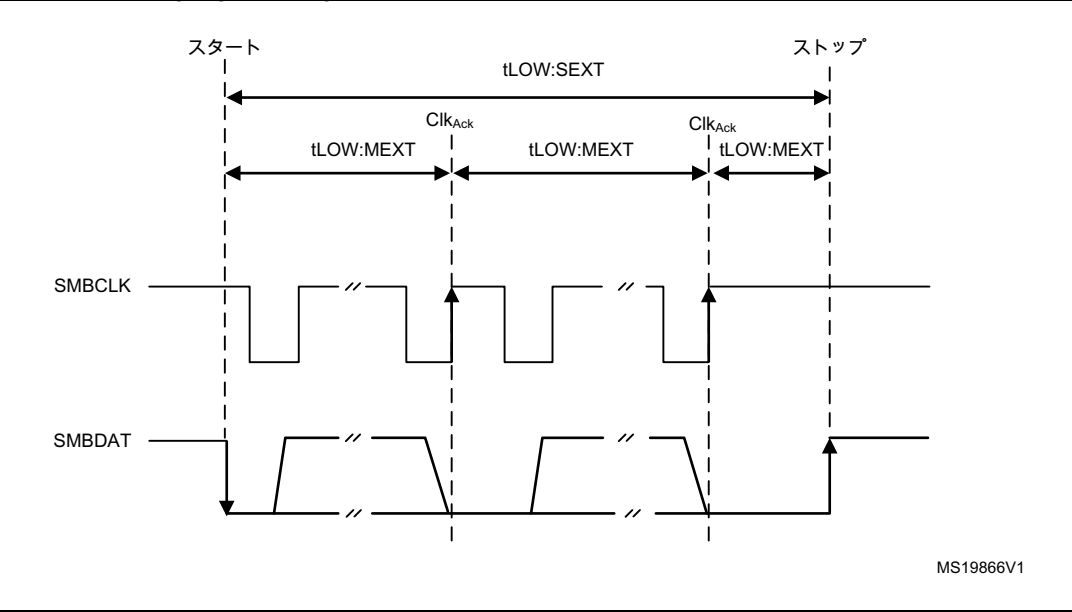
このペリフェラルは、SMBus 仕様 ver. 2.0 で定義された 3 つのタイムアウトに準拠するために、ハードウェアタイマが組み込まれています。

表 107. SMBus タイムアウト仕様

| 記号 | パラメータ | リミット | | 単位 |
|-----------------------------|-------------------------|------|-----|----|
| | | 最小値 | 最大値 | |
| t_{TIMEOUT} | クロックロータイムアウト検出 | 25 | 35 | ms |
| $t_{\text{LOW:SEXT}}^{(1)}$ | 累積クロックロー延長時間 (スレーブデバイス) | | 25 | ms |
| $t_{\text{LOW:MEXT}}^{(2)}$ | 累積クロックロー延長時間 (マスタデバイス) | | 10 | ms |

- $t_{\text{LOW:SEXT}}$ は、特定のスレーブデバイスが初めのSTART から STOP までの1つのメッセージのクロックサイクルを延長できる累積時間です。別のスレーブデバイスまたはマスタもクロックを延長して、合計のクロックロー延長時間が $t_{\text{LOW:SEXT}}$ より大きくなる場合があります。したがって、このパラメータは、スレーブデバイスをフルスピードのマスタの単独のターゲットとして測定されます。
- $t_{\text{LOW:MEXT}}$ は、マスタがメッセージの各バイト内のクロックサイクルを START-to-ACK、ACK-to-ACK、または ACK-to-STOP から定義に従って延長できる累積時間です。スレーブデバイスまたは別のマスタもクロックを延長して、合計のクロックロー時間が $t_{\text{LOW:MEXT}}$ より大きくなる場合があります。したがって、このパラメータは、フルスピードスレーブデバイスをマスタの単独のターゲットとして測定されます。

図 222. $t_{\text{LOW:SEXT}}$ 、 $t_{\text{LOW:MEXT}}$ のタイムアウト間隔



バスアイドル検出

マスタは、クロックおよびデータ信号が $t_{\text{HIGH,MAX}}$ より大きい t_{IDLE} の間ハイであった場合、バスはフリーであるとみなすことができます (表 104: I2C-SMBUS 仕様のクロックタイミングを参照してください)

このタイミングパラメータは、マスタがバスに動的に追加し、SMBCLK または SMBDAT ラインで状態遷移を検出できなかった可能性のあるコンディションをカバーします。この場合、マスタは十分に長い時間待って、転送が進行中でないことを確認する必要があります。このペリフェラルは、ハードウェアバスアイドル検出をサポートします。

28.4.12 SMBus 初期化

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

SMBus 通信を行うためには、I2C 初期化に加えて、他にも特定の初期化を行う必要があります：

受信コマンドおよびデータ確認応答制御（スレーブモード）

SMBus レシーバは、受信した各コマンドまたはデータを NACK できなければなりません。スレーブモードで ACK 制御を可能にするためには、I2Cx_CR1 レジスタの SBC ビットをセットすることによって、スレーブバイト制御モードを有効にする必要があります。詳細については、 [スレーブバイト制御モード \(632 ページ\)](#) を参照してください。

特定アドレス（スレーブモード）

必要な場合は、特定の SMBus アドレスを有効にしてください。詳細については、 [バスアイドル検出 \(655 ページ\)](#) を参照してください。

- SMBus デバイスのデフォルトアドレス (0b1100 001) は、I2Cx_CR1 レジスタの SMBDEN ビットをセットすることによって有効になります。
- SMBus ホストアドレス (0b0001 000) は、I2Cx_CR1 レジスタの SMBHEN ビットをセットすることによって有効になります。
- アラート応答アドレス (0b0001100) は、I2Cx_CR1 レジスタの ALERTEN ビットをセットすることによって有効になります。

パケットエラーチェック

PEC 計算を有効にするには、I2C_CR1 レジスタの PECEN ビットをセットします。その場合、PEC 転送はハードウェアバイトカウンタ (I2Cx_CR2 レジスタの NBYTES[7:0]) を使用して管理されます。PECEN ビットは、I2C を有効にする前に設定する必要があります。

PEC 転送はハードウェアバイトカウンタによって管理されるので、スレーブモードで SMBus とインタフェースするときには SBC ビットをセットする必要があります。PEC は、PECBYTE ビットがセットされ、RELOAD ビットがクリアされたとき、NBYTES-1 データの転送後に転送されます。RELOAD がセットされた場合、PECBYTE は効果がありません。

注意： I2C が有効なときには、PECEN 設定の変更はできません。

表 108. SMBUS の PEC 設定表

| モード | SBC ビット | RELOAD ビット | AUTOEND ビット | PECBYTE ビット |
|----------------------------------|------------|---------------|----------------|----------------|
| マスタ Tx/Rx NBYTES + PEC+ STOP | x | 0 | 1 | 1 |
| マスタ Tx/Rx NBYTES + PEC + ReSTART | x | 0 | 0 | 1 |
| スレーブ Tx/Rx と PEC | 1 | 0 | x | 1 |



タイムアウト検出

タイムアウト検出は、I2Cx_TIMEOUTR レジスタの TIMOUTEN および TEXTEN ビットをセットすることによって有効になります。SMBus 仕様 ver. 2.0 で指定された最大時間の前にタイムアウトを検出するようにタイマをプログラムする必要があります。

- t_{TIMEOUT} チェック

t_{TIMEOUT} チェックを有効にするためには、12 ビットの TIMEOUTA[11:0] ビットを t_{TIMEOUT} パラメータをチェックするためにタイマ再ロード値でプログラムする必要があります。SCL ローレベルタイムアウトを検出するためには、TIDLE ビットを 0 に設定する必要があります。

その場合、タイマは、I2Cx_TIMEOUTR レジスタの TIMOUTEN をセットすることによって有効になります。

SCL が $(\text{TIMEOUTA}+1) \times 2048 \times t_{\text{I2CCLK}}$ より長い時間、ローに設定された場合、I2Cx_ISR レジスタの TIMEOUT フラグがセットされます。

[表 109: さまざまな I2CCLK 周波数での TIMEOUTA の設定例 \(最大値 \$t_{\text{TIMEOUT}} = 25 \text{ ms}\$ \)](#) を参照してください。

注意： TIMOUTEN ビットがセットされているときには、TIMEOUTA[11:0] ビットおよび TIDLE ビットの設定変更はできません。

- $t_{\text{LOW:SEXT}}$ および $t_{\text{LOW:MEXT}}$ チェック

ペリフェラルがマスタとして設定されているか、スレーブとして設定されているかに応じて、12 ビットの TIMEOUTB タイマは、スレーブの場合は $t_{\text{LOW:SEXT}}$ をチェックするために、マスタの場合は $t_{\text{LOW:MEXT}}$ をチェックするために、設定する必要があります。標準では最大値のみが規定されているので、両方について同じ値を選ぶことができます。

その場合、タイマは、I2Cx_TIMEOUTR レジスタの TEXTEN ビットをセットすることによって有効になります。

SMBus ペリフェラルが、 $(\text{TIMEOUTB}+1) \times 2048 \times t_{\text{I2CCLK}}$ より長い時間および [バスアイドル検出 \(655 ページ\)](#) セクションで述べられているタイムアウト間隔で、累積 SCL ストレッチを実行した場合、I2Cx_ISR レジスタの TIMEOUT フラグがセットされます。

[表 110: さまざまな I2CCLK 周波数での TIMEOUTB の設定例](#) を参照してください。

注意： TEXTEN ビットがセットされているときには、TIMEOUTB 設定変更はできません。

バスアイドル検出

t_{IDLE} チェックを有効にするためには、12 ビットの TIMEOUTA[11:0] フィールドを t_{IDLE} パラメータを得るためにタイマ再ロード値でプログラムする必要があります。SCL および SDA ハイレベルタイムアウトを検出するためには、TIDLE ビットを 1 に設定する必要があります。

その場合、タイマは、I2Cx_TIMEOUTR レジスタの TIMOUTEN ビットをセットすることによって有効になります。

SCL および SDA の両方のラインが $(\text{TIMEOUTA}+1) \times 4 \times t_{\text{I2CCLK}}$ より長い間ハイのままであった場合、I2Cx_ISR レジスタの TIMEOUT フラグがセットされます。

[表 111: さまざまな I2CCLK 周波数での TIMEOUTA の設定例 \(最大値 \$t_{\text{IDLE}} = 50 \text{ ms}\$ \)](#) を参照してください。

注意： TIMOUTEN がセットされているときに、TIMEOUTA および TIDLE 設定を変更することはできません。

28.4.13 SMBus : I2Cx_TIMEOUTR レジスタの設定例

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

- t_{TIMEOUT} の最大時間を 25 ms に設定 :

表 109. さまざまな I2CCLK 周波数での TIMEOUTA の設定例
(最大値 $t_{\text{TIMEOUT}} = 25 \text{ ms}$)

| f_{I2CCLK} | TIMEOUTA[11:0] ビット | TIDLE ビット | TIMEOUTEN ビット | t_{TIMEOUT} |
|---------------------|-----------------------|--------------|------------------|---|
| 8 MHz | 0x61 | 0 | 1 | $98 \times 2048 \times 125 \text{ ns} = 25 \text{ ms}$ |
| 16 MHz | 0xC3 | 0 | 1 | $196 \times 2048 \times 62.5 \text{ ns} = 25 \text{ ms}$ |
| 32 MHz | 0x186 | 0 | 1 | $391 \times 2048 \times 31.25 \text{ ns} = 25 \text{ ms}$ |

- $t_{\text{LOW:SEXT}}$ および $t_{\text{LOW:MEXT}}$ の最大時間を 8 ms に設定 :

表 110. さまざまな I2CCLK 周波数での TIMEOUTB の設定例

| f_{I2CCLK} | TIMEOUTB[11:0] ビット | TEXTEN ビット | $t_{\text{LOW:EXT}}$ |
|---------------------|-----------------------|------------|--|
| 8 MHz | 0x1F | 1 | $32 \times 2048 \times 125 \text{ ns} = 8 \text{ ms}$ |
| 16 MHz | 0x3F | 1 | $64 \times 2048 \times 62.5 \text{ ns} = 8 \text{ ms}$ |
| 32 MHz | 0x7C | 1 | $125 \times 2048 \times 31.25 \text{ ns} = 8 \text{ ms}$ |

- t_{IDLE} の最大時間を 50 μs に設定

表 111. さまざまな I2CCLK 周波数での TIMEOUTA の設定例
(最大値 $t_{\text{IDLE}} = 50 \text{ ms}$)

| f_{I2CCLK} | TIMEOUTA[11:0] ビット | TIDLE ビット | TIMEOUTEN ビット | t_{TIDLE} |
|---------------------|-----------------------|-----------|------------------|---|
| 8 MHz | 0x63 | 1 | 1 | $100 \times 4 \times 125 \text{ ns} = 50 \mu\text{s}$ |
| 16 MHz | 0xC7 | 1 | 1 | $200 \times 4 \times 62.5 \text{ ns} = 50 \mu\text{s}$ |
| 32 MHz | 0x18F | 1 | 1 | $400 \times 4 \times 31.25 \text{ ns} = 50 \mu\text{s}$ |

28.4.14 SMBus スレーブモード

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

I2C スレーブ転送管理 ([セクション 28.4.8 : I2C スレーブモード](#) を参照) に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus スレーブトランスミッタ

IP が SMBus で使用されるときには、SBCは、プログラムされたデータバイト数の終わりの PEC 送信を可能にするため、1 にプログラムする必要があります。PECBYTE ビットがセットされているときには、NBYTES[7:0] でプログラムされたバイト数には PEC 送信が含まれます。その場合、TXIS 割り込みの合計数は NBYTES-1 であり、NBYTES-1 データ転送後にマスタが追加のバイトをリクエストした場合、I2Cx_PECR レジスタの内容が自動的に送信されます。

注意： PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。

図 223. N バイト + PEC の場合の SMBus スレーブトランスミッタの転送シーケンスフローチャート

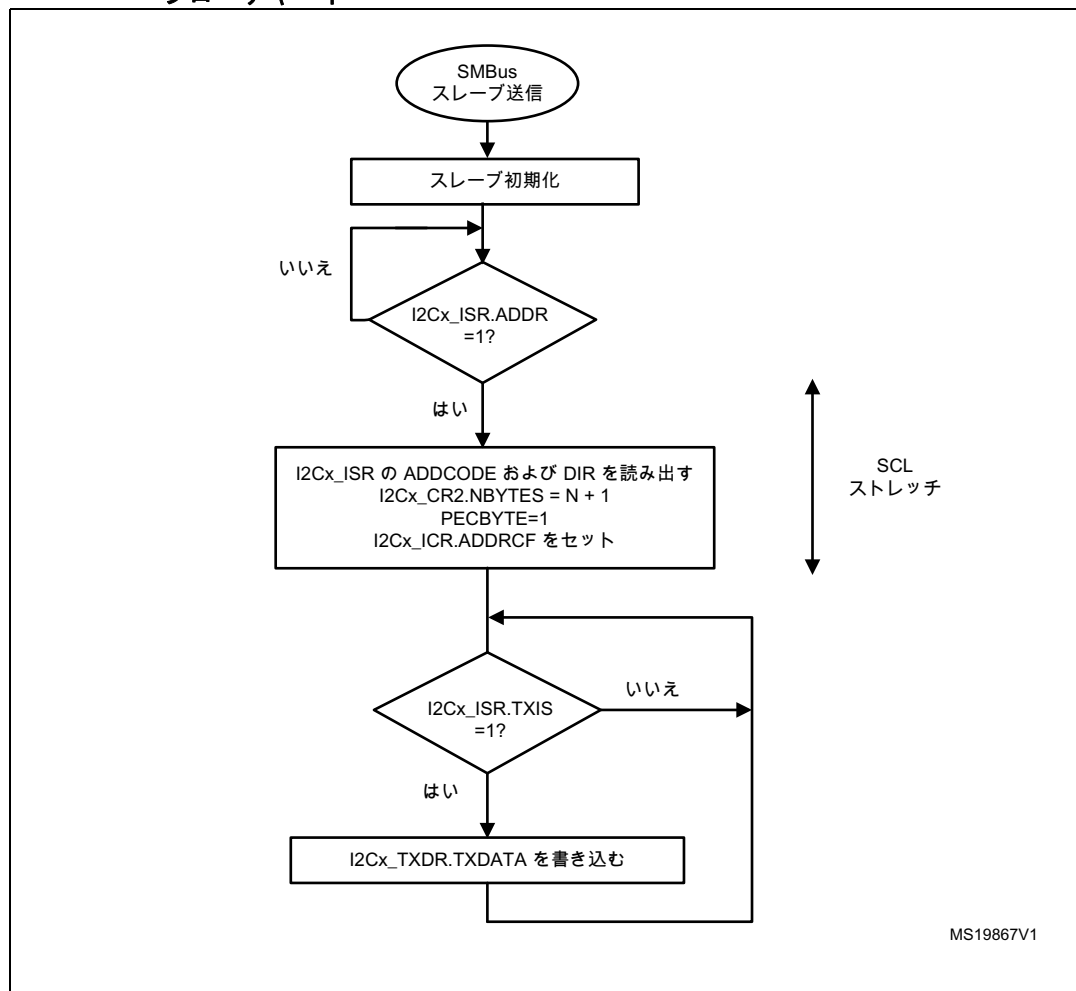
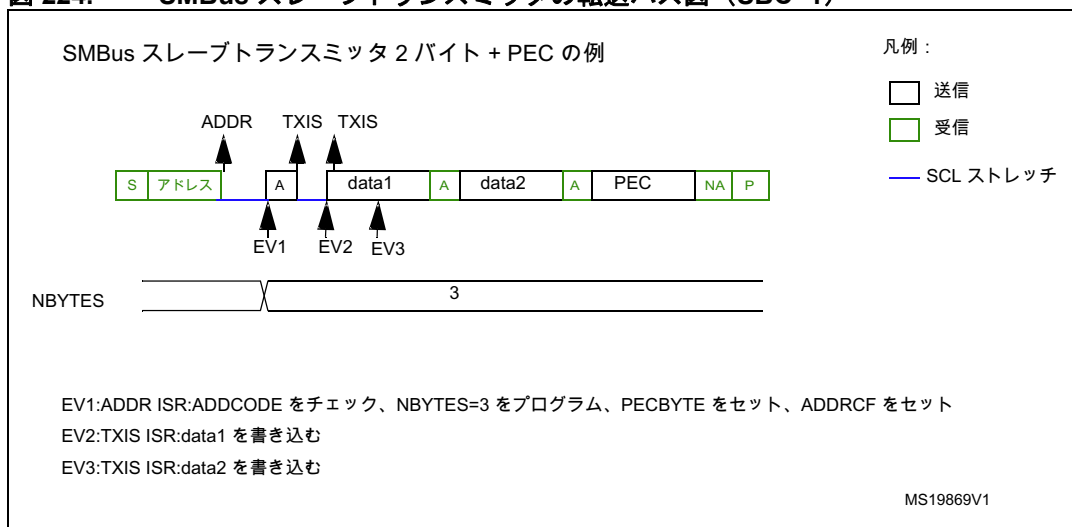


図 224. SMBus スレーブトランスミッタの転送バス図 (SBC=1)



SMBus スレーブレシーバ

I2C が SMBus モードで使用されるときには、SBC は、プログラムされたデータバイト数の終わりの PEC チェックを可能にするため、1 にプログラムする必要があります。各バイトの ACK 制御を可能にするためには、再ロードモードを選択する必要があります (RELOAD=1)。詳細については、[スレーブバイト制御モード \(632 ページ\)](#) を参照してください。

PEC バイトをチェックするためには、RELOAD ビットをクリアして、PECBYTE ビットをセットする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが内部 I2Cx_PECR レジスタの内容と比較されます。ACK ビットの値にかかわらず、比較が一致しなかった場合は NACK が自動的に生成され、比較が一致した場合は ACK が自動的に生成されます。PEC バイトが受信されると、他のデータと同様に I2Cx_RXDR レジスタにコピーされ、RXNE フラグがセットされます。

PEC 不一致の場合、PECERR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

ACK ソフトウェア制御が不要な場合は、PECBYTE=1 をプログラムし、同じ書き込み操作で NBYTES を連続フローで受信するバイト数にプログラムします。NBYTES-1 が受信された後、次の受信バイトが PEC であるかどうかチェックされます。

注意： PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。

図 225. N バイト + PEC の場合の SMBus スレーブレシーバの転送シーケンスフローチャート

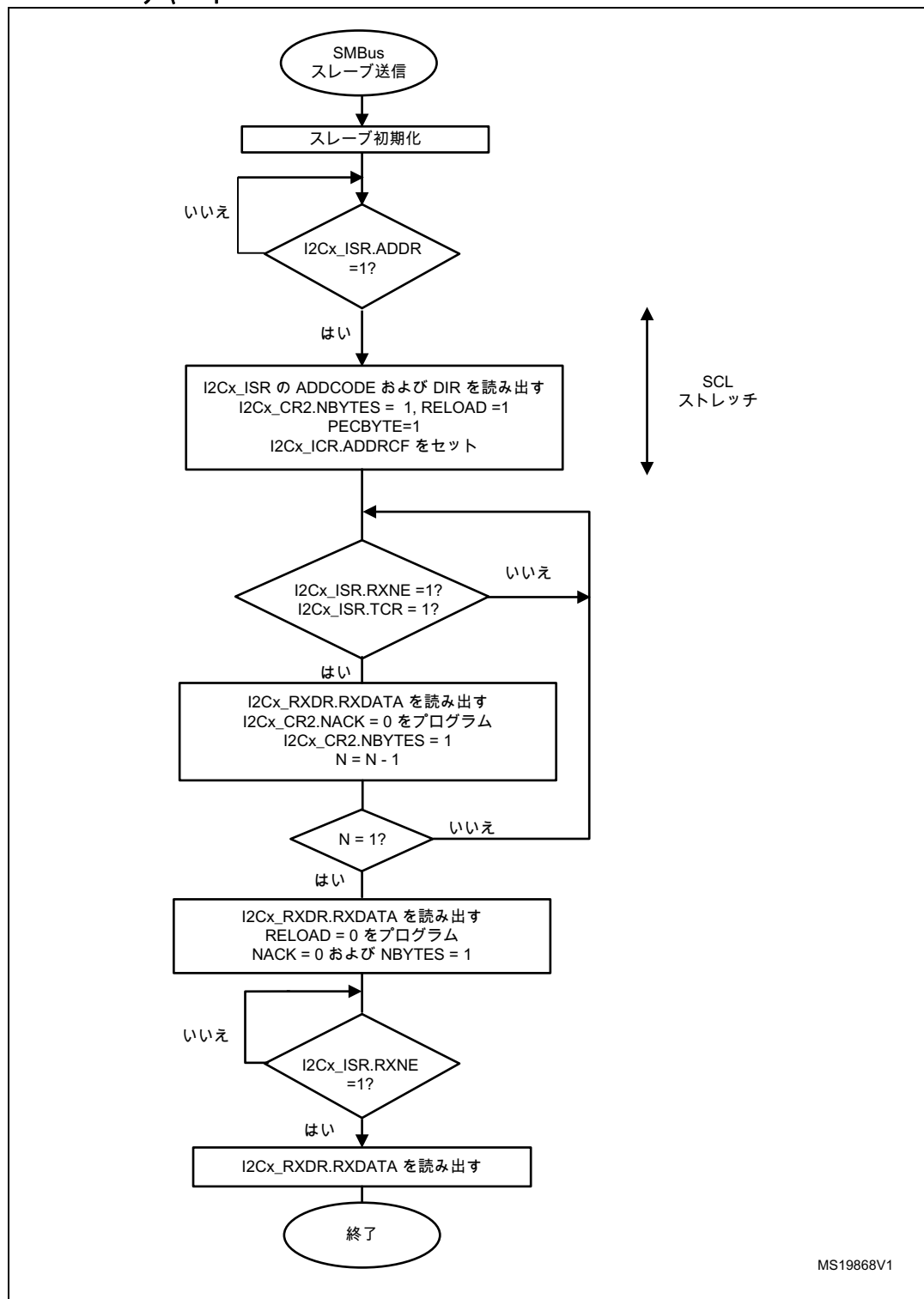
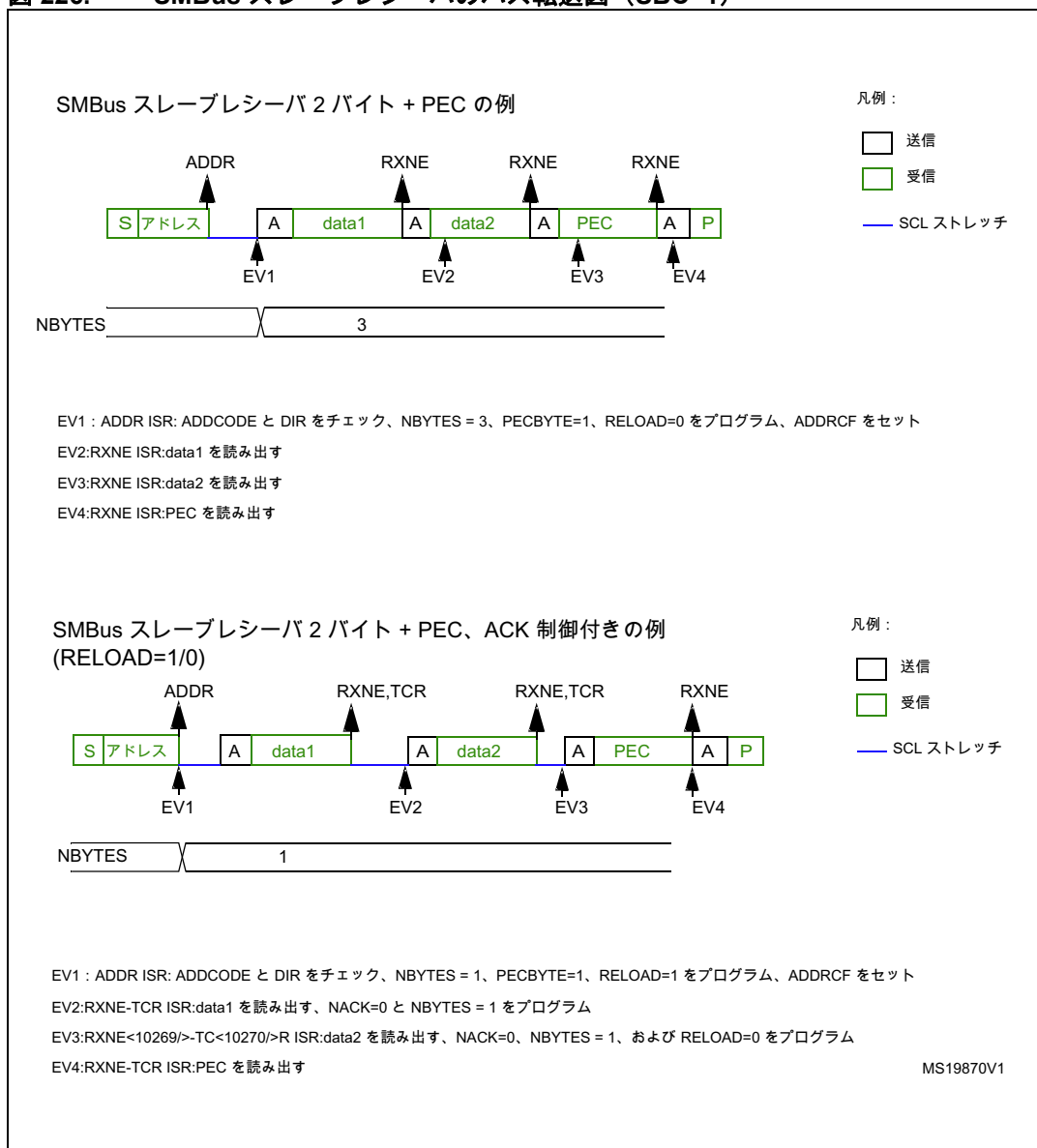


図 226. SMBus スレーブレシーバのバス転送図 (SBC=1)



このセクションは、SMBus 機能がサポートされるときにのみ適用されます。[セクション 28.3 : I2C の実装](#) を参照してください。

I2C マスタ転送管理 ([セクション 28.4.9 : I2C マスタモード](#) を参照) に加えて、SMBus をサポートするために、いくつか追加のソフトウェアフローチャートが用意されています。

SMBus マスタトランスミッタ

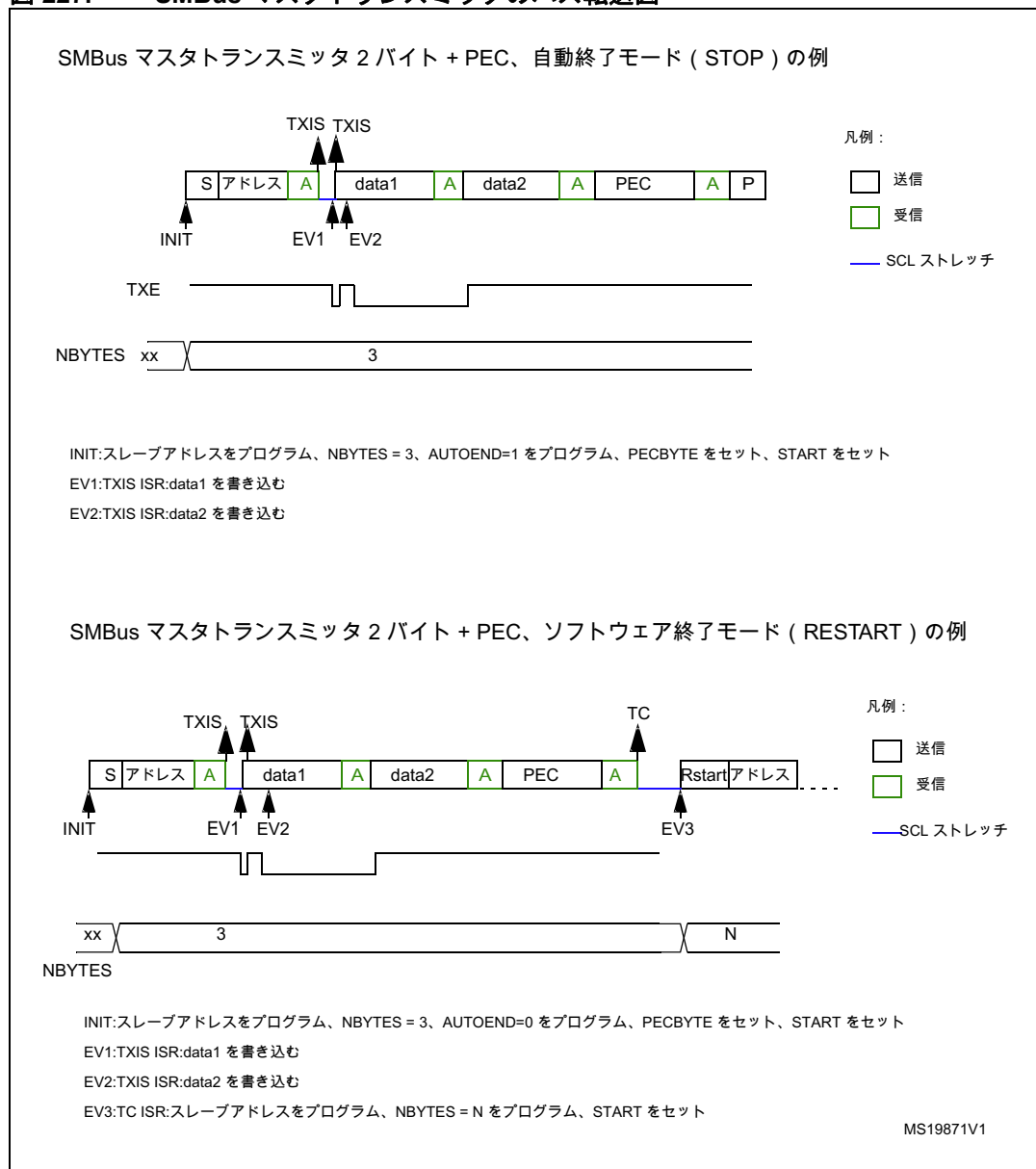
SMBus マスタが PEC を送信したいときには、START ビットをセットする前に、PECBYTE ビットをセットする必要があります。バイト数を NBYTES[7:0] フィールドでプログラムする必要があります。この場合、TXIS 割り込みの合計数は NBYTES-1 になります。したがって、NBYTES=0x1 のときに PECBYTE ビットがセットされた場合、I2Cx_PECR レジスタの内容が自動的に送信されます。

SMBus マスタが PEC 後に STOP コンディションを送信したい場合は、自動終了モードを選択してください (AUTOEND=1)。この場合、PEC 送信に続いて、STOP コンディションが自動的に送信されます。

SMBus マスタが PEC 後に RESTART コンディションを送信したい場合は、ソフトウェアモードを選択してください (AUTOEND=0)。この場合、NBYTES-1 が送信されると、PEC 送信後に I2Cx_PECR レジスタの内容が送信され、TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションを TC 割り込みサブルーチンでプログラムする必要があります。

注意： PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。

図 227. SMBus マスタトランスミッタのバス転送図



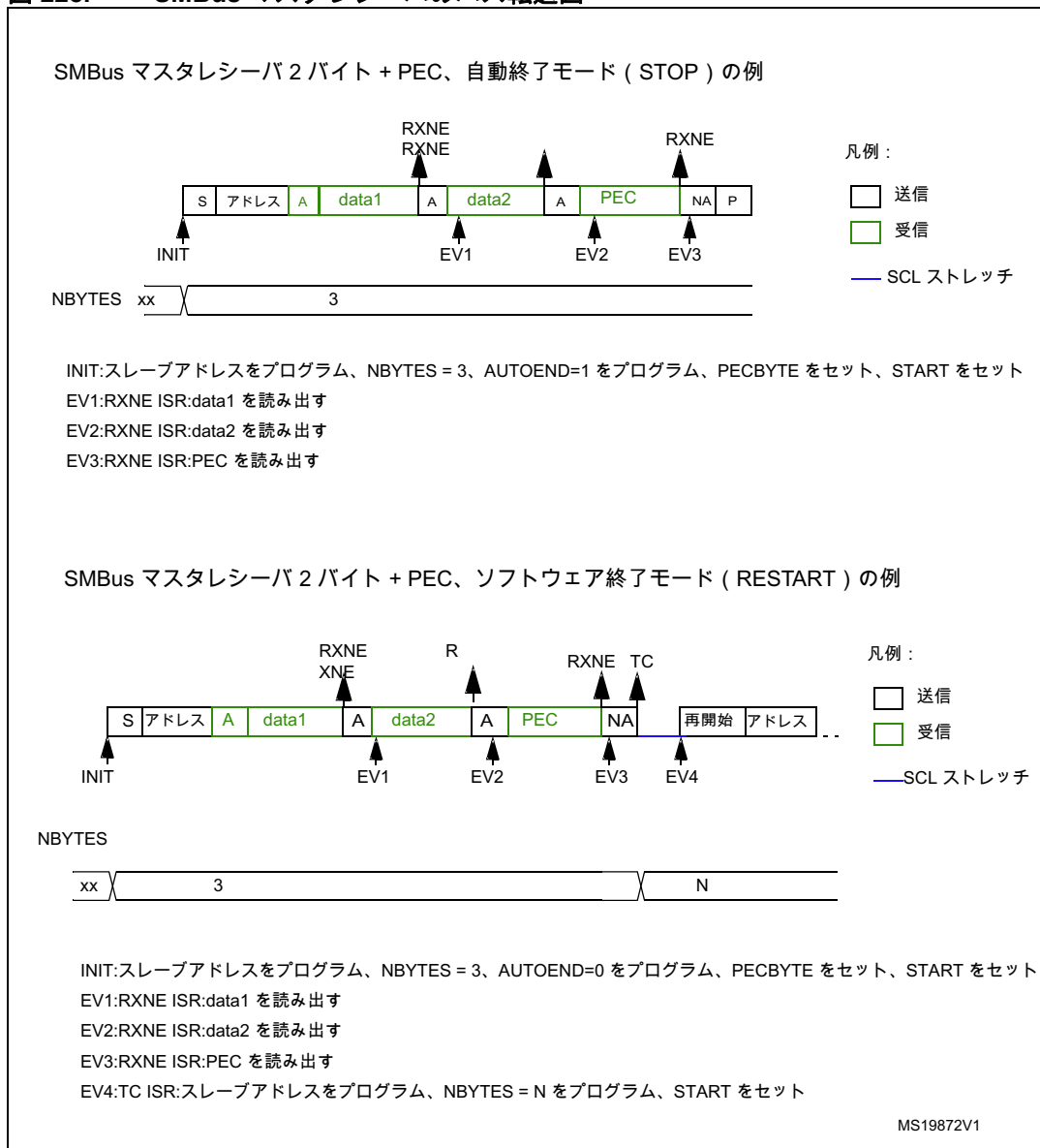
SMBus マスタレシーバ

SMBus マスタが転送終了時に PEC を受信してから STOP を受信したいときには、自動終了モードを選択できます (AUTOEND=1)。START ビットをセットする前に、PECBYTE ビットをセットする必要があります。スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2Cx_PECR レジスタの内容と自動的に照合されます。PEC バイトに対して NACK 応答が与えられた後、STOP コンディションが送信されます。

SMBus マスタが転送終了時に PEC バイトを受信してから RESTART を受信したいときには、ソフトウェアモードを選択する必要があります (AUTOEND=0)。START ビットをセットする前に、PECBYTE ビットをセットする必要があります。スレーブアドレスをプログラムする必要があります。この場合、NBYTES-1 データが受信された後、次の受信バイトが I2Cx_PECR レジスタの内容と自動的に照合されます。PEC バイト受信後に TC フラグがセットされ、SCL ラインローをストレッチします。RESTART コンディションは、TC 割り込みサブルーチンでプログラムできます。

注意： PECBYTE ビットは、RELOAD ビットがセットされているときには効果がありません。

図 228. SMBus マスタレシーバのバス転送図



28.4.15 アドレス一致時に STOP モードからウェイクアップ

このセクションは、STOP モードからのウェイクアップ機能がサポートされるときにのみ適用されます。[セクション 28.3 : I2C の実装](#)を参照してください。

I2C は、アドレス指定されたとき、MCU を STOP モードからウェイクアップできます (APB クロックはオフ)。すべてのアドレッシングモードがサポートされます。

STOP モードからのウェイクアップを有効にするには、I2Cx_CR1 レジスタの WUPEN ビットをセットします。STOP モードからのウェイクアップを可能にするには、HSI オシレータを I2CCLK のクロックソースとして選択する必要があります。

STOP モード中、HSI はオフです。START が検出されると、I2C インタフェースは HSI をオンに切り替えて、HSI がウェイクアップするまで SCLローをストレッチします。

HSI は、アドレス受信に使用されます。

アドレス一致の場合、I2C は、MCU のウェイクアップ時間の間、SCLローをストレッチします。ストレッチは、ADDR フラグがソフトウェアによってクリアされたときにリリースされ、転送は通常通りに続行されます。

アドレスが一致しなかった場合、HSI は再びオフになり、MCU はウェイクアップしません。

注： *I2C クロックがシステムクロックの場合、または WUPEN = 0 の場合、START 受信後も HSI オシレータはオンになりません。*

ADDR 割り込みによってのみ、MCU をウェイクアップできます。したがって、I2C がマスタとして、または ADDR フラグのセット後にアドレス指定されたスレーブとして転送を行っているときには、STOP モードに入らないでください。これを管理するには、ADDR 割り込みルーチンで SLEEPDEEP ビットをクリアして、STOPF フラグのセット後にのみ再びオンにセットします。

注意： デジタルフィルタは、STOP モードからのウェイクアップ機能と互換性がありません。DNF ビットが 0 でない場合、WUPEN ビットをセットしても効果はありません。

注意： この機能は、I2C クロックソースが HSI オシレータのときのみ使用できます。

注意： STOP モードからのウェイクアップ機能の正しい動作を保証するには、クロックストレッチを有効にする必要があります (NOSTRETCH=0)。

注意： STOP モードからのウェイクアップが無効な場合 (WUPEN=0)、STOP モードに入る前に、I2C ペリフェラルを無効にする必要があります (PE=0)。

28.4.16 エラー条件

以下は、通信エラーを引き起こす可能性のあるエラー条件です。

バスエラー (BERR)

バスエラーは、START または STOP コンディションが検出され、複数の 9 SCL クロックパルス後になかったときに検出されます。START または STOP コンディションは、SCL がハイと有的时候に SDA エッジが発生した場合に検出されます。

バスエラーフラグは、I2C がマスタまたはアドレス指定されたスレーブとして転送に関与する場合にのみ (すなわち、スレーブモードのアドレスフェーズでないとき)、セットされます。

スレーブモードで START または RESTART の誤配置が検出された場合、I2C は、正しい START コンディションの場合と同様に、アドレス認識状態に入ります。

バスエラーが検出されると、I2Cx_ISR レジスタの BERR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アービトレーション喪失 (ARLO)

アービトレーション喪失は、SDA ラインでハイレベルが送信されたが、SCL 立ち上がりエッジでローレベルがサンプリングされたときに検出されます。

- マスタモードでは、アービトレーション喪失は、アドレスフェーズ、データフェーズ、およびデータ確認応答フェーズで検出されます。この場合、SDA および SCL ラインはリリースされ、START 制御ビットがハードウェアによってクリアされ、マスタは自動的にスレーブモードに切り替わります。
- スレーブモードでは、アービトレーション喪失は、データフェーズとデータ確認応答フェーズで検出されます。この場合、転送は中止され、SCL および SDA ラインがリリースされます。

アービトレーション喪失が検出されると、I2Cx_ISR レジスタの ARLO フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

オーバーラン／アンダーランエラー (OVR)

オーバーランまたはアンダーラインエラーは、スレーブモードで NOSTRETCH=1 のとき、および次のときに検出されます：

- 受信時、新しいバイトが受信され、RXDR レジスタがまだ読み出されていないとき。新しい受信バイトは失われ、新しいバイトへの応答として NACK が自動的に送信されます。
- 送信時：
 - － STOPF=1 のときには、最初のデータバイトが送信されなければなりません。TXE=0、0xFF の場合、I2Cx_TXDR レジスタの内容が送信され、そうでない場合は送信されません。
 - － 新しいバイトが送信されるべきときに、I2Cx_TXDR レジスタがまだ書き込まれていなかった場合、0xFF が送信されます。

オーバーランまたはアンダーランエラーが検出されると、I2Cx_ISR レジスタの OVR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

パケットエラーチェックエラー (PECERR)

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

PEC エラーは、受信した PEC バイトが I2Cx_PECR レジスタの内容と一致しなかったときに検出されます。正しくない PEC の受信後、NACK が自動的に送信されます。

PEC エラーが検出されると、I2Cx_ISR レジスタの PECERR フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

タイムアウトエラー (TIMEOUT)

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

タイムアウトエラーは、次のような条件で発生します：

- TIDLE=0 であり、SCL が TIMEOUTA[11:0] ビットで定義された時間だけローのままであった場合：これは SMBus タイムアウトの検出に使用されます。
- TIDLE=1 であり、SDA および SCL が TIMEOUTA[11:0] ビットで定義された時間だけハイのままであった場合：これはバスアイドル状態の検出に使用されます。
- マスタ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合 (SMBus $t_{\text{LOW:MEXT}}$ パラメータ)。
- スレーブ累積クロックロー延長時間が TIMEOUTB[11:0] ビットで定義された時間に達した場合 (SMBus $t_{\text{LOW:SEXT}}$ パラメータ)。

マスタモードでタイムアウト違反が検出されると、STOP コンディションが自動的に送信されます。

スレーブモードでタイムアウト違反が検出されると、SDA および SCL ラインが自動的にリリースされます。

タイムアウトエラーが検出されると、I2Cx_ISR レジスタの TIMEOUT フラグがセットされ、I2Cx_CR1 レジスタの ERRIE ビットがセットされていた場合は割り込みが生成されます。

アラート (ALERT)

このセクションは、SMBus 機能がサポートされるときにのみ適用されます。 [セクション 28.3 : I2C の実装](#) を参照してください。

ALERT フラグは、I2C インタフェースがホストとして設定され (SMBHEN=1)、アラートピン検出が有効であり (ALERTEN=1)、SMBA ピンで立ち下がりがエッジが検出されたときにセットされます。I2Cx_CR1 レジスタの ERRIE ビットがセットされている場合は、割り込みが生成されます。

28.4.17 DMA リクエスト

DMA を使用した送信

送信について DMA (Direct Memory Access) を有効にするには、I2C_CR2 レジスタの TXDMAEN ビットをセットします。TXIS ビットがセットされるたびに、データは、DMA ペリフェラル ([セクション 11: ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)) を参照) を使用して設定された SRAM 領域から I2Cx_TXDR レジスタにロードされます。

データのみが DMA で転送されます。

- マスタモード: 初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます (送信されたスレーブアドレスを DMA で転送することはできません)。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。 [マスタトランスミッタ \(644 ページ\)](#) を参照してください。
- スレーブモードでは:
 - NOSTRETCH=0 では、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
 - NOSTRETCH=1 では、アドレス一致イベントの前に DMA を初期化する必要があります。
- SMBus をサポートする場合: PEC 転送は NBYTES カウンタによって管理されます。 [SMBus スレーブトランスミッタ \(659 ページ\)](#) および [SMBus マスタトランスミッタ \(662 ページ\)](#) を参照してください。

注: DMA が送信に使用される場合、TXIE ビットが有効である必要はありません。

DMA を使用した受信

受信について DMA (Direct Memory Access) を有効にするには、I2C_CR1 レジスタの RXDMAEN ビットをセットします。RXNE ビットがセットされているときには、データは、I2Cx_TXDR レジスタから DMA ペリフェラル ([セクション 11: ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)) を参照) を使用して設定された SRAM 領域にロードされます。データのみ (PEC を含む) が DMA で転送されます。

- マスタモード、初期化、スレーブアドレス、方向、バイト数、および START ビットはソフトウェアによってプログラムされます。すべてのデータが DMA を使用して転送されるときには、START ビットをセットする前に、DMA を初期化する必要があります。転送の終了は、NBYTES カウンタによって管理されます。コード例については、「付録」セクション [A.14.6: I2C slave transmitter code example](#) を参照してください。
- NOSTRETCH=0 のスレーブモードでは、すべてのデータが DMA を使用して転送されるときには、アドレス一致イベントの前、または ADDR 割り込みサブルーチンで、ADDR をクリアする前に DMA を初期化する必要があります。
- SMBus がサポートされる場合 ([セクション 28.3: I2C の実装](#) を参照): PEC 転送は NBYTES カウンタによって管理されます。 [SMBus スレーブレシーバ \(660 ページ\)](#) および [SMBus マスタレシーバ \(664 ページ\)](#) を参照してください。

注: DMA が受信に使用される場合、RXIE ビットが有効である必要はありません。

28.4.18 デバッグモード

マイクロコントローラがデバッグモードに入ると（コア停止）、DBG モジュールの DBG_I2Cx_SMBUS_TIMEOUT 設定ビットに応じて、SMBUS タイムアウトは、通常の動作を続行するか、あるいは停止します。

28.5 I2C 低電力モード

表 112. 低電力モード

| モード | 説明 |
|---------|--|
| スリープ | 効果なし I2C 割り込みによって、デバイスは SLEEP モードから復帰します。 |
| STOP | I2C レジスタの内容は保たれます。WUPEN=1 の場合：アドレス認識が機能します。I2C アドレス一致条件によって、デバイスは STOP モードから復帰します。WUPEN=0 の場合：STOP モードに入る前に I2C を無効にする必要があります。 |
| STANDBY | I2C ペリフェラルはパワーダウンされ、STANBY 終了後に再初期化する必要があります。 |

28.6 I2C 割り込み

次の表に、I2C 割り込みリクエストの一覧を示します。

表 113. I2C 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イベントフラグ/ 割り込みのクリア方法 | 割り込み有効制御 ビット |
|-----------------------|---------|--|-----------------|
| 受信バッファノットエンプティ | RXNE | I2Cx_RXDR レジスタを 読み出す | RXIE |
| 送信バッファ割り込みステータス | TXIS | I2Cx_TXDR レジスタに 書き込む | TXIE |
| STOP 検出割り込みフラグ | STOPF | STOPCF=1 を書き込む | STOPIE |
| 転送完了再ロード | TCR | I2Cx_CR2 の NBYTES[7:0] ¹ に 0 を書き込む | TCIE |
| 転送完了 | TC | START=1 または STOP=1 を 書き込む | |
| アドレス一致 | ADDR | ADDRCF=1 を書き込む | ADDRIE |
| NACK 受信 | NACKF | NACKCF=1 を書き込む | NACKIE |
| バスエラー | BERR | BERRCF=1 を書き込む | ERRIE |
| アービトレーション喪失 | ARLO | ARLOCF=1 を書き込む | |
| オーバーラン/アンダーラン | OVR | OVRCF=1 を書き込む | |
| PEC エラー | PECERR | PECERRCF=1 を書き込む | |
| タイムアウト/ t_{LOW} エラー | TIMEOUT | TIMEOUTCF=1 を書き込む | |
| SMBus アラート | ALERT | ALERTCF=1 を書き込む | |

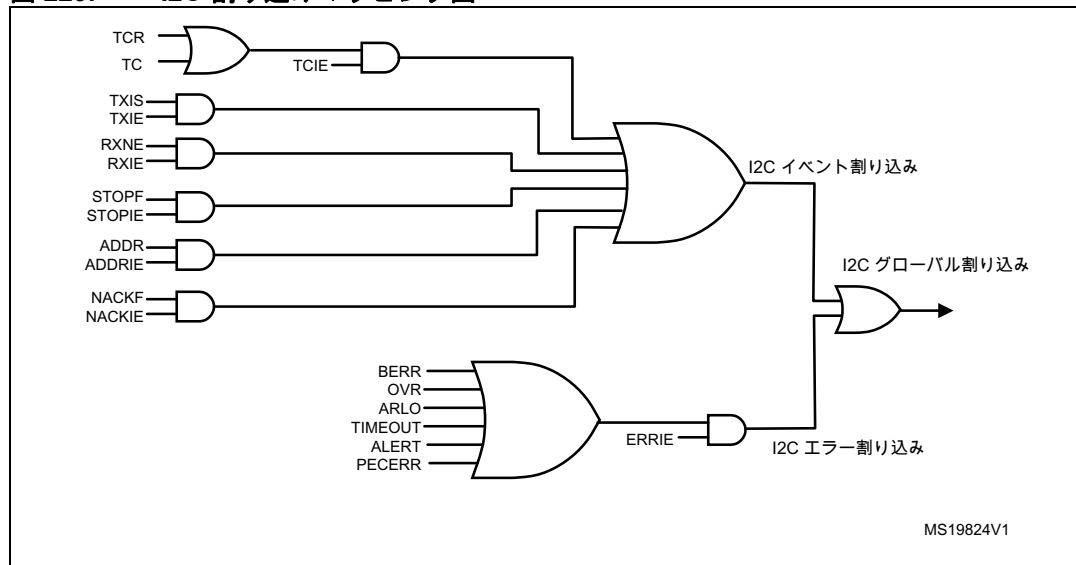
製品実装に応じて、これらの割り込みイベントのすべてで同じ割り込みベクトルを共有するか (I2C グローバル割り込み)、2 つの割り込みベクトル (I2C イベント割り込みと I2C エラー割り込み) にグループ化することができます。詳細については、[表 45 : ベクタテーブル](#) を参照してください。

I2C 割り込みを有効にするには、次のシーケンスが必要です：

1. NVIC で I2C IRQ チャンネルを設定し、有効にします。
2. 割り込みを生成するように I2C を設定します。

I2C ウェイクアップイベントは EXTI コントローラに接続されます ([セクション 13.5: EXTI レジスタ](#) を参照)。

図 229. I2C 割り込みマッピング図



28.7 I2C レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタは、ワード (32 ビット) 単位でアクセスされます。

28.7.1 制御レジスタ 1 (I2Cx_CR1)

アドレスオフセット : 0x00

リセット値 : 0x0000 0000

アクセス : このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 $2 \times \text{PCLK1} + 6 \times \text{I2CCLK}$ です。

| | | | | | | | | | | | | | | | |
|---------|---------|------|--------|------|------|------|------|-------|---------|--------|--------|--------|-------|-----------|-----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PECEN | ALERTEN | SMBDEN | SMBHEN | GCEN | WUPEN | NOSTRETCH | SBC |
| | | | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RXDMAEN | TXDMAEN | Res. | ANFOFF | DNF | | | | ERRIE | TCIE | STOPIE | NACKIE | ADDRIE | RXIE | TXIE | PE |
| r/w | r/w | | r/w | r/w | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

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

ビット 23 **PECEN** : PEC イネーブル

- 0 : PEC 計算は無効です。
- 1 : PEC 計算は有効です。

注 : **SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。**

ビット 22 **ALERTEN**:SMBus アラート有効

デバイスモード (SMBHEN=0) :

- 0 : SMBA ピンをハイにリリースし、アラート応答アドレスヘッダを無効にします。0001100x の後に NACK が続きます。
- 1 : SMBA ピンをローに駆動し、アラート応答アドレスヘッダを有効にします : 0001100x の後に ACK が続きます。

ホストモード (SMBHEN=1) :

- 0 : SMBus アラートピン (SMBA) はサポートされません。
- 1 : SMBus アラートピン (SMBA) はサポートされます。

注 : **ALERTEN=0 のときには、SMBA ピンを標準 GPIO として使用できます。**
SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。

ビット 21 **SMBDEN** : SMBus デバイスデフォルトアドレス有効

- 0 : デバイスデフォルトアドレス無効。アドレス 0b1100001x は NACK されます。
- 1 : デバイスデフォルトアドレス有効。アドレス 0b1100001x は ACK されます。

注 : **SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。**

ビット 20 **SMBHEN** : SMBus ホストアドレス有効

- 0 : ホストアドレス無効。アドレス 0b0001000x は NACK されます。
- 1 : ホストアドレス有効。アドレス 0b0001000x は ACK されます。

注 : **SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。**



ビット 19 **GCEN** : 同報イネーブル

- 0 : 同報は無効です。アドレス 0b00000000 は NACK されます。
- 1 : 同報は有効です。アドレス 0b00000000 は ACK されます。

ビット 18 **WUPEN** : STOP モードからのウェイクアップ有効

- 0 : STOP モードからのウェイクアップ無効。
- 1 : STOP モードからのウェイクアップ有効。

注 : **STOP モードからのウェイクアップ機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3: I2C の実装を参照してください。**

注 : **WUPEN は、DNF = 0000 のときのみセットできます。**

ビット 17 **NOSTRETCH** : クロックストレッチ無効

このビットは、スレーブモードでのクロックストレッチを無効にするために使用されます。

- 0 : クロックストレッチ有効
- 1 : クロックストレッチ無効

注 : **このビットは、I2C が無効 (PE = 0) のときのみプログラムできます。**

ビット 16 **SBC** : スレーブバイト制御

このビットは、スレーブモードでのハードウェアバイト制御を有効にするために使用されます。

- 0 : スレーブバイト制御無効
- 1 : スレーブバイト制御有効

ビット 15 **RXDMAEN** : DMA 受信リクエスト有効

- 0 : DMA モードは受信に無効
- 1 : DMA モードは受信に有効

ビット 14 **TXDMAEN** : DMA 送信リクエスト有効

- 0 : DMA モードは送信に無効
- 1 : DMA モードは送信に有効

ビット 13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **ANFOFF** : アナログノイズフィルタ OFF

- 0 : アナログノイズフィルタ有効
- 1 : アナログノイズフィルタ無効

注 : **このビットは、I2C が無効 (PE = 0) のときのみプログラムできます。**

ビット 11:8 **DNF[3:0]** : デジタルノイズフィルタ

これらのビットは、SDA および SCL 入力のデジタルノイズフィルタを設定するために使用されます。デジタルフィルタは、最大 $DNF[3:0] \cdot t_{I2CCCLK}$ の長さのスパイクを除去します。

- 0000 : デジタルフィルタ無効
- 0001 : デジタルフィルタは有効であり、最大 $1 \cdot t_{I2CCCLK}$ の除去能力を持ちます。

...

- 1111 : デジタルフィルタは有効であり、最大 $15 \cdot t_{I2CCCLK}$ の除去能力を持ちます。

注 : **アナログフィルタも有効化した場合、デジタルフィルタがアナログフィルタに追加されます。このフィルタは、I2C が無効 (PE = 0) のときのみプログラムできます。**

ビット 7 **ERRIE** : エラー割り込み有効

0 : エラー検出割り込み無効

1 : エラー検出割り込み有効

注 : 次のようなエラーが発生すると、割り込みが生成されます :

アービトレーション喪失 (ARLO)

バスエラー検出 (BERR)

オーバーラン/アンダーラン (OVR)

タイムアウト検出 (TIMEOUT)

PEC エラー検出 (PECERR)

アラートピンイベント検出 (ALERT)

ビット 6 **TCIE** : 転送完了割り込み有効

0 : 転送完了割り込み無効

1 : 転送完了割り込み有効

注 : 次のようなイベントが発生すると、割り込みが生成されます :

転送完了 (TC)

転送完了再ロード (TCR)

ビット 5 **STOPIE** : STOP 検出割り込み有効

0 : STOP 検出 (STOPF) 割り込み無効

1 : STOP 検出 (STOPF) 割り込み有効

ビット 4 **NACKIE** : 非確認応答受信割り込み有効

0 : 非確認応答 (NACKF) 受信割り込み無効

1 : 非確認応答 (NACKF) 受信割り込み有効

ビット 3 **ADDRIE** : アドレス一致割り込み有効 (スレーブのみ)

0 : アドレス一致 (ADDR) 割り込み無効

1 : アドレス一致 (ADDR) 割り込み有効

ビット 2 **RXIE** : RX 割り込み有効

0 : 受信 (RXNE) 割り込み無効

1 : 受信 (RXNE) 割り込み有効

ビット 1 **TXIE** : TX 割り込み有効

0 : 送信 (TXIS) 割り込み無効

1 : 送信 (TXIS) 割り込み有効

ビット 0 **PE** : ペリフェラル有効

0 : ペリフェラルは無効です。

1 : ペリフェラルは有効です。

注 : **PE=0** のとき、**I2C SCL** および **SDA** ラインはリリースされます。内部ステートマシンおよびステータスビットはリセット値に戻されます。クリアされたときには、**PE** は少なくとも 3 APB クロックサイクルの間、ローに保たれる必要があります。

28.7.2 制御レジスタ 2 (I2Cx_CR2)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

アクセス : このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 $2 \times \text{PCLK1} + 6 \times \text{I2CCLK}$ です。

| | | | | | | | | | | | | | | | |
|------|------|-------|-------------|-------|-------------|-------------|------------|-------------|----|----|----|----|----|----|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | PEC BYTE | AUTOE ND | RE LOAD | NBYTES[7:0] | | | | | | | |
| | | | | | rs | rw | rw | rw | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| NACK | STOP | START | HEAD1 OR | ADD10 | RD_WR N | SADD[9:0] | | | | | | | | | |
| rs | rs | rs | rw | rw | rw | rw | | | | | | | | | |

ビット 31:27 予約済みであり、リセット値のままにしておかなければなりません。

ビット 26 **PECBYTE** : パケットエラーチェックバイト

このビットはソフトウェアによってセットされ、PEC が転送されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、また、PE=0 のとき、ハードウェアによってクリアされます。

0 : PEC 転送なし。

1 : PEC 送信／受信がリクエストされます。

注 : このビットに 0 を書き込んでも、効果はありません。

このビットは、RELOAD がセットされているときには効果がありません。

このビットは、SBC=0 のとき、スレーブモードでは効果がありません。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。

ビット 25 **AUTOEND** : 自動終了モード (マスタモード)

このビットは、ソフトウェアによってセット／クリアされます。

0 : ソフトウェア終了モード : NBYTES データが転送されると TC フラグがセットされ、SCL ローをストレッチします。

1 : 自動終了モード : NBYTES データが転送されると、STOP コンディションが自動的に送信されます。

注 : このビットは、スレーブモードまたは RELOAD ビットがセットされているときには効果がありません。

ビット 24 **RELOAD** : NBYTES 再ロードモード

このビットは、ソフトウェアによってセット／クリアされます。

0 : 転送は、NBYTES データの転送後、完了します (STOP または RESTART が続きます)。

1 : 転送は、NBYTES データの転送後に完了しません (NBYTES が再ロードされます)。NBYTES データが転送されると TCR フラグがセットされ、SCL ローをストレッチします。

ビット 23:16 **NBYTES[7:0]** : バイト数

送受信されるバイト数は、ここでプログラムされます。このフィールドは、SBC=0 のスレーブモードでは効果がありません。

注 : START ビットがセットされているときに、これらのビットを変更することはできません。

ビット 15 NACK : NACK 生成 (スレーブモード)

このビットはソフトウェアによってセットされ、NACK が送信されたとき、または STOP コンディションあるいはアドレス一致を受信したとき、または PE=0 のとき、ハードウェアによってクリアされます。

0 : 現在の受信バイト後に ACK が送信されます。

1 : 現在の受信バイト後に NACK が送信されます。

注 : このビットに 0 を書き込んでも、効果はありません。

このビットは、スレーブモードでのみ使用されます : マスタレシーバモードでは、NACK ビットの値にかかわらず、STOP または RESTART コンディション前の最後のバイト後に NACK が自動的に生成されます。

スレーブレシーバ NOSTRETCH モードでオーバーランが発生すると、NACK ビットの値にかかわらず、NACK が自動的に生成されます。

ハードウェア PEC チェックが有効なとき (PECBYTE=1)、PEC 確認応答値は NACK 値に依存しません。

ビット 14 STOP : STOP 生成 (マスタモード)

このビットはソフトウェアによってセットされ、STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。

マスタモード :

0 : STOP 生成なし。

1 : 現在のバイト転送後の STOP 生成。

注 : このビットに 0 を書き込んでも、効果はありません。

ビット 13 START : START 生成

このビットはソフトウェアによってセットされ、START とアドレスシーケンスが送信された後、アービトラクション喪失によって、タイムアウトエラー検出によって、または PE=0 のときに、ハードウェアによってクリアされます。I2Cx_ICR レジスタの ADDRCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアすることもできます。

0 : START 生成なし。

1 : RESTART/START 生成 :

– I2C がすでにマスタモードであり、AUTOEND=0 の場合、このビットをセットすると、RELOAD=0 のとき、NBYTES 転送の終了後に REPEATED START コンディションが生成されます。

– そうでない場合、このビットをセットすると、バスがフリーになると、START コンディションが生成されます。

注 : このビットに 0 を書き込んでも、効果はありません。

バスが BUSY の場合、または I2C がスレーブモードの場合でも、START ビットをセットできません。

このビットは、RELOAD がセットされているときには効果がありません。

ビット 12 HEAD10R : 読み出し方向のみの 10 ビットアドレスヘッダ (マスタレシーバモード)

0 : マスタは完全な 10 ビットスレーブアドレス読み出しシーケンスを送信します : START + 2 バイトの書き込み方向の 10 ビットアドレス + RESTART + 読み出し方向の 10 ビットアドレスの最初の 7 ビット。

1 : マスタは 10 ビットアドレスの最初の 7 ビットのみを送信し、その後に読み出し方向を送信します。

注 : START ビットがセットされているときに、このビットを変更することはできません。

ビット 11 ADD10 : 10 ビットアドレッシングモード (マスタモード)

0 : マスタは 7 ビットアドレッシングモードで動作します。

1 : マスタは 10 ビットアドレッシングモードで動作します。

注 : START ビットがセットされているときに、このビットを変更することはできません。

ビット 10 RD_WRN : 転送方向 (マスタモード)

0 : マスタは書き込み転送をリクエストします。

1 : マスタは読み出し転送をリクエストします。

注 : START ビットがセットされているときに、このビットを変更することはできません。



ビット 9:8 **SADD[9:8]** : スレーブアドレスビット 9:8 (マスタモード)

7 ビットアドレッシングモード (ADD10=0) :

これらのビットは無視されます。

10 ビットアドレッシングモード (ADD10=1) :

これらのビットには、送信されるスレーブアドレスのビット 9:8 を書き込みます。

注: *START ビットがセットされているときに、これらのビットを変更することはできません。*

ビット 7:1 **SADD[7:1]** : スレーブアドレスビット 7:1 (マスタモード)

7 ビットアドレッシングモード (ADD10=0) :

これらのビットには、送信される 7 ビットのスレーブアドレスを書き込みます。

10 ビットアドレッシングモード (ADD10=1) :

これらのビットには、送信されるスレーブアドレスのビット 7:1 を書き込みます。

注: *START ビットがセットされているときに、これらのビットを変更することはできません。*

ビット 0 **SADD0** : スレーブアドレスビット 0 (マスタモード)

7 ビットアドレッシングモード (ADD10=0) :

このビットは無視されます。

10 ビットアドレッシングモード (ADD10=1) :

このビットには、送信されるスレーブアドレスのビット 0 を書き込みます。

注: *START ビットがセットされているときに、これらのビットを変更することはできません。*

28.7.3 Own Address 1 レジスタ (I2Cx_OAR1)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

アクセス : このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 $2 \times \text{PCLK1} + 6 \times \text{I2CCLK}$ です。

| | | | | | | | | | | | | | | | |
|-------|------|------|------|------|-------------|----------|------|----------|------|------|------|------|------|------|--------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OA1EN | Res. | Res. | Res. | Res. | OA1 MODE | OA1[9:8] | | OA1[7:1] | | | | | | | OA1[0] |
| rw | | | | | rw | rw | | rw | | | | | | | rw |

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15 **OA1EN** : Own Address 1有効

- 0 : Own Address 1無効受信されたスレーブアドレス OA1 は NACK されます。
- 1 : Own Address 1は有効です。受信されたスレーブアドレス OA1 は ACK されます。

ビット 14:11 予約済みであり、リセット値のままにしておかなければなりません。

ビット 10 **OA1MODE** Own Address 1 10 ビットモード

- 0 : Own Address 1は 7 ビットアドレスです。
- 1 : Own Address 1は 10 ビットアドレスです。

注 : このビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 9:8 **OA1[9:8]** : インタフェースアドレス

- 7 ビットアドレッシングモード : 無視されます。
- 10 ビットアドレッシングモード : アドレスのビット 9:8。

注 : これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 7:1 **OA1[7:1]** : インタフェースアドレス

アドレスのビット 7:1

注 : これらのビットは、OA1EN=0 のときのみ書き込むことができます。

ビット 0 **OA1[0]** : インタフェースアドレス

- 7 ビットアドレッシングモード : 無視されます。
- 10 ビットアドレッシングモード : アドレスのビット 0。

注 : このビットは、OA1EN=0 のときのみ書き込むことができます。

28.7.4 Own Address 2 レジスタ (I2Cx_OAR2)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

アクセス : このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 $2 \times \text{PCLK1} + 6 \times \text{I2CCLK}$ です。

| | | | | | | | | | | | | | | | |
|-------|------|------|------|------|-------------|------|------|------|----------|------|------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OA2EN | Res. | Res. | Res. | Res. | OA2MSK[2:0] | | | | OA2[7:1] | | | | | | |
| rw | | | | | rw | | | | rw | | | | | | |

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15 **OA2EN** : Own Address 2 有効

- 0 : Own Address 2無効受信されたスレーブアドレス OA2 は NACK されます。
- 1 : Own Address 2 有効受信されたスレーブアドレス OA2 は ACK されます。

ビット 14:11 予約済みであり、リセット値のままにしておかなければなりません。

ビット 10:8 **OA2MSK[2:0]** : Own Address 2マスク

- 000 : マスクなし。
- 001 : OA2[1] はマスクされ、無視されます。OA2[7:2] のみ比較されます。
- 010 : OA2[2:1] はマスクされ、無視されます。OA2[7:3] のみ比較されます。
- 011 : OA2[3:1] はマスクされ、無視されます。OA2[7:4] のみ比較されます。
- 100 : OA2[4:1] はマスクされ、無視されます。OA2[7:5] のみ比較されます。
- 101 : OA2[5:1] はマスクされ、無視されます。OA2[7:6] のみ比較されます。
- 110 : OA2[6:1] はマスクされ、無視されます。OA2[7] のみ比較されます。
- 111 : OA2[7:1] はマスクされ、無視されます。比較は行われず、すべての（予約済みを除く）7 ビット受信アドレスが確認応答されます。

注 : これらのビットは、OA2EN=0 のときのみ書き込むことができます。
OA2MSK が 0 でなくなると、予約済み I2C アドレス (0b0000xxx および 0b1111xxx) は、比較が一致した場合でも確認応答されません。

ビット 7:1 **OA2[7:1]** : インタフェースアドレス

アドレスのビット 7:1

注 : これらのビットは、OA2EN=0 のときのみ書き込むことができます。

ビット 0 予約済みであり、リセット値のままにしておかなければなりません。

28.7.5 タイミングレジスタ (I2Cx_TIMINGR)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

アクセス : ウェイト状態なし

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

ビット 31:28 **PRESC[3:0]** : タイミングプリスケアラ

このフィールドは、データのセットアップおよびホールドカウンタ ([I2C のタイミング \(625 ページ\)](#) を参照) と SCL ハイおよびローレベルカウンタ ([I2C マスタ初期化 \(640 ページ\)](#) を参照) に使用されるクロック周期 t_{PRESC} を生成するため、I2CCLK をプリスケールするために使用されます。

$$t_{\text{PRESC}} = (\text{PRESC} + 1) \times t_{\text{I2CCLK}}$$

ビット 27:24 予約済みであり、リセット値のままにしておかなければなりません。

ビット 23:20 **SCLDEL[3:0]** : データセットアップ時間

このフィールドは、送信モードの SDA エッジと SCL 立ち上がりエッジの間に遅延 t_{SCLDEL} を生成するために使用されます。

$$t_{\text{SCLDEL}} = (\text{SCLDEL} + 1) \times t_{\text{PRESC}}$$

注 : t_{SCLDEL} は、 $t_{\text{SU:DAT}}$ タイミングを生成するために使用されます。

ビット 19:16 **SDADEL[3:0]** : データホールド時間

このフィールドは、送信モードの SCL 立ち下がりエッジと SDA エッジの間に遅延 t_{SDADEL} を生成するために使用されます。

$$t_{\text{SDADEL}} = \text{SDADEL} \times t_{\text{PRESC}}$$

注 : t_{SDADEL} は、 $t_{\text{HD:DAT}}$ タイミングを生成するために使用されます。

ビット 15:8 **SCLH[7:0]** : SCL ハイ周期 (マスタモード)

このフィールドは、マスタモードで SCL ハイ周期を生成するために使用されます。

$$t_{\text{SCLH}} = (\text{SCLH} + 1) \times t_{\text{PRESC}}$$

注 : t_{SCLH} は、 $t_{\text{SU:STO}}$ および $t_{\text{HD:STA}}$ タイミングを生成するためにも使用されます。

ビット 7:0 **SCLL[7:0]** : SCL ロー周期 (マスタモード)

このフィールドは、マスタモードで SCL ロー周期を生成するために使用されます。

$$t_{\text{SCLL}} = (\text{SCLL} + 1) \times t_{\text{PRESC}}$$

注 : t_{SCLL} は、 t_{BUF} および $t_{\text{SU:STA}}$ タイミングを生成するためにも使用されます。

注 : このレジスタは、I2C が無効 ($\text{PE}=0$) のときに設定する必要があります。

28.7.6 タイムアウトレジスタ (I2Cx_TIMEOUTR)

アドレスオフセット : 0x14

リセット値 : 0x0000 0000

アクセス : このレジスタへの書き込みアクセスが進行中のときに書き込みアクセスが発生した場合を除き、ウェイト状態なし。この場合、前の書き込みアクセスが完了するまで、2 番目の書き込みアクセスにウェイト状態が挿入されます。2 番目の書き込みアクセスの遅延は、最大 $2 \times \text{PCLK1} + 6 \times \text{I2CCLK}$ です。

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

ビット 31 **TEXTEN** : 拡張クロックタイムアウト有効

- 0 : 拡張クロックタイムアウト検出は無効です。
- 1 : 拡張クロックタイムアウト検出は有効です。 $t_{\text{LOW:EXT}}$ を超える累積 SCL ストレッチが I2C インタフェースによって行われると、タイムアウトエラーが検出されます (TIMEOUT=1)。

ビット 30:29 予約済みであり、リセット値のままにしておかなければなりません。

ビット 27:16 **TIMEOUTB[11:0]** : バスタイムアウト B

このフィールドは、累積クロック拡張タイムアウトを設定するために使用されます :
マスタモードでは、マスタ累積クロックロー拡張時間 ($t_{\text{LOW:MEXT}}$) が検出されます。
スレーブモードでは、スレーブ累積クロックロー拡張時間 ($t_{\text{LOW:SEXT}}$) が検出されます。
 $t_{\text{LOW:EXT}} = (\text{TIMEOUTB} + 1) \times 2048 \times t_{\text{I2CCLK}}$

注 : これらのビットは、**TEXTEN=0** のときのみ書き込むことができます。

ビット 15 **TIMOUTEN** : クロックタイムアウト有効

- 0 : SCL タイムアウト検出は無効です。
- 1 : SCL タイムアウト検出は有効です : SCL が t_{TIMEOUT} ($\text{TIDLE}=0$) を超えてローであるか、 t_{IDLE} ($\text{TIDLE}=1$) を超えてハイであった場合、タイムアウトエラーが検出されます (TIMEOUT=1)。

ビット 14:13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **TIDLE** : アイドルクロックタイムアウト検出

- 0 : TIMEOUTA は、SCL ロータイムアウトの検出に使用されます。
- 1 : TIMEOUTA は、SCL と SDA の両方のハイタイムアウト (バスアイドル条件) の検出に使用されます。

注 : このビットは、**TIMOUTEN=0** のときのみ書き込むことができます。

ビット 11:0 **TIMEOUTA[11:0]** : バスタイムアウト A

このフィールドは、以下を設定するために使用されます :
– $\text{TIDLE}=0$ のときの SCL ロータイムアウト条件 t_{TIMEOUT}
 $t_{\text{TIMEOUT}} = (\text{TIMEOUTA} + 1) \times 2048 \times t_{\text{I2CCLK}}$
– $\text{TIDLE}=1$ のときのバスアイドル条件 (SCL と SDA の両方のハイ)
 $t_{\text{IDLE}} = (\text{TIMEOUTA} + 1) \times 4 \times t_{\text{I2CCLK}}$

注 : これらのビットは、**TIMOUTEN=0** のときのみ書き込むことができます。

注 : **SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 28.3 : I2C の実装** を参照してください。

28.7.7 割り込みおよびステータスレジスタ (I2Cx_ISR)

アドレスオフセット : 0x18
リセット値 : 0x0000 0001
アクセス : ウェイト状態なし

| | | | | | | | | | | | | | | | |
|------|------|-------|----------|---------|------|------|------|--------------|----|-------|-------|------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADDCODE[6:0] | | | | | | | DIR |
| | | | | | | | | r | | | | | | | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BUSY | Res. | ALERT | TIME OUT | PEC ERR | OVR | ARLO | BERR | TCR | TC | STOPF | NACKF | ADDR | RXNE | TXIS | TXE |
| r | | r | r | r | r | r | r | r | r | r | r | r | r | r_w1 | r_w1 |

- ビット 31:24 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 23:17 **ADDCODE[6:0]**: アドレス一致コード (スレーブモード)
これらのビットは、アドレス一致イベントが発生したときに (ADDR = 1)、受信したアドレスで更新されます。
10 ビットアドレスの場合、ADDCODE は 10 ビットのヘッダとその後のアドレスの 2 つの MSB を示します。
- ビット 16 **DIR**: 転送方向 (スレーブモード)
このフラグは、アドレス一致イベントが発生したときに (ADDR=1)、更新されます。
0: 書き込み転送、スレーブはレシーバモードになります。
1: 読み出し転送、スレーブはトランスミッタモードになります。
- ビット 15 **BUSY**: バスビジー
このフラグは、バスで通信が進行中であることを示します。START コンディションが検出されたときに、ハードウェアによってセットされます。STOP コンディションが検出されたとき、または PE=0 のときにハードウェアによってクリアされます。
- ビット 14 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 13 **ALERT**: SMBus アラート
このフラグは、SMBHEN=1 (SMBus ホスト設定)、ALERTEN=1、および SMBALERT イベント (立ち下がリエッジ) が SMBA ピンで検出されたときに、ハードウェアによってセットされます。ALERTCF ビットをセットすることによって、ソフトウェアによってクリアされます。
注: このビットは、PE=0 のとき、ハードウェアによってクリアされます。
SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。
- ビット 12 **TIMEOUT**: タイムアウトまたは t_{LOW} 検出フラグ
このフラグは、タイムアウトまたは拡張クロックタイムアウトが発生したときに、ハードウェアによってセットされます。TIMEOUTCF ビットをセットすることによって、ソフトウェアによってクリアされます。
注: このビットは、PE=0 のとき、ハードウェアによってクリアされます。
SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。

ビット 11 PECERR : 受信時の PEC エラー

このフラグは、受信した PEC が PEC レジスタの内容に一致しないときに、ハードウェアによってセットされます。正しくない PEC の受信後、NACK が自動的に送信されます。PECCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。

ビット 10 OVR : オーバーラン/アンダーラン (スレーブモード)

このフラグは、NOSTRETCH=1 のスレーブモードで、オーバーラン/アンダーランエラーが発生したときに、ハードウェアによってセットされます。OVRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 9 ARLO : アービトレーション喪失

このフラグは、アービトレーション喪失の場合に、ハードウェアによってセットされます。ARLOCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 8 BERR : バスエラー

このフラグは、ペリフェラルが転送に関与しているので、START または STOP コンディションの誤配置が検出されたときに、ハードウェアによってセットされます。このフラグは、スレーブモードのアドレスフェーズではセットされません。BERRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 7 TCR : 転送完了再ロード

このフラグは、RELOAD=1 および NBYTES データが転送されたときに、ハードウェアによってセットされます。NBYTES にゼロ以外の値が書き込まれたときにソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

このフラグは、マスタモード、または SBC ビットがセットされているときのスレーブモードのみ使用されます。

ビット 6 TC : 転送完了 (マスタモード)

このフラグは、RELOAD=0、AUTOEND=0、および NBYTES データが転送されたときに、ハードウェアによってセットされます。START ビットまたは STOP ビットがセットされたときに、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 5 STOPF : STOP 検出フラグ

このフラグは、バス上で STOP コンディションが検出され、ペリフェラルがこの転送に関与しているときに、ハードウェアによってセットされます:

– マスタとして。ただし、STOP コンディションがペリフェラルによって生成される場合。

– または、スレーブとして。ただし、ペリフェラルがこの転送中にアドレス指定されていた場合。

STOPCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 4 NACKF : 非確認応答受信フラグ

このフラグは、バイト送信後に NACK を受信したときに、ハードウェアによってセットされます。NACKCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

ビット 3 ADDR : アドレス一致 (スレーブモード)

このビットは、受信したスレーブアドレスが有効なスレーブアドレスの 1 つに一致したときに、ハードウェアによってセットされます。ADDRCF ビットをセットすることによって、ソフトウェアによってクリアされます。

注: このビットは、 $PE=0$ のとき、ハードウェアによってクリアされます。

- ビット 2 **RXNE** : 受信データレジスタノットエンプティ (レシーバ)
このビットは、受信データが I2Cx_RXDR レジスタにコピーされ、読み出す準備ができたときに、ハードウェアによってセットされます。I2Cx_RXDR が読み出されたときにクリアされます。
注: このビットは、PE=0 のとき、ハードウェアによってクリアされます。
- ビット 1 **TXIS** : 送信割り込みステータス (トランスミッタ)
このビットは、I2Cx_TXDR レジスタが空であり、送信データを I2Cx_TXDR レジスタに書き込む必要があるときに、ハードウェアによってセットされます。次の送信データが I2Cx_TXDR レジスタに書き込まれたときにクリアされます。
このビットは、NOSTRETCH=1 のときのみ、ソフトウェアによって 1 を書き込んで、TXIS イベントを生成することができます (TXIE=1 の場合に割り込み、または TXDMAEN=1 の場合に DMA リクエスト)。
注: このビットは、PE=0 のとき、ハードウェアによってクリアされます。
- ビット 0 **TXE** : 送信データレジスタエンプティ (トランスミッタ)
このビットは、I2Cx_TXDR レジスタが空のときに、ハードウェアによってセットされます。次の送信データが I2Cx_TXDR レジスタに書き込まれたときにクリアされます。
このビットは、ソフトウェアによって 1 を書き込んで、送信データレジスタ I2Cx_TXDR をフラッシュできます。
注: このビットは、PE=0 のとき、ハードウェアによってセットされます。

28.7.8 割り込みクリアレジスタ (I2Cx_ICR)

アドレスオフセット : 0x1C
リセット値 : 0x0000 0000
アクセス : ウェイト状態なし

| | | | | | | | | | | | | | | | |
|------|------|---------|----------|-------|-------|--------|--------|------|------|--------|--------|--------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | ALERTCF | TIMOUTCF | PECCF | OVRCF | ARLOCF | BERRCF | Res. | Res. | STOPCF | NACKCF | ADDRCF | Res. | Res. | Res. |
| | | w | w | w | w | w | w | | | w | w | w | | | |

- ビット 31:14 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 13 **ALERTCF** : アラートフラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの ALERT フラグがクリアされます。
注: SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。
- ビット 12 **TIMOUTCF** : タイムアウト検出フラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの TIMEOUT フラグがクリアされます。
注: SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。
- ビット 11 **PECCF** : PEC エラーフラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの PECERR フラグがクリアされます。
注: SMBus 機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 28.3 : I2C の実装を参照してください。

- ビット 10 **OVRCF** : オーバーラン／アンダーランフラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの OVR フラグがクリアされます。
- ビット 9 **ARLOCF** : アービトレーション喪失フラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの ARLO フラグがクリアされます。
- ビット 8 **BERRCF** : バスエラーフラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの BERRF フラグがクリアされます。
- ビット 7:6 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 5 **STOPCF** : STOP 検出フラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの STOPF フラグがクリアされます。
- ビット 4 **NACKCF** : 非確認応答フラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの ACKF フラグがクリアされます。
- ビット 3 **ADDRCF** : アドレス一致フラグクリア
このビットに 1 を書き込むと、I2Cx_ISR レジスタの ADDR フラグがクリアされます。このビットに 1 を書き込むと、I2Cx_ISR レジスタの START ビットもクリアされます。
- ビット 2:0 予約済みであり、リセット値のままにしておかなければなりません。

28.7.9 PEC レジスタ (I2Cx_PECR)

アドレスオフセット : 0x20
リセット値 : 0x0000 0000
アクセス : ウェイト状態なし

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

- ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 7:0 **PEC[7:0]** パケットエラーチェックレジスタ
PECEN=1 のとき、このフィールドは内部 PEC を含みます。
PEC は、PE=0 のとき、ハードウェアによってクリアされます。

注 : **SMBus 機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって強制的に 0x00000000 に設定されます。セクション 28.3 : I2C の実装 を参照してください。**

28.7.10 受信データレジスタ (I2Cx_RXDR)

アドレスオフセット : 0x24
リセット値 : 0x0000 0000
アクセス : ウェイト状態なし

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

ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。
ビット 7:0 **RXDATA[7:0]** 8 ビット受信データ
I²C バスから受信したデータバイト。

28.7.11 送信データレジスタ (I2Cx_TXDR)

アドレスオフセット : 0x28
リセット値 : 0x0000 0000
アクセス : ウェイト状態なし

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

ビット 31:8 予約済みであり、リセット値のままにしておかなければなりません。
ビット 7:0 **TXDATA[7:0]** 8 ビット送信データ
I²C バスに送信されるデータバイト。
注 : これらのビットは、TXE=1 のときのみ書き込むことができます。

28.7.12 I2C レジスタマップ

次の表に、I2C のレジスタマップとリセット値を示します。

表 114. I2C レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | |
|-----------|---------------|------------|------|------|------|----------------|---------|---------|-------------|--------------|---------|--------|-------------|------|-------|-----------|-----------|----------|---------|---------|----------|----------------|-----------------|----------|-------------|-------|------|--------|--------|--------|--------|--------|--------|------|------|----|
| 0x0 | I2Cx_CR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PECEN | ALERTEN | SMBDEN | SMBHEN | GCEN | WUPEN | NOSTRETCH | SBC | RXDMAEN | TXDMAEN | Res. | ANFOFF | DNF[3:0] | | | | ERRIE | | | | TCIE | STOPIE | NACKIE | ADDRIE | RXIE | TXIE | PE |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| 0x4 | I2Cx_CR2 | Res. | Res. | Res. | Res. | Res. | PECBYTE | AUTOEND | RELOAD | NBYTES[7:0] | | | | | | | NACK | STOP | START | HEAD10R | ADD10 | RD_WRN | SADD[9:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| 0x8 | I2Cx_OAR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OA1EN | Res. | Res. | Res. | Res. | OA1MODE | OA1[9:0] | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0xC | I2Cx_OAR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | OA2EN | Res. | Res. | Res. | Res. | OA2MSK [2:0] | OA2[7:1] | | | | | Res. | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x10 | I2Cx_TIMINGR | PRESC[3:0] | | | | Res. | Res. | Res. | SCLDEL[3:0] | | | | SDADEL[3:0] | | | | SCLH[7:0] | | | | | SCLL[7:0] | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x14 | I2Cx_TIMEOUTR | TEXTEN | Res. | Res. | Res. | TIMEOUTB[11:0] | | | | | | | | | | | | TIMOUTEN | Res. | Res. | TIDLE | TIMEOUTA[11:0] | | | | | | | | | | | | | | |
| | リセット値 | 0 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x18 | I2Cx_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADDCODE[6:0] | | | | | | | DIR | BUSY | Res. | ALERT | TIMEOUT | PECERR | OVR | ARLO | BERR | TCR | TC | STOPF | NACKF | ADDR | RXNE | TXIS | TXE | | | |
| | リセット値 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | | |
| 0x1C | I2Cx_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ALERTCF | TIMOUTCF | PECFCF | OVRFCF | ARLOCF | BERRCF | Res. | Res. | STOPCF | NACKCF | ADDRCF | Res. | Res. | Res. | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | | | | | | |
| 0x20 | I2Cx_PECR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PEC[7:0] | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x24 | I2Cx_RXDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RXDATA[7:0] | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x28 | I2Cx_TXDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXDATA[7:0] | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |



レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

29 USART (Universal synchronous asynchronous receiver transmitter)

29.1 概要

USART (Universal Synchronous Asynchronous Receiver Transmitter) を使用すると、業界標準の NRZ 非同期シリアルデータフォーマットを必要とする外部機器と柔軟に全二重データ交換を行うことができます。USART は、フラクショナルボーレートジェネレータを使用して、非常に広範囲のボーレートを提供します。

USART は、同期式単方向通信と半二重単線通信をサポートします。また、LIN (Local Interconnect Network)、スマートカードプロトコル、および IrDA (Infrared Data Association) SIR ENDEC 仕様、および モデム動作 (CTS/RTS) もサポートします。マルチプロセッサ通信もサポートします。

マルチバッファ設定で DMA (直接メモリアクセス) を使用することによって、高速データ通信が可能です。

29.2 USART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット (マーク/スペース)
- 16 倍または 8 倍に設定可能なオーバーサンプリング方式によって、速度とクロック誤差の間の柔軟性を実現
- 最大 4 Mbit/s (クロック周波数が 32 MHz のとき) まで共通にプログラム可能な送信及び受信ボーレートと 8 倍のオーバーサンプリング
- デュアルクロックドメイン可能
- UART 機能と STOP モードからのウェイクアップ
- PCLK 再プログラミングから独立した便利なボーレートプログラミング
- 自動ボーレート検出
- プログラム可能なデータワード長 (7 または 8 または 9 ビット)
- データ順序をプログラム可能 (MSB ファーストまたは LSB ファースト シフト)
- 設定可能なストップビット (1 または 2 個のストップビット)
- 同期通信のための同期モードとクロック出力
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して専用 SRAM にバッファリング
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御

- 通信制御／エラー検出フラグ
- パリティ制御：
 - － パリティビットの送信
 - － 受信したデータバイトのパリティ検査
- 14 の割り込みソース（フラグ付き）
- マルチプロセッサ通信
アドレスが一致しない場合、USART はミュートモードに入ります。
- ミュートモードからのウェイクアップ（アイドルライン検出またはアドレスマーク検出による）

29.3 USART の拡張機能

- LIN マスタの同期ブレーク送信機能と LIN スレーブのブレーク検出機能
 - － USART が LIN 用にハードウェア設定されている場合、13 ビットのブレーク生成と 10/11 ビットのブレーク検出
- 通常モードで 3/16 ビット期間をサポートする IrDA SIR エンコーダデコーダ
- スマートカードモード
 - － ISO/IEC 7816-3 標準で定義されているスマートカードの T=0 および T=1 非同期プロトコルをサポート
 - － スマートカード動作に 1.5 個のストップビット
- Modbus 通信のサポート
 - － タイムアウト機能
 - － CR/LF キャラクタ認識

29.4 USART の実装

表 115. STM32L0x3 USART の機能

| USART のモード／機能 ⁽¹⁾ | USART1/2 | LPUART1 |
|----------------------------------|---------------|---------|
| モデムのハードウェアフロー制御 | X | X |
| DMA を使用した連続通信 | X | X |
| マルチプロセッサ通信 | X | X |
| 同期モード | X | - |
| スマートカードモード | X | - |
| 単線半二重通信 | X | X |
| Ir SIR ENDEC ブロック | X | - |
| LIN モード | X | - |
| デュアルクロックドメインと STOP モードからのウェイクアップ | X | X |
| レシーバタイムアウト割り込み | X | - |
| Modbus 通信 | X | - |
| 自動ボーレート検出 | X | - |
| ドライバインペブル | X | X |
| USART データ長 | 7、8、および 9 ビット | |

1. X = サポート。

29.5 USART の機能詳細

USART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力 (RX) と送信データ出力 (TX) です。

- **RX** : 受信データ入力。
これはシリアルデータ入力です。データ復旧にはオーバーサンプリング技術が使用され、有効な入力データとノイズを区別しています。
- **TX** : 送信データ出力。
トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線 およびスマートカードモードでは、この I/O はデータの送受信に使用されます。

シリアルデータは、通常の USART モードでは、これらのピンを通じて送受信されます。フレームは、以下で構成されます。

- 送受信前のアイドルライン
- スタートビット
- データワード (7、8、または 9 ビット) LSBファースト
- フレームの完了を示す 1 個、1.5 個、2 個のストップビット
- USART インタフェースはボーレートジェネレータを使用
- ステータスレジスタ (USARTx_ISR)
- 受信および送信データレジスタ (USARTx_RDR、USARTx_TDR)
- ボーレートレジスタ (USARTx_BRR)
- ガードタイムレジスタ (USARTx_GTPR) - スマートカードモードの場合

各ビットの定義については、[セクション 29.7 : USART レジスタ \(730 ページ\)](#) を参照してください。

同期モードおよびスマートカードモードでインタフェースするには、以下のピンが必要です。

- **SCLK** : クロック出力。このピンは、SPI マスタモードに対応する同期送信用のトランスミッタデータクロックを出力します (スタートビットとストップビットのクロックパルスはなく、ソフトウェアオプションで最後のデータビットのクロックパルスを送信します)。これと並行して、RX でデータを同期受信できます。これを使用して、シフトレジスタを持つペリフェラル (例 : LCD ドライバ) を制御できます。クロックの位相と極性は、ソフトウェアでプログラム可能です。スマートカードモードでは、SCLK 出力はスマートカードにクロックを供給できます。

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **nCTS** : Clear To Send は、ハイレベルのとき、現在の転送の終わりにデータ送信をブロックします。
- **nRTS** : Request To Send は、ローレベルのとき、USART がデータを受信する準備ができたことを示します。

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE** : Driver Enable は、外部トランシーバの送信モードを有効にします。

注 : **DE と nRTS は同じピンを共有します。**

[illegible]

- この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や損害などに付きましては如何なる責任にも負いません。

29.5.1 USART キャラクタの説明

ワード長は、USARTx_CR1 レジスタの M[1:0] ビットをプログラムすることによって、7、8、または 9 ビットとして選択できます (図 231 を参照)。

- 7 ビットのキャラクタ長 : M[1:0] = 10
- 8 ビットのキャラクタ長 : M[1:0] = 00
- 9 ビットのキャラクタ長 : M[1:0] = 01

注 : 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート (0x7F および 0x55 フレーム検出) はサポートされません。7 ビットモードは、一部の USART でのみサポートされます。

デフォルト設定では、信号 (TX または RX) はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

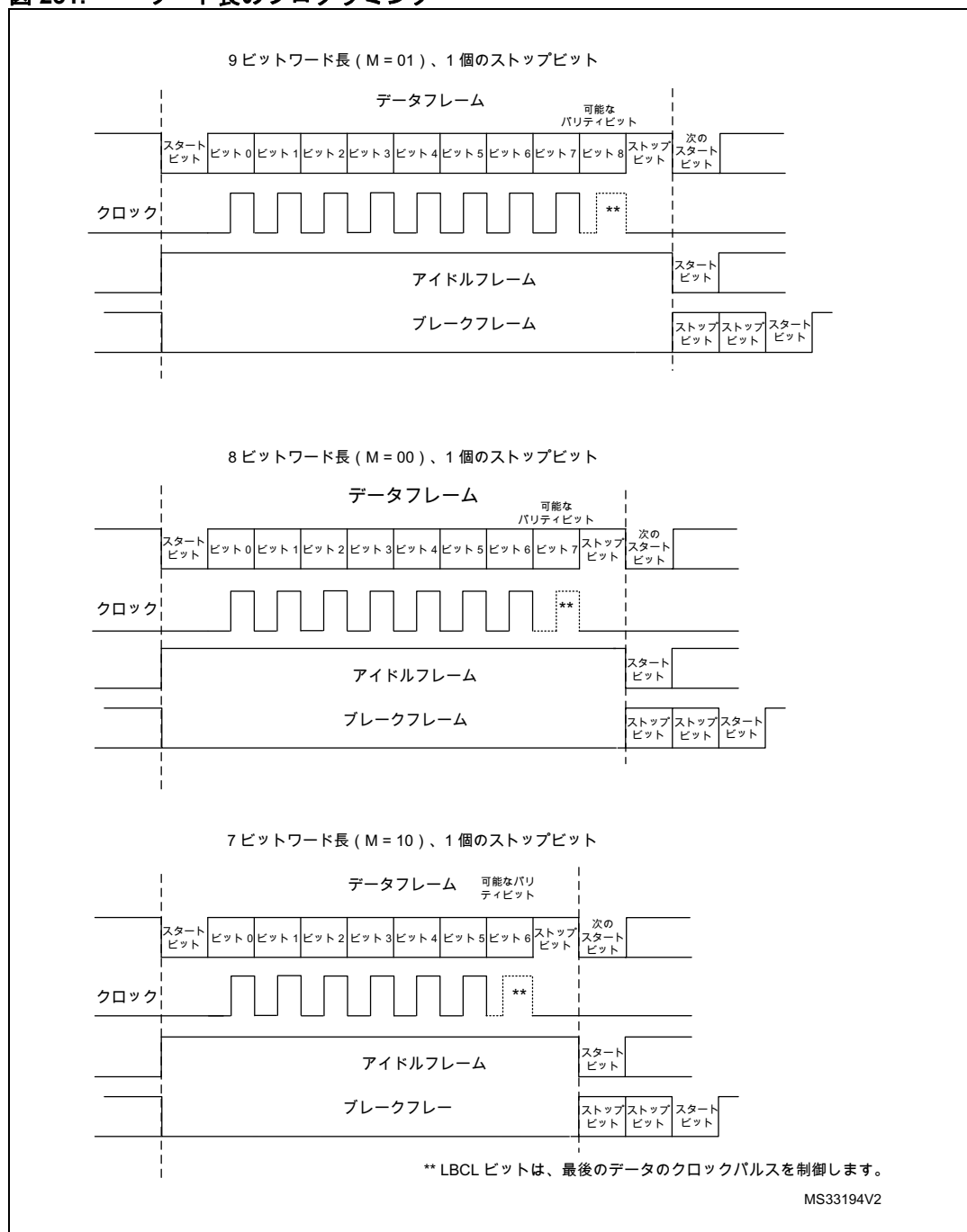
アイドルキャラクタは、1 フレーム分の「1」であると解釈されます (「1」の数にはストップビットの数も含まれます)。

ブレークキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレークフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は、共通のボーレートジェネレータによって駆動され、それぞれのクロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 231. ワード長のプログラミング



29.5.2 トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7 または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット (TE) をセットする必要があります。送信シフトレジスタ内のデータは TX ピンで出力され、対応するクロックパルスは SCLK ピンで出力されます。

キャラクタ送信

USART 送信時、データはLSBファースト（デフォルト設定）で TX ピンにシフトアウトされます。このモードでは、USARTx_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ（TDR）で構成されます（図 230 を参照）。

各キャラクタの前には、スタートビット（1 ビット周期、論理レベルロー）があります。キャラクタは、設定可能な数のストップビットで終端されます。

USART では、1、1.5、および 2 個のストップビットがサポートされます。

注：

送信データを USARTx_TDR に書き込む前に、TE ビットをセットする必要があります。

データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中のデータは失われます。

TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

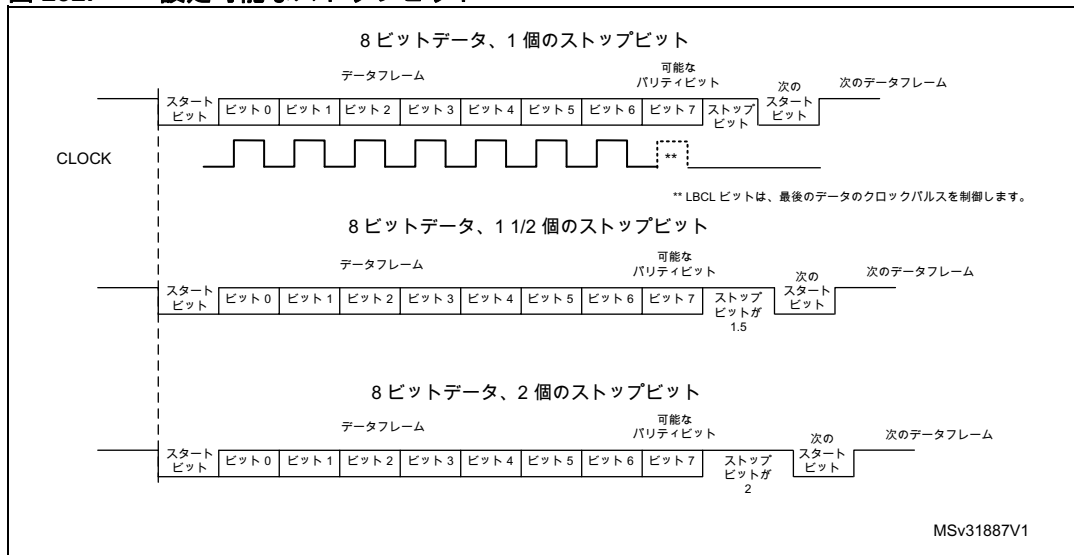
各キャラクタとともに送信されるストップビットの数は、制御レジスタ 2 のビット 13、12 でプログラミングできます。

- ストップビット1個：ストップビット数のデフォルト値です。
- ストップビット2個：通常の USART モード、単線モード、およびモデムモードでサポートされます。
- **ストップビット1、5個**：スマートカードモードで使用されます。

アイドルフレームの送信にはストップビットが含まれます。

ブ레이크送信は、10 個のロービット（M[1:0] = 00 のとき）または 11 個のロービット（M[1:0] = 01 のとき）または 9 個のロービット（M[1:0] = 10 のとき）の後に 2 個のストップビットが続きます（図 232 を参照）。長いブ레이크（9/10/11 個のロービットを超える長さのブ레이크）を送信することはできません。

図 232. 設定可能なストップビット



キャラクタ送信手順

1. ワード長を定義するには、USARTx_CR1 の M ビットをプログラムします。
2. USARTx_BRR レジスタを使用して、目的のボーレートを選択します。
3. USARTx_CR2 レジスタでストップビットの数をプログラミングします。
4. USARTx_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合は、USARTx_CR3 レジスタの DMA 有効 (DMAT) を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. USARTx_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを USARTx_TDR レジスタに書き込みます (これによって TXE ビットがクリアされます)。シングルバッファの場合、送信される各データにこれを繰り返します。
8. USARTx_TDR レジスタに最後のデータを書き込んだ後、TC=1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。これは、たとえば最後の送信内容の破壊を避けるために、USART が無効になったり HALT モードに入ったりする場合に必要です。

1 バイト通信

TXE ビットは、常に送信データレジスタへの書き込みによってクリアされます。

TXE ビットは、ハードウェアによってセットされ、次を示します。

- データは USARTx_TDR レジスタからシフトレジスタへ移動され、データ送信が開始しています。
- USARTx_TDR レジスタは空です。
- 次のデータを、前のデータに上書きせずに、USARTx_TDR レジスタに書き込みます。

TXEIE ビットがセットされている場合、このフラグは割り込みを生成します。

送信が行われている場合、USARTx_TDR レジスタへの書き込み命令によってデータが TDR レジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

送信が行われていないときには、USARTx_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

フレームが送信され (ストップビットの後)、TXE ビットがセットされると、TC ビットはハイレベルになります。USARTx_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

USARTx_TDR レジスタに最後のデータを書き込んだ後は、USART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=1 になるまで待つ必要があります (図 233 : 送信時の TC/TXE の動作を参照)。

The diagram illustrates the timing of USART transmission across three frames (フレーム 1, フレーム 2, フレーム 3). The TX line shows the serial data output. The TXE flag indicates when the TDR register is empty and ready for the next byte. The USART_TDR register contains the data bytes F1, F2, and F3. The TC flag indicates the end of the current frame.

TXE Flag Behavior:

- At the start of each frame, TXE is set by hardware.
- When the last byte of a frame is transmitted, TXE is cleared by software.
- At the start of the next frame, TXE is set by hardware.

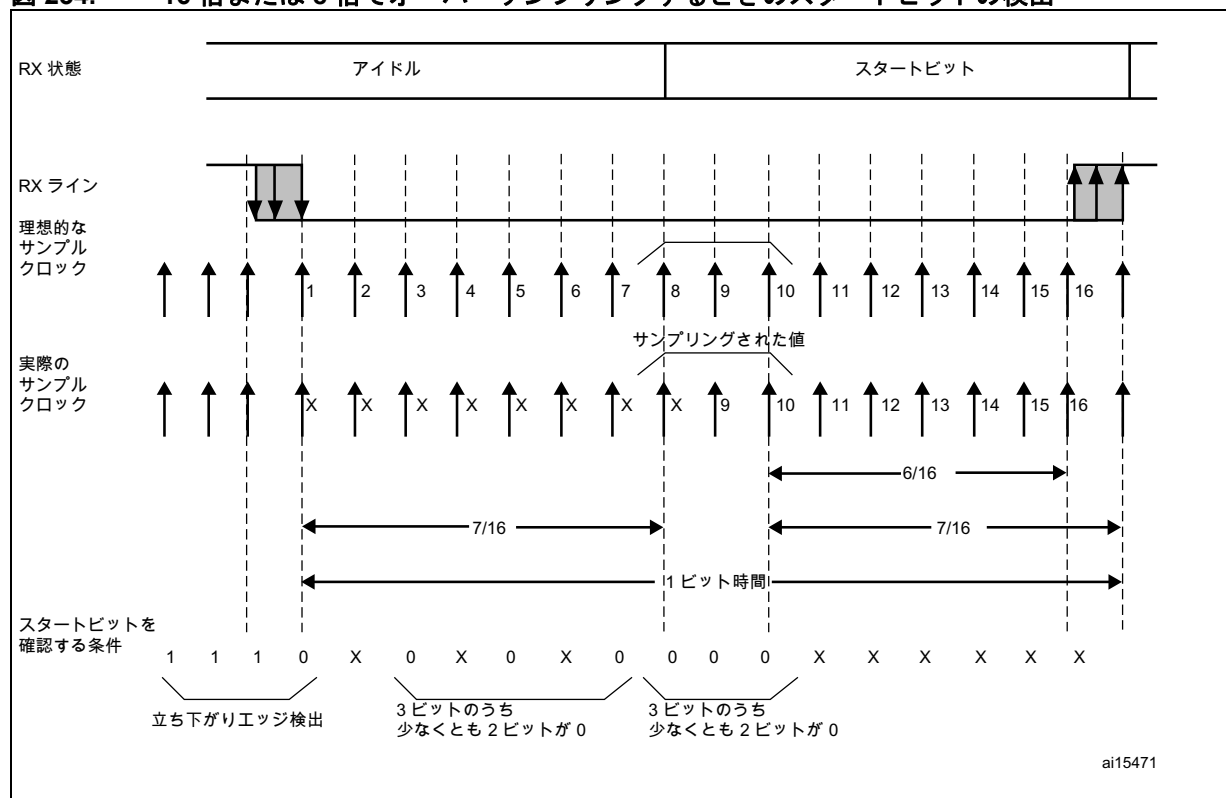
TC Flag Behavior:

- At the end of each frame, TC is set by hardware.
- When the next frame starts, TC is cleared by hardware.

Software Actions:

- When TXE is set, software can write the next byte to TDR.
- When TC is set, software knows the current frame is complete.

図 234. 16 倍または 8 倍でオーバーサンプリングするときのスタートビットの検出



注： シーケンスが完了していない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻って（フラグはセットされません）、立ち下がりエッジを待ちます。

サンプリングされた 3 つのビットが 0 の場合（第 3、第 5、および第 7 ビットでの最初のサンプリングで 3 つのビットが 0 であり、第 8、第 9、および第 10 ビットでの 2 回目のサンプリングでも 3 つのビットが 0）、スタートビットが確認されます（RXNE フラグがセットされ、RXNEIE=1 の場合は割り込みが生成されます）。

次の場合、スタートビットは確認されますが（RXNE フラグがセットされ、RXNEIE=1 の場合は割り込みが生成されます）、NF ノイズフラグがセットされます。

a. 両方のサンプリングについて、サンプリングされた 3 ビットのうちの 2 ビットが 0 の場合（第 3、第 5、および第 7 ビットのサンプリングと、第 8、第 9、および第 10 ビットのサンプリング）。

または

b. いずれか一方のサンプリングで（第 3、第 5、および第 7 ビットでのサンプリングまたは第 8、第 9、および第 10 ビットでのサンプリング）、3 つのビットのうち 2 つが 0 の場合。

a. または b. のいずれの条件も満たされない場合、スタートビットの検出は中止され、レシーバはアイドル状態に戻ります（フラグはセットされません）。

キャラクタの受信

USART の受信時には、データは RX ピンを通じて LSB ファーストでシフトインされます。このモードでは、USARTx_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ (RDR) で構成されます。

キャラクタ受信手順

1. ワード長を定義するには、USARTx_CR1 の M ビットをプログラムします。
2. ボーレートレジスタ USARTx_BRR を使用して、目的のボーレートを選択します。
3. USARTx_CR2 レジスタでストップビットの数をプログラミングします。
4. USARTx_CR1 レジスタの UE ビットに 1 を書き込んで、USART を有効にします。
5. マルチバッファ通信を行う場合には、USARTx_CR3 レジスタの DMA 有効 (DMAR) を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. USARTx_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読み出し可能です (関連するエラーフラグも同様です)。
- RXNEIE ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。RXNE によって PE フラグもセットできます。
- マルチバッファでは、RXNE はバイト受信ごとにセットされ、受信データレジスタの DMA 読み出しによってクリアされます。
- シングルバッファモードでは、RXNE ビットのクリアは、ソフトウェアによる USARTx_RDR レジスタの読み出しによって行われます。RXNE フラグは、USARTx_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。

ブレークキャラクタ

ブレークキャラクタを受信すると、USART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出されると、受信データキャラクタの場合と同じ手順が行われ、IDLEIE ビットがセットされていた場合は割り込みが生成されます。

オーバーランエラー

RXNE ビットがリセットされていないときにキャラクタを受信すると、オーバーランエラーが発生します。RXNE ビットがクリアされない限り、データをシフトレジスタから RDR レジスタに転送することはできません。

RXNE フラグは、バイトを受信するたびにセットされます。次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

- ORE ビットがセットされます。
- RDR の内容は失われません。USARTx_RDR の読み出しが行われると、前のデータが使用できません。
- シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
- RXNEIE ビットがセットされているか、EIE ビットがセットされている場合、割り込みが生成されます。
- ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： *ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。この場合、次の 2 つの可能性がありえます。*

- RXNE=1 の場合、最後の有効なデータが受信レジスタ RDR に格納され、読み出すことができます。

- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものがないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースと適切なオーバーサンプリング方式の選択

クロックソースの選択は、クロック制御システムを通じて行われます（[セクション 7：リセットおよびクロック制御 \(RCC\)](#) を参照）。クロックソースは、USART を有効にする（UE ビットをセットする）前に選ぶ必要があります。

クロックソースの選択は、2 つの基準に従って行われる必要があります。

- USART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、 f_{CK} です。

デュアルクロックドメインと STOP モードからのウェイクアップ機能がサポートされる時には、クロックソースは次のソースの 1 つにできます。PCLK（デフォルト）、LSE、HSI16、または SYSCLK。そうでない場合、USART のクロックソースは PCLK です。

LSE または HSI16 をクロックソースとして選択すると、USART は MCU が低電力モードのときにデータを受信できます。受信データとウェイクアップモードの選択に応じて、USART より必要なときに MCU をウェイクアップし、ソフトウェアが USARTx_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、USART 通信を可能にするためには、システムを有効にする必要があります。

通信速度の範囲（特に最大通信速度）もクロックソースによって決まります。

レシーバは有効な受信データとノイズを区別して、データを復旧するユーザ設定可能なさまざまなオーバーサンプリング技術を実装しています（同期モードの場合を除く）。これにより、最大通信速度とノイズノッククロック精度の耐性の間でトレードオフができます。

オーバーサンプリング方式は、USARTx_CR1 レジスタの OVER8 ビットをプログラムすることによって選択でき、ボーレートクロックの 16 倍または 8 倍にすることができます (図 235 および図 236)。

アプリケーションに応じて、

- 高速 (最大 $f_{CK}/8$) を達成するには、8 倍 (OVER8=1) のオーバーサンプリングを選択します。この場合、クロック偏差に対するレシーバの最大許容誤差は軽減されます (セクション 29.5.5: クロック偏差に対する USART レシーバの許容誤差 (706 ページ) を参照)。
- クロック偏差に対するレシーバの許容誤差を増やすには、16 倍 (OVER8=0) のオーバーサンプリングを選択します。この場合、最大速度は $f_{CK}/16$ に制限されます。 f_{CK} はクロックソース周波数です。

論理レベルの評価方法を選択するには、USARTx_CR3 レジスタの ONEBIT ビットをプログラミングします。これには 2 つのオプションがあります。

- 受信されたビットの中央にある 3 つのサンプルの多数決。この場合、多数決に使用された 3 つのサンプルが等しくないとき、NF ビットがセットされます。
- 受信されたビットの中央にある 1 つのサンプル。

アプリケーションに応じて、

- ノイズの多い環境で操作するときは 3 つのサンプルの多数決方式 (ONEBIT=0) を選択します。そしてノイズが検出された場合 (図 116 を参照) は、サンプリング中にグリッチが発生していることとなりますので、そのデータを除去します。
- ラインがノイズフリーであるときは、1 つのサンプルによる方式 (ONEBIT=1) を選択し、クロック偏差に対するレシーバの許容誤差を増やします (セクション 29.5.5: クロック偏差に対する USART レシーバの許容誤差 (706 ページ) を参照)。この場合、NF ビットはセットされません。

フレーム内でノイズが検出された場合 :

- RXNE ビットの立ち上がりエッジで、NF ビットがセットされます。
- 無効なデータがシフトレジスタから USARTx_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNE ビットと同時に立ち上がります。マルチバッファ通信の場合、USARTx_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

NF ビットは、ICR レジスタの NFCF ビットをセットすることによってリセットされます。

注 : 8 倍のオーバーサンプリングは、スマートカード、IrDA、および LIN の各モードでは使用できません。これらのモードでは、OVER8 ビットはハードウェアによって 0 に固定されています。

図 235. データサンプリング (16 倍のオーバーサンプリング)

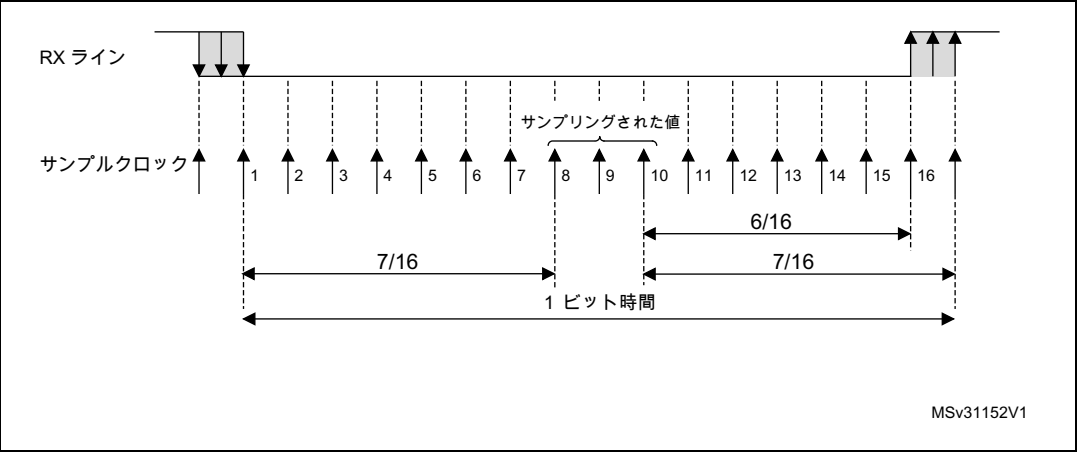


図 236. データサンプリング (8 倍のオーバーサンプリング)

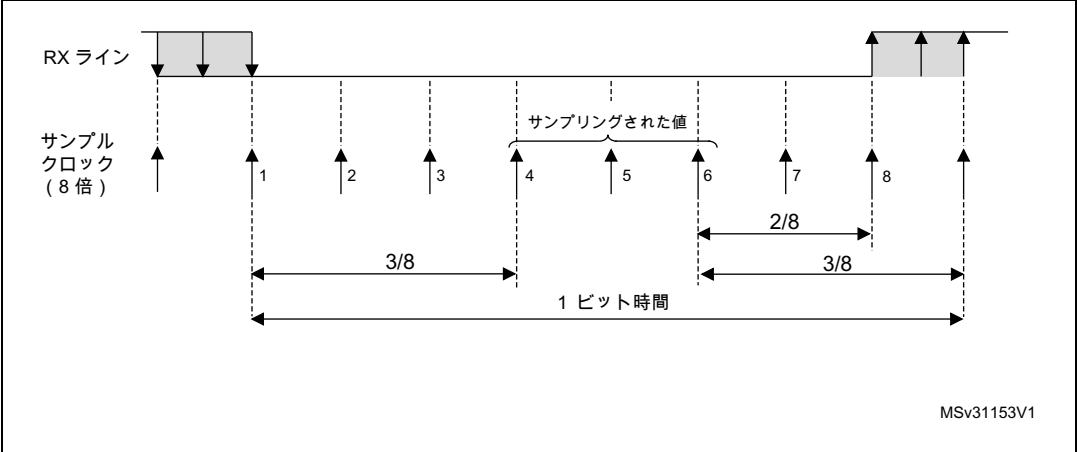


表 116. サンプリングされたデータからのノイズ検出

| サンプリングされた値 | NE ステータス | 受信ビットの値 |
|------------|----------|---------|
| 000 | 0 | 0 |
| 001 | 1 | 0 |
| 010 | 1 | 0 |
| 011 | 1 | 1 |
| 100 | 1 | 0 |
| 101 | 1 | 1 |
| 110 | 1 | 1 |
| 111 | 0 | 1 |

フレーミングエラー

フレーミングエラーは、次の場合に検出されます。

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから USARTx_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNE ビットと同時に立ち上がります。マルチバッファ通信の場合、USARTx_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

USARTx_ICR レジスタの FECF に 1 を書き込むことによって、FE ビットがリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、制御レジスタ 2 の制御ビットを通じて設定でき、通常モードでは 1 または 2、スマートカードモードでは 1.5 にできます。

- **1 個のストップビット**：ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- **1.5 個のストップビット (スマートカードモード)**：スマートカードモードでの送信時は、デバイスは、データが正しく送信されたことをチェックする必要があります。したがって、レシーバブロックを有効にし (USARTx_CR1 レジスタの RE=1)、ストップビットをチェックして、スマートカードがパリティエラーを検出したかどうかをテストする必要があります。パリティエラーが発生した場合、スマートカードはサンプリング時のデータ信号を強制的にローレベルにします (これは、フレーミングエラーとしてフラグされる NACK 信号です)。その後、1.5 個のストップビットの最後に、RXNE ビットとともに FE フラグがセットされます。ストップビット 1.5 個のサンプリングは、16 番目、17 番目、および 18 番目のサンプルで (ストップビットの開始から 1 ボークロック周期後に) 行われます。1.5 個のストップビットは、2 つの部分に分解できます。すなわち、何も起こらない 0.5 ボークロック周期と、途中でサンプリングが行われる通常の 1 ストップビット周期です。詳細については、[セクション 29.5.13 : スマートカードモード \(717 ページ\)](#) を参照してください。
- **2 個のストップビット**：ストップビット 2 個のサンプリングは、最初のストップビットの 8 番目、9 番目、および 10 番目のサンプルで行われます。最初のストップビットでフレーミングエラーが検出された場合、フレーミングエラーフラグがセットされます。2 番目のストップビットでは、フレーミングエラーの検査は行われません。RXNE フラグは、最初のストップビットの最後でセットされます。

29.5.4 ボーレート生成

レシーバとトランスミッタ (Rx と Tx) のボーレートは、USARTx_BRR レジスタでプログラムされたものと同じ値に設定されます。

Equation 1 : 標準 USART のボーレート (SPI モードを含む) (OVER8 = 0 または 1)

16 倍のオーバーサンプリングの場合、等式は次のとおりです。

$$\text{Tx/Rx ボー} = \frac{f_{\text{CK}}}{\text{USARTDIV}}$$

8 倍のオーバーサンプリングの場合、等式は次のとおりです。

$$\text{Tx/Rx ボー} = \frac{2 \times f_{\text{CK}}}{\text{USARTDIV}}$$

: スマートカード、LIN、および IrDA モードのボーレート (OVER8 = 0)

$$\text{Tx/Rx ボー} = \frac{f_{\text{CK}}}{\text{USARTDIV}}$$

USARTDIV は、符号なしの固定小数点数であり、USARTx_BRR レジスタでコード化されます。

- OVER8 = 0 のとき、BRR = USARTDIV です。
- OVER8 = 1 のとき、
 - BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。
 - BRR[3] は、クリアされたままにする必要があります。
 - BRR[15:4] = USARTDIV[15:4]

注 : USARTx_BRR レジスタへの書き込み操作の後、ボーカウンタはボーレジスタの新しい値に更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。

16 倍または 8 倍のオーバーサンプリングの場合、USARTDIV は 16d 以上である必要があります。

USARTx_BRR レジスタの値から USARTDIV を得る方法

Example 1

$f_{\text{CK}} = 8 \text{ MHz}$ で 9600 ボーを得るには、

- 16 倍のオーバーサンプリングの場合 :
USARTDIV = 8 000 000/9600
BRR = USARTDIV = 833d = 0341h
- 8 倍のオーバーサンプリングの場合 :
USARTDIV = 2 * 8 000 000/9600
USARTDIV = 1666,66 (1667d = 683h)
BRR[3:0] = 3h << 1 = 1h
BRR = 0x681

Example 2

- $f_{CK} = 32 \text{ MHz}$ で 921.6 Kbaud を得るには、
- 16 倍のオーバーサンプリングの場合 :
USARTDIV = 32 000 000/921 600
BRR = USARTDIV = 35d = 23h
 - 8 倍のオーバーサンプリングの場合 :
USARTDIV = 2 * 32 000 000/921 600
USARTDIV = 70d = 46h
BRR[3:0] = USARTDIV[3:0] >> 1 = 6h >> 1 = 3h
BRR = 0x43

表 117. 16 倍または 8 倍のオーバーサンプリングの両方の場合の $f_{CK} = 32 \text{ MHz}$ でのプログラムされたボーレートの誤差計算⁽¹⁾

| ボーレート | | 16 倍のオーバーサンプリング (OVER8=0) | | | 8 倍のオーバーサンプリング (OVER8=1) | | |
|-------|------------|---------------------------|--------|--|--------------------------|--------|------|
| S.No | 目標 | 実際 | BRR | 誤差 (%) = (計算値 - 目標値) ボーレート / 目標 ボーレート | 実際 | BRR | 誤差 |
| 1 | 2.4 KBps | 2.4 KBps | 0x3415 | 0 | 2.4 KBps | 0x6825 | 0 |
| 2 | 9.6 KBps | 9.6 KBps | 0xD05 | 0 | 9.6 KBps | 0x1A05 | 0 |
| 3 | 19.2 KBps | 19.19 KBps | 0x683 | 0.02 | 19.2 KBps | 0xD02 | 0 |
| 4 | 38.4 KBps | 38.41 KBps | 0x341 | 0.04 | 38.39 KBps | 0x681 | 0.02 |
| 5 | 57.6 KBps | 57.55 KBps | 0x22C | 0.08 | 57.6 KBps | 0x453 | 0 |
| 6 | 115.2 KBps | 115.1 KBps | 0x116 | 0.08 | 115.11 KBps | 0x226 | 0.08 |
| 7 | 230.4 KBps | 230.21 KBps | 0x8B | 0.08 | 230.21 KBps | 0x113 | 0.08 |
| 8 | 460.8 KBps | 463.76 KBps | 0x045 | 0.64 | 460.06 KBps | 0x85 | 0.08 |
| 9 | 921.6 KBps | 914.28 KBps | 0x23 | 0.79 | 927.5 KBps | 0x42 | 0.79 |
| 10 | 2 MBps | 2 MBps | 0x10 | 0 | 2 MBps | 0x20 | 0 |
| 12 | 4MBps | 4MBps | 該当なし | 該当なし | 4MBps | 0x10 | 0 |

1. CPU クロック周波数が低いほど、ボーレートの精度も低下しますボーレートの達成可能な上限は、これらのデータによって決定できます。

29.5.5 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバは、クロックシステムの合計偏差が USART レシーバの許容誤差未満の場合のみ、正しく動作します。合計偏差の要因は、次のとおりです。

- DTRA : トランスミッタの誤差に起因する偏差 (トランスミッタのローカルオシレータの偏差も含みます)
- DQUANT : レシーバのボーレート量子化に起因する誤差
- DREC : レシーバのローカルオシレータの偏差
- DTCL : 送信ラインに起因する偏差 (一般には、ローからハイへの遷移タイミングとハイからローへの遷移タイミングの間に非対称性をもたらす可能性のあるトランシーバに起因)

$$DTRA + DQUANT + DREC + DTCL + DWU < \text{USART receiver's 許容誤差}$$

ここで、

DWU は、STOP モードからのウェイクアップが使用されたときのサンプリングポイントの偏差によるエラーです。

$t_{WU\text{STOP}}$ は、STOP モードからのウェイクアップ時間であり、製品のデータシートで指定されています。

USART レシーバは、次の選択に応じて、表 118 および 表 119 で指定された最大許容偏差まで、データを正しく受信できます。

- USARTx_CR1 レジスタの M ビットで定義された 9、10、または 11 ビットキャラクタ長
- USARTx_CR1 レジスタの OVER8 ビットによって定義された 8 倍または 16 倍のオーバーサンプリング
- USARTx_BRR レジスタのビット BRR[3:0] が 0000 に等しいかどうか。
- データのサンプリングに 1 ビットを使用するか 3 ビットを使用するか (USARTx_CR3 レジスタの ONEBIT ビットの値に依存)。

表 118. BRR [3:0] = 0000 のときの USART レシーバの許容誤差

| M ビット | OVER8 ビット = 0 | | OVER8 ビット = 1 | |
|-------|---------------|----------|---------------|----------|
| | ONEBIT=0 | ONEBIT=1 | ONEBIT=0 | ONEBIT=1 |
| 00 | 3.75% | 4.375% | 2.50% | 3.75% |
| 01 | 3.41% | 3.97% | 2.27% | 3.41% |
| 10 | 4.16% | 4.86% | 2.77% | 4.16% |

表 119. BRR[3:0] が 0000 でないときの USART レシーバの許容誤差

| M ビット | OVER8 ビット = 0 | | OVER8 ビット = 1 | |
|-------|---------------|----------|---------------|----------|
| | ONEBIT=0 | ONEBIT=1 | ONEBIT=0 | ONEBIT=1 |
| 00 | 3.33% | 3.88% | 2% | 3% |
| 01 | 3.03% | 3.53% | 1.82% | 2.73% |
| 10 | 3.7% | 4.31% | 2.22% | 3.33% |

注 : 表 118 および表 119 で指定されたデータは、M ビット = 00 のとき、受信フレームに正確に 10 ビット時間のアイドルフレームが含まれる特殊なケースで、若干異なることがあります (M ビット = 01 のときには 11 ビット時間、または M ビット = 10 のときには 9 ビット時間)。

29.5.6 自動ボーレート検出

USART は、1 キャラクタの受信に基づいて、USARTx_BRR レジスタ値を検出し、自動的にセットすることができます。自動ボーレート検出は、2 つの状況で便利です。

- システムの通信速度が事前に分かっていないとき。
- システムが比較的低い精度のクロックソースを使用している場合、このメカニズムによって、クロック偏差を測定しなくても、正しいボーレートを求めることができます。

クロックソース周波数は、予期される通信速度と互換性がなければなりません（16 倍のオーバーサンプリングのとき、ボーレートは $f_{CK}/65535$ から $f_{CK}/16$ までの範囲内です。8 倍のオーバーサンプリングのとき、ボーレートは $f_{CK}/65535$ から $f_{CK}/8$ までの範囲内です）。

自動ボーレート検出を有効にする前に、自動ボーレート検出モードを選択する必要があります。キャラクタパターンに基づいて、いくつかのモード があります。

モードは、USARTx_CR2 レジスタの ABRMOD[1:0] フィールドで選択できます。これらの自動ボーレートモードでは、同期データ受信中にボーレートが数回測定され、各測定値が前回の測定値と比較されます。

以下のモードがあります。

- **モード 0** : 1 のビットで始まるキャラクタ。この場合、USART はスタートビットの時間を測定します（立ち下がリエッジから立ち上がりエッジまで）。
- **モード 1** : 10xx ビットパターンで始まるキャラクタ。この場合、USART はスタートと最初のデータビットの時間を測定します。低速な信号傾斜の場合の精度を高めるために、測定は立ち下がリエッジから立ち上がりエッジまでで行われます。
- **モード 2** : 0x7F キャラクタフレーム (LSB ファーストモードでは 0x7F キャラクタ、または MSB ファーストモードでは 0xFE)。この場合、ボーレートは、最初はスタートビット (BRs) の終了時に更新され、次にビット 6 の終了時に更新されます（立ち下がリエッジから立ち上がりエッジまで行われた測定に基づいて、BR6）。ビット 0 からビット 6 までが BRs でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。
- **モード 3** : 0x55 キャラクタフレーム。この場合、ボーレートは、最初はスタートビット (BRs) の終了時に更新され、次にビット 0 の終了時に更新され（立ち下がリエッジから立ち上がりエッジまで行われた測定に基づいて、BR0）、最後にビット 6 (BR6) の終了時に更新されます。ビット 0 は BRs でサンプリングされ、ビット 1 からビット 6 までは BR0 でサンプリングされ、キャラクタの残りのビットは BR6 でサンプリングされます。

並行して、RX ラインの中間遷移ごとに別のチェックが行われます。RX の遷移がレシーバと十分に同期していない場合はエラーが生成されます（レシーバは、ビット 0 で計算されたボーレートに基づきます）。

自動ボーレート検出を有効にする前に、ゼロ以外のボーレート値を書き込むことによって、USARTx_BRR レジスタを初期化する必要があります。

自動ボーレート検出を有効にするには、USARTx_CR2 レジスタの ABREN ビットをセットします。USART は、RX ラインの最初のキャラクタを待ちます。自動ボーレート動作の完了は、USARTx_ISR レジスタの ABRF フラグのセットによって示されます。ラインにノイズが多い場合、正しいボーレート検出を保証できません。この場合、BRR 値が破損して、ABRE エラーフラグがセットされることがあります。また、通信速度が自動ボーレート検出の範囲と互換性がない場合にも発生します（ビット時間が 16 から 65536 までのクロック周期でなく（16 倍のオーバーサンプリング）、8 から 65536 までのクロック周期でない（8 倍のオーバーサンプリング））。

RXNE 割り込みは、操作の終了を知らせます。

その後いつでも、ABRF フラグをリセットすることによって（0 を書き込むことによって）、自動ボーレート検出を再開できます。

注： *自動ボーレート操作中に USART が無効化された場合 (UE=0)、BRR 値が破損することがあります。*



29.5.7 マルチプロセッサ通信

マルチプロセッサ通信では、次のビットをクリアされた状態に保つ必要があります。

- USART_CR2 レジスタの LINEN ビット
- USART_CR3 レジスタの HDSEL、IREN、および SCEN ビット

USART とのマルチプロセッサ通信が可能です (ネットワーク内で複数の USART を接続して)。たとえば、1 つの USART をマスタとして、その TX 出力を別の USART の RX 入力に接続することができます。別の USART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する USART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、USARTx_CR1 レジスタの MME ビットをセットする必要があります。

ミュートモードでは、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- USARTx_ISR レジスタの RWU ビットは 1 にセットされます。特定の条件下では、USARTx_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

USART は、USARTx_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出 (WAKE=0)

MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、USART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、USARTx_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を [図 237](#) に示します。

図 237. アイドルライン検出を使用したミュートモード



注： IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません (RWU はセットされません)。

ラインが IDLE のときに USART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます (1 キャラクタフレームの受信後だけでなく)。

4 ビット/7 ビットアドレスマーク検出 (WAKE=1)

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDm7 ビットを使用して行われます。この 4 ビット/7 ビットワードは、レシーバによって、USARTx_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： 7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス (ADD[5:0] および ADD[7:0]) で行われます。

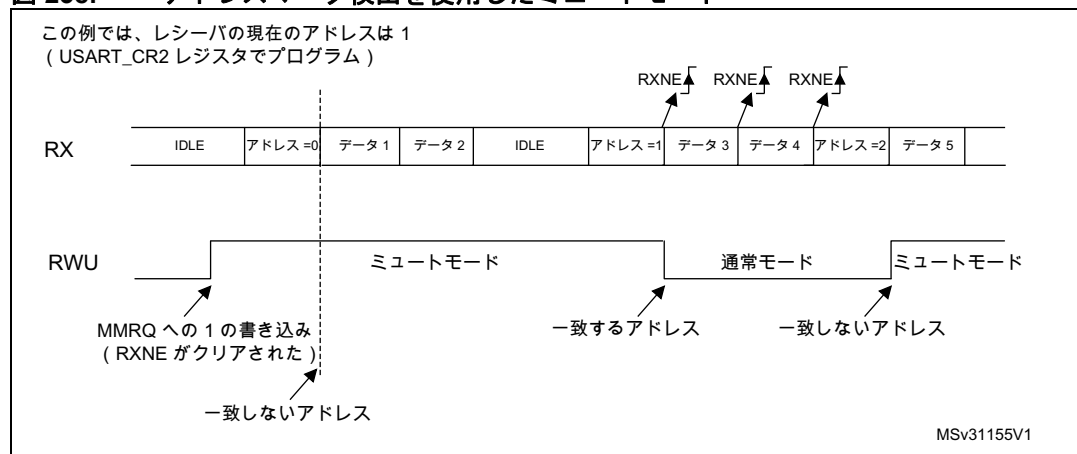
プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

MMRQ ビットに 1 が書き込まれたときにも、USART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、USART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE ビットがセットされます。

アドレスマーク検出を使用したミュートモードの動作例を [図 238](#) に示します。

図 238. アドレスマーク検出を使用したミュートモード



29.5.8 Modbus 通信

USART は、Modbus/RTU および Modbus/ASCII プロトコルの実装に対する基本的サポートを備えています。Modbus/RTU は、半二重のブロック転送プロトコルです。プロトコルの制御部分（アドレス認識、ブロック整合性制御、およびコマンド解釈）は、ソフトウェアで実装する必要があります。

USART は、ソフトウェアに負荷をかけず、他のリソースを使用せずに、ブロック検出の終了に対する基本的サポートを提供します。

Modbus/RTU

このモードでは、1 つのブロックの終了は 2 キャラクタ時間を超える「サイレンス」（アイドルライン）によって認識されます。この機能は、プログラム可能なタイムアウト機能を通じて実装されます。

タイムアウト機能と割り込みは、USARTx_CR2 レジスタの RTOEN ビットと、USARTx_CR1 レジスタの RTOIE を通じて有効にする必要があります。2 キャラクタ時間に対応する値（たとえば、22 x ビット時間）を RTO レジスタでプログラムする必要があります。最後のストップビットの受信後、この時間にわたって受信ラインがアイドルのときには、割り込みが生成されて、現在のブロック受信が完了したことをソフトウェアに知らせます。

Modbus/ASCII

このモードでは、ブロックの終了は特定の（CR/LF）キャラクタシーケンスによって認識されます。USART は、キャラクタ一致機能を使用して、このメカニズムを管理します。

ADD[7:0] フィールドで LF ASCII コードをプログラムし、キャラクタ一致割り込みを有効にすることによって（CMIE=1）、LF が受信されたときにソフトウェアに通知し、ソフトウェアは DMA バッファの CR/LF をチェックできます。

29.5.9 パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティ検査）を有効にするには、USARTx_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な USART フレームフォーマットを表 120 に示します。

表 120. フレームフォーマット

| M ビット | PCE ビット | USART フレーム ⁽¹⁾ |
|-------|---------|---------------------------|
| 00 | 0 | SB 8 ビットデータ STB |
| 00 | 1 | SB 7 ビットデータ PB STB |
| 01 | 0 | SB 9 ビットデータ STB |
| 01 | 1 | SB 8 ビットデータ PB STB |
| 10 | 0 | SB 7 ビットデータ STB |
| 10 | 1 | SB 6 ビットデータ PB STB |

1. 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、9、8、または 7 番目）。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット (M ビットの 値に応じて) とパリティビットから構成されるフレーム内で「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合 (USARTx_CR1 の PS ビット = 0) でも、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット (M ビットの 値に応じて) とパリティビットから構成されるフレーム内で「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合 (USARTx_CR1 の PS ビット = 1) でも、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、USARTx_ISR レジスタの PE フラグがセットされ、USARTx_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、USARTx_ICR レジスタの PECF に 1 を書き込むことによってクリアされます。

送信中のパリティ生成

USARTx_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます (偶数パリティが選択された場合 (PS=0) は偶数個の「1」、奇数パリティが選択された場合 (PS=1) は奇数個の「1」)。

29.5.10 LIN (Local Interconnection Network) モード

このセクションは、LIN モードがサポートされるときにのみ適用されます。[セクション 29.4: USART の実装 \(690 ページ\)](#)を参照してください。

LIN モードを選択するには、USARTx_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットをクリアされた状態に保つ必要があります。

- USARTx_CR2 レジスタの CLKEN
- USARTx_CR3 レジスタの STOP[1:0]、SCEN、HDSEL、および IREN

LIN 送信

LIN マスタ送信の場合、[セクション 29.5.2: トランスミッタ](#)に説明されている手順を適用する必要があります。通常の USART 送信と同じですが、次のような違いがあります。

- M ビットをクリアして、8 ビットワード長を設定します。
- LIN モードに入るには、LINEN ビットをセットします。この場合、SBKRQ ビットをセットすると、13 個の「0」ビットがブレイクキャラクタとして送信されます。その後、値「1」の 2 ビットが送信され、次の START 検出が可能になります。

LIN 受信

LIN モードが有効になると、ブレーク検出回路が有効になります。この検出は、通常の USART レシーバとは完全に独立しています。アイドル状態時やフレームの処理中には、発生たびにブレークが検出できます。

レシーバが有効になると (USARTx_CR1 レジスタの RE=1)、RX 入力の START 信号を探します。スタートビットの検出方法は、ブレークキャラクタやデータの検索方法と同じです。スタートビットが検出された後、データの場合とまったく同様に次のビットがサンプリングされます (8、9、および 10 番目のサンプル)。10 個 (USARTx_CR2 レジスタの LBDL=0) または 11 個 (USARTx_CR2 レジスタの LBDL=1) の連続したビットが "0" として検出され、その後にデリミタキャラクタが続く場合、USARTx_ISR レジスタの LBDF フラグがセットされます。LBDIE ビットが 1 の場合、割り込みが生成されます。ブレークを確認する前に、RX ラインがハイレベルに戻ったことを知らせるデリミタが検査されます。

この 10 または 11 が発生する前に "1" がサンプリングされた場合、ブレーク検出回路は現在の検出をキャンセルし、再びスタートビットを検索します。

LIN モードが無効にされた場合 (LINEN=0)、レシーバは、ブレーク検出を考慮することなく、通常の USART として機能し続けます。

LIN モードが有効にされた場合 (LINEN=1)、フレーミングエラーが発生 (つまり、ブレークフレームの場合と同様に、"0" の位置でストップビットを検出) すると、レシーバは停止し、ブレーク検出回路が "1" (ブレークワードが完全でなかった場合) またはデリミタキャラクタ (ブレークが検出された場合) を受信するまで停止状態を維持します。

ブレーク検出回路ステートマシンの動作とブレークフラグを [図 239 : LIN モードでのブレーク検出 \(11 ビットブレーク長、LBDL=1\) \(713 ページ\)](#) に示します。

ブレークフレームの例を [図 240 : LIN モードでのブレーク検出とフレーミングエラー検出 \(714 ページ\)](#) に示します。

図 239. LIN モードでのブレイク検出 (11 ビットブレイク長、LBDF=1)

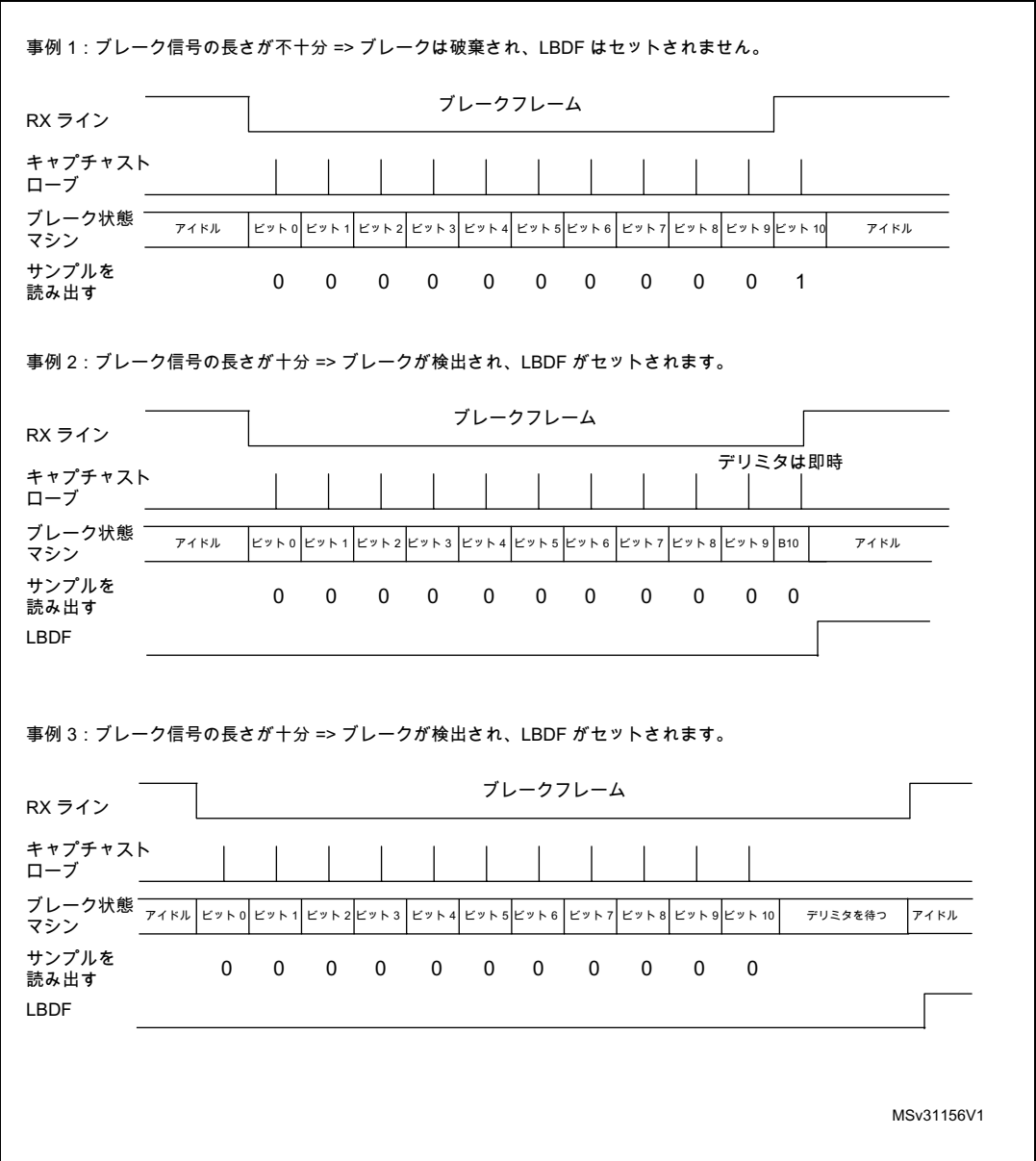
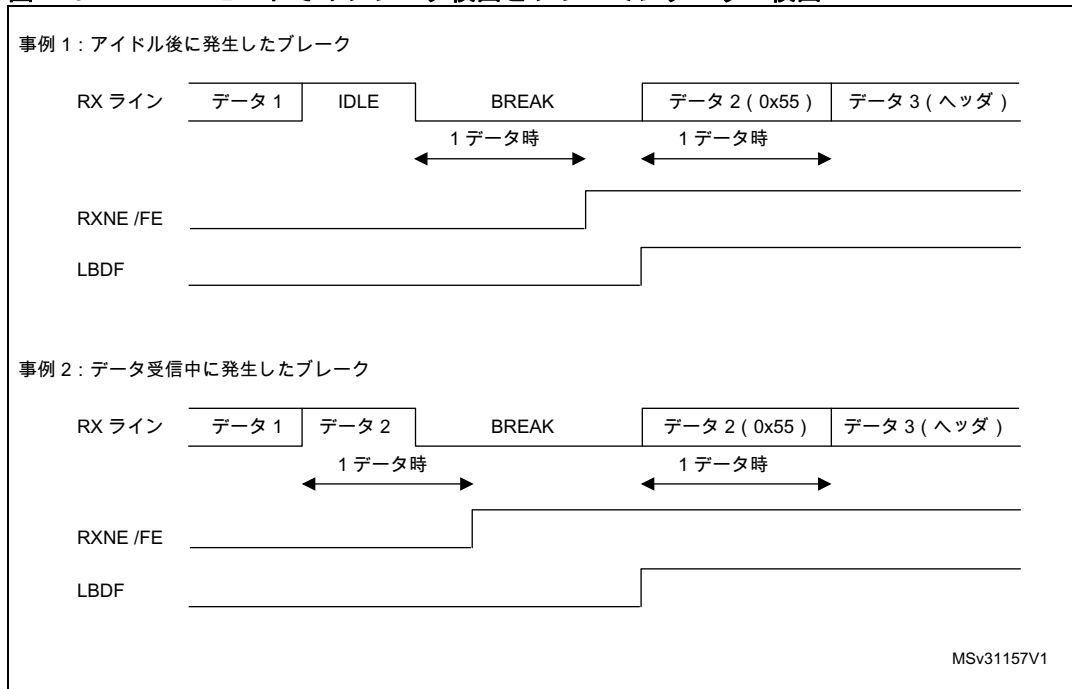


図 240. LIN モードでのブ레이크検出とフレーミングエラー検出



29.5.11 USART 同期モード

同期モードを選択するには、USARTx_CR2 レジスタの CLKEN ビットに 1 を書き込みます。同期モードでは、次のビットをクリアされた状態に保つ必要があります。

- USARTx_CR2 レジスタの LINEN ビット
- USARTx_CR3 レジスタの SCEN、HDSEL、および IREN ビット

このモードでは、USART を使用して、双方向同期シリアル通信をマスタモードで制御できます。SCLK ピンは USART トランスミッタクロックの出力です。スタートビットとストップビットの処理中には、SCLK ピンにクロックパルスは送信されません。USARTx_CR2 レジスタの LBCL ビットの状態によっては、有効な最後のデータビット（アドレスマーク）の処理中にクロックパルスが生成されることもあります。USARTx_CR2 レジスタの CPOL ビットは、クロック極性を選択するために使用され、USARTx_CR2 レジスタの CPHA ビットは、外部クロックの位相を選択するために使用されます（図 241、図 242、および図 243 を参照）。

アイドル時、プリアンブル処理時、およびブ레이크送信時には、外部 SCLK クロックは起動されません。

同期モードでは、USART トランスミッタは非同期モードの場合とまったく同じように機能します。しかし、CPOL と CPHA に基づいて SCLK が TX と同期すると、TX 上のデータが同期します。

このモードでは、USART レシーバは非同期モードの場合とは異なる動作をします。RE=1 の場合、データはオーバーサンプリングなしで、SCLK (CPOL と CPHA に応じて立上りまたは立下りエッジ) でサンプリングされます。ボーレート (1/16 ビット時間) に応じたセットアップ時間とホールド時間を守る必要があります。

注： SCLK ピンは TX ピンとともに機能します。したがって、クロックが供給されるのは、トランスミッタが有効であり (TE=1)、データが送信中 (データレジスタ USARTx_TDR への書き込み) の場合に限られます。つまり、データ送信を行わずに同期データを受信することはできません。

USART が無効にされたときには (UE=0)、クロックパルスが正常に機能するように、LBCL、CPOL、および CPHA ビットを選択する必要があります。

図 241. USART の同期送信の例

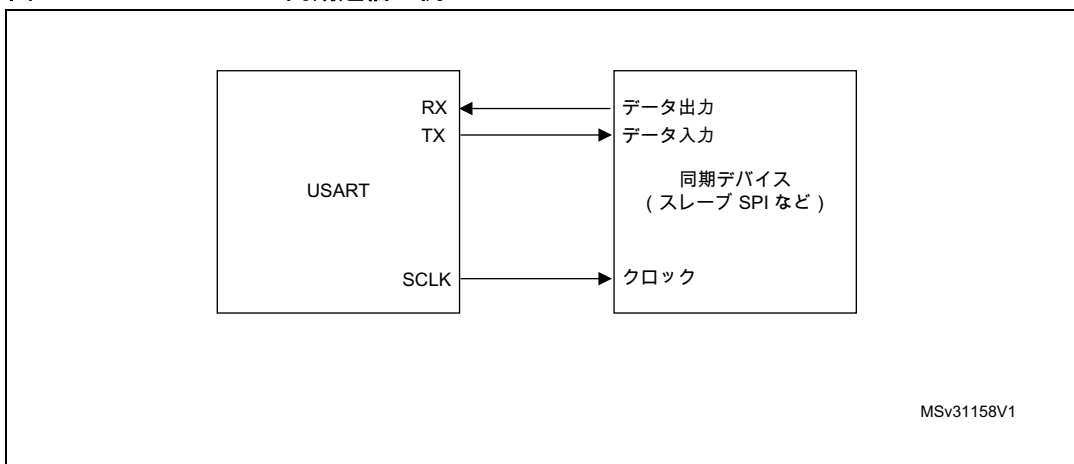


図 242. USART データクロックタイミング図 (M ビット = 00)

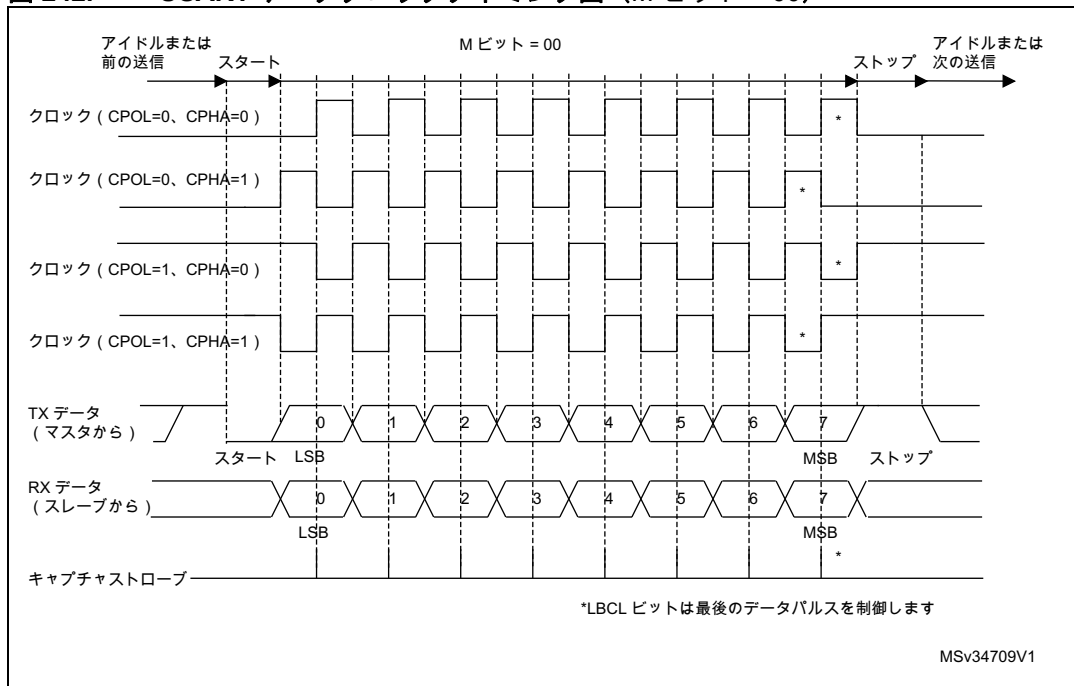


図 243. USART データクロックタイミング図 (M ビット = 01)

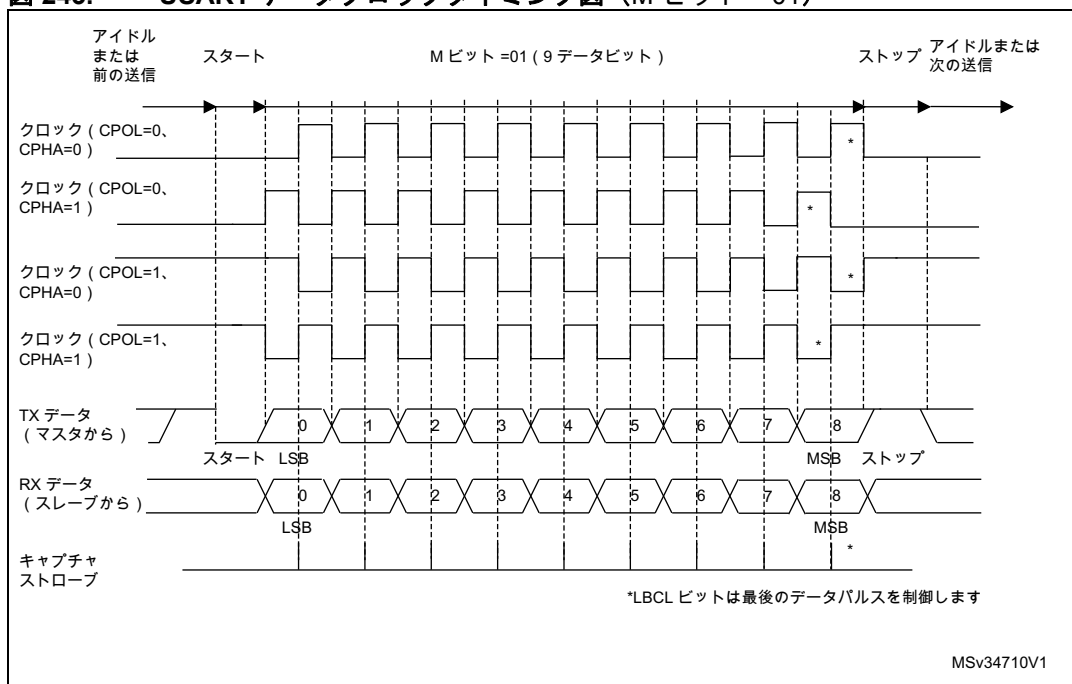
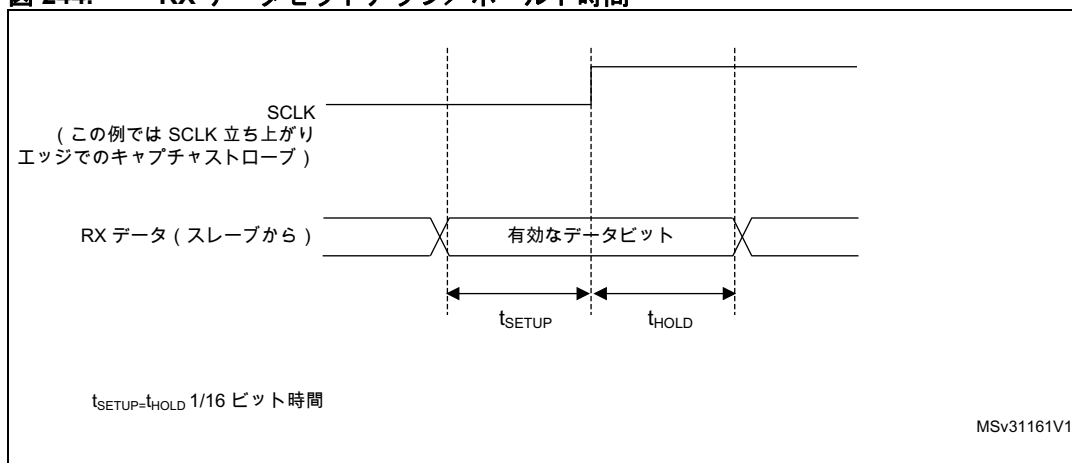


図 244. RX データセットアップ/ホールド時間



注: スマートカードモードでは SCLK の機能が異なります。詳細については、[セクション 29.5.13: スマートカードモード](#) を参照してください。

29.5.12 単線半二重通信

単線半二重モードを選択するには、USARTx_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- USARTx_CR2 レジスタの LINEN および CLKEN ビット
- USARTx_CR3 レジスタの SCEN および IREN ビット

USART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、USART_CR3 レジスタの制御ビット HDSELで行います。

HDSEL ビットに 1 が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の USART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

29.5.13 スマートカードモード

このセクションは、スマートカードモードがサポートされるときにのみ適用されます。 [セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

スマートカードモードを選択するには、USARTx_CR3 レジスタの SCEN ビットをセットします。スマートカードモードでは、次のビットをクリアされた状態に保つ必要があります。

- USARTx_CR2 レジスタの LINEN ビット
- USARTx_CR3 レジスタの HDSEL および IREN ビット

さらに、スマートカードにクロックを供給するために CLKEN ビットをセットすることもできます。

スマートカードインタフェースは、ISO 7816-3 標準で定義された非同期プロトコルスマートカードをサポートするように設計されています。T=0（キャラクタモード）と T=1（ブロックモード）の両方がサポートされます。

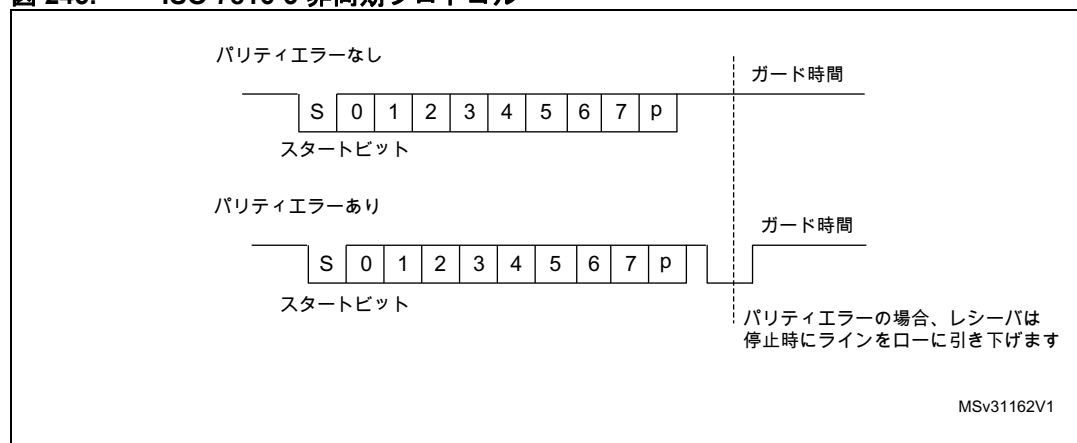
USART は次のように設定してください。

- 8 ビット+パリティ：M ビット=01 および USARTx_CR1 レジスタの PCE=1
- 1.5 個のストップビット：USARTx_CR2 レジスタの STOP=11

T=0（キャラクタ）モードでは、パリティエラーはガードタイム中の各キャラクタの終わりに示されます。

[図 245](#) に、パリティエラーの有無によるデータラインの状況の変化の例を示します。

図 245. ISO 7816-3 非同期プロトコル



スマートカードと接続されると、USART の TX 出力は、やはりスマートカードによって駆動される双方向ラインを駆動します。TX ピンは、オープンドレインとして設定される必要があります。

スマートカードモードは、単線半二重通信プロトコルを実装します。

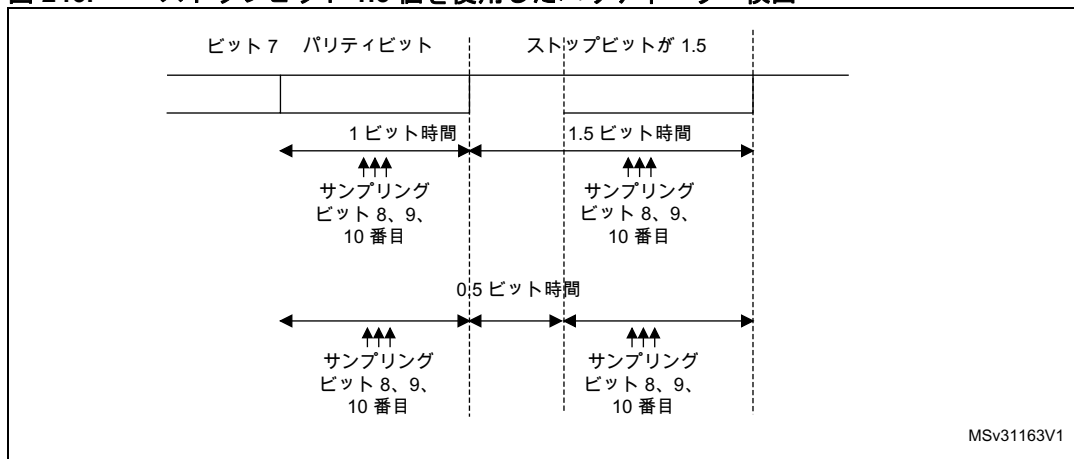
- 送信シフトレジスタからのデータの送信は、少なくとも 1/2 ボークロックの遅れが保証されます。通常動作では、満杯の送信シフトレジスタは、次のボークロックエッジでシフト動作を開始します。スマートカードモードでは、この送信は、保証された 1/2 ボークロック分だけさらに遅れます。
- 送信時、スマートカードがパリティエラーを検出した場合には、ラインをローに駆動することによって (NACK)、この条件を USART に知らせます。この NACK 信号 (1 ボークロックの間、送信ラインをローに引き下げ) は、1.5 個のストップビットが組み込まれたトランスミッタ側にフレーミングエラーを引き起こします。USART は、プロトコルに従って、データの自動再送信を処理できます。再試行回数は、SCARCNT ビットフィールドでプログラムされます。プログラムされた再試行回数後も USART が NACK を受信し続けた場合は、送信を停止して、エラーをフレーミングエラーとして通知します。TXE ビットは、USARTx_RQR レジスタの TXFRQ ビットを使用してクリアできます。
- 送信時のスマートカード自動再試行: USART による NACK の検出と反復キャラクタのスタートビットの間に 2.5 ボー周期の遅延が挿入されます。最後の反復キャラクタの受信終了時、ただちに TC ビットがセットされます (ガードタイムはありません)。ソフトウェアで再び繰り返した場合は、規格によって指定されている 2 ボー周期以上を確保する必要があります。
- 1.5 個のストップビット周期でプログラムされたフレームの受信時にパリティエラーが検出された場合、受信フレームの完了後 1 ボークロック周期間、送信ラインがローに引き下げられます。これは、USART に送信されたデータが正しく受信されなかったことをスマートカードに知らせるためです。NACK 制御ビットがセットされている場合、パリティエラーはレシーバによって "NACK" され、そうでない場合、NACK は送信されません (T=1 モードで使用されます)。受信したキャラクタにエラーがあった場合、RXNE/受信 DMA リクエストは有効になりません。プロトコルの仕様に従って、スマートカードは同じキャラクタを再送信する必要があります。SCARCNT ビットフィールドで指定された最大試行回数後も、受信したキャラクタにエラーがあった場合、USART は NACK の送信を停止して、エラーをパリティエラーとして通知します。
- 受信時のスマートカード自動再試行: USART がカードを NACK したが、カードがキャラクタを繰り返さなかった場合、BUSY フラグはセットされたままです。
- 送信時、USART は 2 つの連続するキャラクタの間にガードタイム (ガードタイムレジスタでプログラム) を挿入します。ガードタイムは前のキャラクタのストップビット後に測定されるので、GT[7:0] レジスタを目的の CGT (7816-3 仕様で定義されている Character Guard Time) から 12 (1 キャラクタの時間) を引いた値にプログラムする必要があります。
- TC フラグのアサーションは、ガードタイムレジスタをプログラムすることによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの NACK によってトランスミッタ端でフレーミングエラーが検出された場合、トランスミッタの受信ブロックは、この NACK をスタートビットとして検出しません。ISO プロトコルによれば、受信される NACK の期間は 1 または 2 ボークロック周期です。
- レシーバ側では、パリティエラーが検出されて NACK が送信された場合、レシーバはこの NACK をスタートビットとして検出しません。

注： スマートカードモードでは、ブレークキャラクタは意味を持ちません。フレーミングエラー発生時のデータ 0x00 は、ブレークではなくデータとして処理されます。

TE ビットをトグルするとき、アイドルフレームは送信されません。アイドルフレームは、他の設定では定義されますが、ISO プロトコルでは定義されていません。

図 246 に、USART による NACK 信号のサンプリング方法を示します。この例では、USART はデータを送信中であり、ストップビットが 1.5 個組み込まれています。データと NACK 信号の整合性を検査するために、USART のレシーバ部が有効にされます。

図 246. ストップビット 1.5 個を使用したパリティエラー検出



USART は、SCLK 出力を通じてスマートカードにクロックを供給できます。スマートカードモードでは、SCLK は通信に関係せず、5 ビットのプリスケアラを通じて単に内部のペリフェラル入カクロックから取得されます。この分周比は、プリスケアラレジスタ USARTx_ で設定されます。SCLK 周波数は、 $f_{CK}/2$ から $f_{CK}/62$ までの範囲でプログラムでき、 f_{CK} はペリフェラル入カクロックです。

ブロックモード (T=1)

T=1 (ブロック) モードでは、パリティエラー送信は、UART_CR3 レジスタの NACK ビットをクリアすることによって無効化されます。

ブロックモードでスマートカードからの読み出しをリクエストするときには、ソフトウェアは USART_CR2 レジスタの RTOEN ビットをセットすることによってレシーバタイムアウト機能を有効にし、RTOR レジスタの RTO ビットフィールドを BWT (ブロックウェイトタイム) - 11 の値にプログラムする必要があります。この期間が終了する前にカードからの応答が受信されなかった場合、RTOF フラグがセットされ、タイムアウト割り込みが生成されます (USART_CR1 レジスタの RTOIE ビットがセットされていた場合)。この期間が終了する前に最初のキャラクタが受信された場合は、RXNE 割り込みによって示されます。

注： ブロックモードのスマートカードからの読み出しに USART を DMA モードで使用するときでも、RXNE 割り込みを有効にする必要があります。並行して、DMA は最初の受信バイトの後でのみ有効にする必要があります。

2 つの連続するキャラクタの間で最大ウェイトタイムの自動チェックを行うには、最初のキャラクタの受信後 (RXNE 割り込み)、RTOR レジスタの RTO ビットフィールドを CWT (キャラクタウェイトタイム) - 11 の値にプログラムする必要があります。この時間は、ボータイム単位で表されます。スマートカードが前のキャラクタの終了後、CWT 未満の時間内に新しいキャラクタを送信しなかった場合、USART は RTOF フラグと割り込み (RTOIE ビットがセットされているとき) によって、これをソフトウェアに通知します。

注： RTO カウンタは、STOP = 00、10 の場合、最後のキャラクタの最初のストップビットの終了からカウントを開始します。STOP = 11 の場合、RTO カウンタは STOP ビットの開始から 1 ビット時間のカウントを開始します。スマートカードプロトコルの定義にあるように、BWT/CWT 値は最後のキャラクタの開始 (スタートビット) から定義されます。RTO レジスタは、最後のキャラクタ自体の長さを考慮して、それぞれ BWT -11 または CWT -11 にプログラムする必要があります。

ブロック長カウンタは、USART が受信するすべてのキャラクタをカウントするために使用されます。このカウンタは、USART の送信時 (TXE=0) にリセットされます。ブロックの長さは、スマートカードによってブロックの 3 番目のバイト (プロローグフィールド) で伝えられます。この値を USARTx_RTOR レジスタの BLEN フィールドでプログラムする必要があります。DMA モードを使用するときには、ブロックの開始前に、このレジスタフィールドを最小値 (0x0) にプログラムする必要があります。この値では、4 番目の受信キャラクタの後に割り込みが生成されます。ソフトウェアは LEN フィールド (3 番目のバイト) を読み出す必要があり、その値は受信バッファから読み出される必要があります。

割り込み駆動受信モードでは、ブロックの長さはソフトウェアによって、または BLEN 値をプログラムすることによってチェックできます。ただし、ブロックの開始前に、BLEN の最大値 (0xFF) をプログラムすることができます。実際の値は、3 番目のキャラクタの受信後にプログラムされます。

ブロックが LRC 水平冗長検査 (1 エピローグバイト) を使用している場合は、BLEN=LEN です。ブロックが CRC メカニズム (2 エピローグバイト) を使用している場合は、BLEN=LEN+1 をプログラムする必要があります。合計ブロック長 (プロローグ、エピローグ、および情報フィールドを含む) は、BLEN+4 に等しくなります。ブロックの終わりは EOBFF フラグと割り込み (EOBIE ビットがセットされているとき) によってソフトウェアに通知されます。

ブロック長エラーの場合、ブロックの終わりは RTO 割り込みによって通知されます (キャラクタウェイトタイムオーバーフロー)。

注： エラーチェックコード (LRC/CRC) は、ソフトウェアによって計算/確認されなければなりません。

ダイレクトおよびインバースコンベンション

スマートカードプロトコルは、ダイレクトとインバースの 2 つコンベンションを定義しています。

ダイレクトコンベンションは、LSB ファースト、論理ビットの値 1 がラインの H 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=0、DATAINV=0 (デフォルト値) をプログラムする必要があります。

インバースコンベンションは、MSB ファースト、論理ビットの値 1 が単線の L 状態に対応、および偶数パリティとして定義されています。このコンベンションを使用するためには、制御ビット MSBFIRST=1、DATAINV=1 をプログラムする必要があります。

注： 論理データ値が反転されると (0=H、1=L)、パリティビットも同じように反転されます。

カードのコンベンションを認識するために、カードは最初のキャラクタ TS を ATR (Answer To Reset) フレームの最初のキャラクタとして送信します。TS には、LHHL LLL LLH と LHHL HHH LLH の 2 つのパターンがあります。

- (H) LHHL LLL LLH は、インバースコンベンションをセットアップします。状態 L が値 1 にエンコードされ、モーメント 2 は最上位ビットを含みます (MSB ファースト)。インバースコンベンションによってデコードされると、送受信されたバイトは 3F に等しくなります。
- (H) LHHL HHH LLH は、ダイレクトコンベンションをセットアップします。状態 H が値 1 にエンコードされ、モーメント 2 は最下位ビットを含みます (LSB ファースト)。ダイレクトコンベンションによってデコードされると、送受信されたバイトは 3B に等しくなります。

キャラクタパリティは、2 から 10 までの 9 個のモーメントに 1 にセットされた偶数個のビットがあったときに正しいとみなされます。

USART はカードが使用するコンベンションを知らないので、いずれのパターンであるかを認識して、それに応じて動作できる必要があります。パターン認識はハードウェアでは行われず、ソフトウェアシーケンスによって行われます。さらに、USART がダイレクトコンベンション（デフォルト）で設定され、カードがインバースコンベンションで応答した場合、TS = LHH LLL LLH になり、USART が受信したキャラクタは 03 になり、パリティは奇数です。

したがって、TS パターン認識には 2 つの方式を使用できます。

方式 1

USART は、標準スマートカードモード／ダイレクトコンベンションでプログラムされます。この場合、TS パターンの受信によってパリティエラー割り込みと、カードに対するエラー信号が生成されます。

- パリティエラー割り込みは、カードがダイレクトコンベンションで正しく応答しなかったことをソフトウェアに知らせます。ソフトウェアは、USART をインバースコンベンションで再プログラムします。
- エラー信号に対して、カードは同じ TS キャラクタを再試行し、再プログラムされた USART によって今度は正しく受信されます。

または、パリティエラー割り込みに応答して、ソフトウェアは USART を再プログラムし、カードに対して新しいリセットコマンドを生成してから、TS を再び待つこともできます。

方式 2

USART は 9 ビット／パリティなしモード、ビット反転なしでプログラムされます。このモードでは、次のように 2 つの TS パターンのいずれかを受信します。

(H) LHH LLL LLH = 0x103 -> インバースコンベンションを選択

(H) LHH LLL LLH = 0x103 -> ダイレクトコンベンションを選択

ソフトウェアは受信されたキャラクタをこの 2 つのパターンと照合して、いずれかに一致した場合は、それに応じて、次のキャラクタ受信に備えて USART をプログラムします。

2 つのうちのどちらも認識されなかった場合、ネゴシエーションを再開するためにカードリセットが生成されます。

29.5.14 IrDA SIR ENDEC ブロック

このセクションは、IrDA モードがサポートされるときにのみ適用されます。[セクション 29.4: USART の実装 \(690 ページ\)](#) を参照してください。

IrDA モードを選択するには、USARTx_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットをクリアされた状態に保つ必要があります。

- USARTx_CR2 レジスタの LINEN、STOP、および CLKEN ビット
- USARTx_CR3 レジスタの SCEN および HDSEL ビット

IrDA SIR 物理層は、ロジック 0 を赤外光パルスとして表現する RZI (Return to Zero, Inverted) 変調方式の使用を指定します ([図 247](#) を参照)。

SIR 送信エンコーダは、USART からの NRZ (Non Return to Zero) 送信ビットストリーム出力を変調します。出力パルスストリームは、外部の出力ドライバと赤外線 LED に送信されます。SIR ENDEC の場合、USART は最大 115.2 Kbps のビットレートしかサポートしません。通常モードでは、送信されるパルス幅は、ビット周期の 3/16 と指定されます。

SIR 受信デコーダは、赤外線検出回路からの RZ (Return to Zero) ビットストリームを復調し、受信した NRZ シリアルビットストリームを USART に出力します。デコーダの入力は、アイドル状態の

ノーマルハイレベル（マーク状態）です。送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。デコーダ入力がローレベルのとき、スタートビットが検出されます。

- IrDA は半二重通信プロトコルです。トランスミッタがビジーである場合（USART が IrDA エンコーダにデータを送信しているとき）、IrDA デコーダは IrDA 受信ライン上にあるすべてのデータを無視します。レシーバがビジーである（USART がデコードされたデータを受信している）場合、IrDA は USART から IrDA への TX 上のデータをエンコードしません。データの受信中は、送信データの破壊を防ぐために、送信を避けてください。
- 0 はハイパルスとして送信され、1 は 0 として送信されます。通常モードでは、パルスの幅は、選択されたビット周期の 3/16 と規定されます（図 248 を参照）。
- SIR デコーダは、IrDA 準拠の受信信号を USART 用のビットストリームに変換します。
- SIR 受信ロジックは、ハイ状態を論理値 1 とみなし、ローパルス（パルス）を論理値 0 とみなします。
- 送信エンコーダの出力は、デコーダ入力とは逆の極性になっています。SIR 出力は、アイドル時にロー状態になります。
- IrDA 仕様では、1.41 us より大きなパルスを受け入れる必要があります。受け入れられるパルス幅は、プログラム可能です。レシーバ側のグリッチ検出回路は、PSC 2 周期（PSC は USARTx_GTPR でプログラムされたプリスケール値）より小さな幅のパルスをフィルタします。PSC 1 周期より小さな幅のパルスは常に拒否されますが、1 周期以上 2 周期未満の幅のパルスは受け入れられることも、拒否されることもあります。2 周期より大きな幅のパルスは、パルスとして受け入れられます。PSC=0 のとき、IrDA エンコーダ/デコーダは機能しません。
- レシーバは、低電力トランスミッタと通信できます。
- IrDA モードでは、USARTx_CR2 レジスタのストップビットを「1 ストップビット」に設定する必要があります。

IrDA 低電力モード

トランスミッタ：

低電力モードでは、パルス幅はビット周期の 3/16 に維持されません。代わりに、パルス幅は低電力ボーレート（最小で 1.42 MHz）の 3 倍となります。一般に、この値は 1.8432 MHz（ $1.42 \text{ MHz} < \text{PSC} < 2.12 \text{ MHz}$ ）です。低電力モードのプログラム可能な分周器は、この値を得るためにシステムクロックを分周します。

レシーバ：

低電力モードでの受信は、通常モードでの受信と同様です。グリッチ検出の場合、USART は 1 PSC よりも短いパルスを破棄する必要があります。有効なローレベルは、その期間が IrDA 低電力ボーレート（USARTx_GTPR の PSC 値）の 2 周期分を超える場合にのみ受け入れられます。

注： PSC 2 周期未満 1 周期以上の幅のパルスは、拒否されることも、拒否されないこともあります。

レシーバのセットアップ時間は、ソフトウェアで管理してください。IrDA 物理層仕様では、送信と受信の間に最小 10 ms の遅延を指定しています（IrDA は半二重プロトコルです）。

図 247. IrDA SIR ENDEC- ブロック図

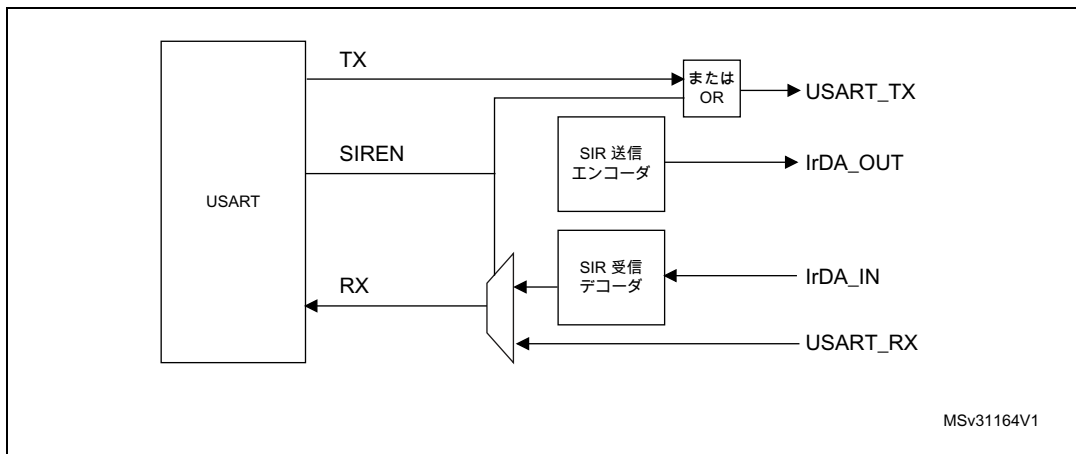
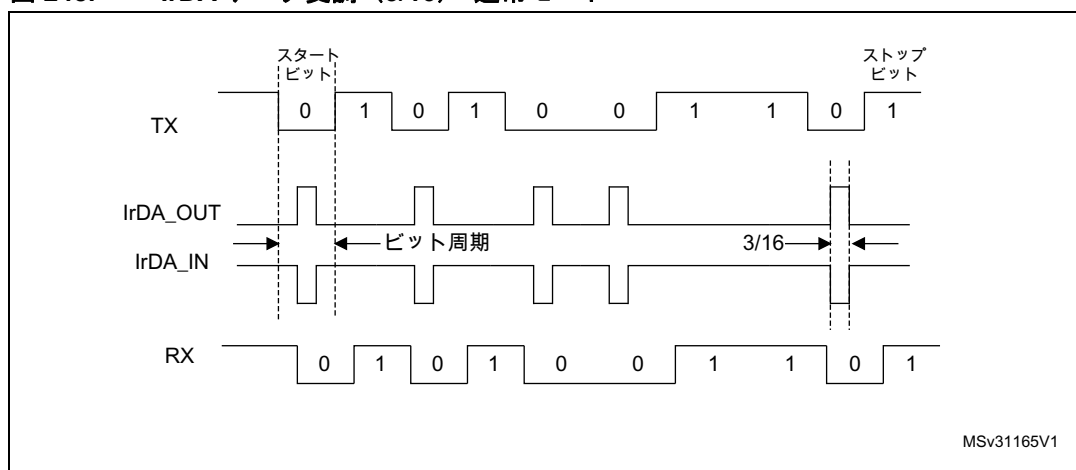


図 248. IrDA データ変調 (3/16) - 通常モード



29.5.15 DMA を使用した連続通信

USART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： **DMA モードがサポートされるかどうかについては、[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。DMA がサポートされない場合は、[セクション 29.5.2 : トランスミッタ](#) または [セクション 29.5.3 : レシーバ](#) での説明に従って USART を使用してください。連続通信を行うには、USARTx_ISR レジスタの TXE/ RXNE フラグをクリアします。**

DMA を使用した送信

DMA モードでの送信を有効にするには、USARTx_CR3 レジスタの DMAT ビットをセットします。TXE ビットがセットされるたびに、データは、DMA ペリフェラル ([セクション 11 : ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)) を参照) を使用して設定された SRAM 領域から USARTx_TDR レジスタにロードされます。DMA チャンネルを USART 送信用に割り付けるには、次の手順を実行します (x はチャンネル番号を示します)

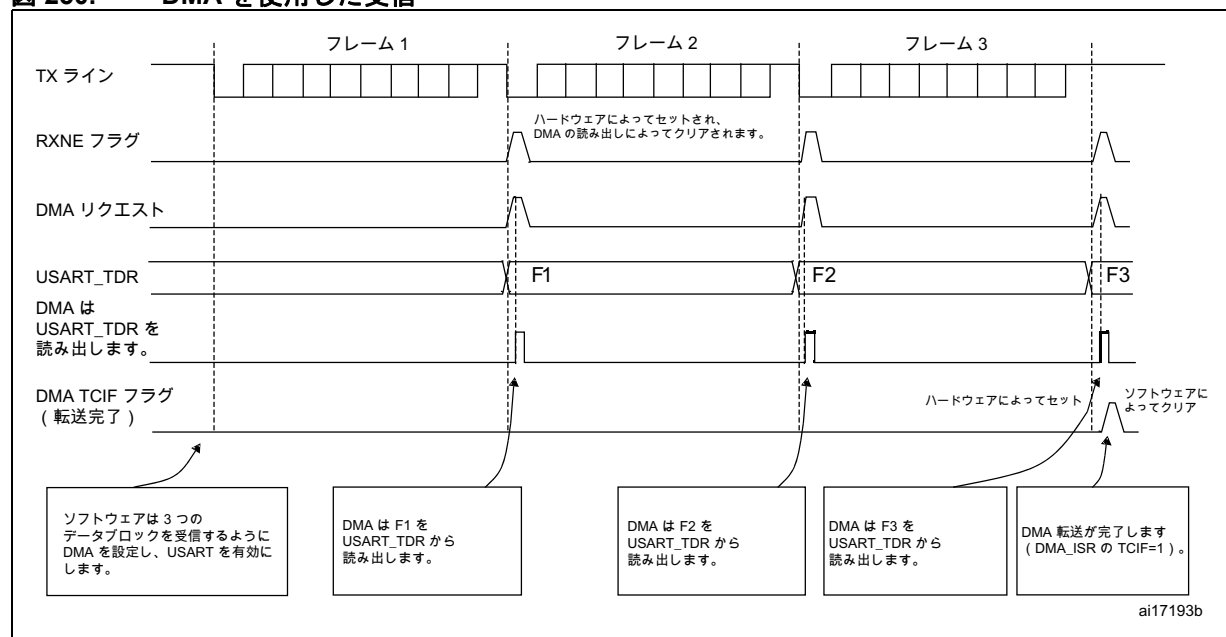
DMA を使用した受信

DMA モードでの受信を有効にするには、USARTx_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、USARTx_RDR レジスタから、DMA ペリフェラル（[セクション 11：ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)）を使用して設定された SRAM 領域にロードされます。DMA チャンネルを USART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに USARTx_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE イベント後に、USARTx_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャンネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャンネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャンネルの割り込みベクタに基づいて割り込みを生成します。

図 250. DMA を使用した受信



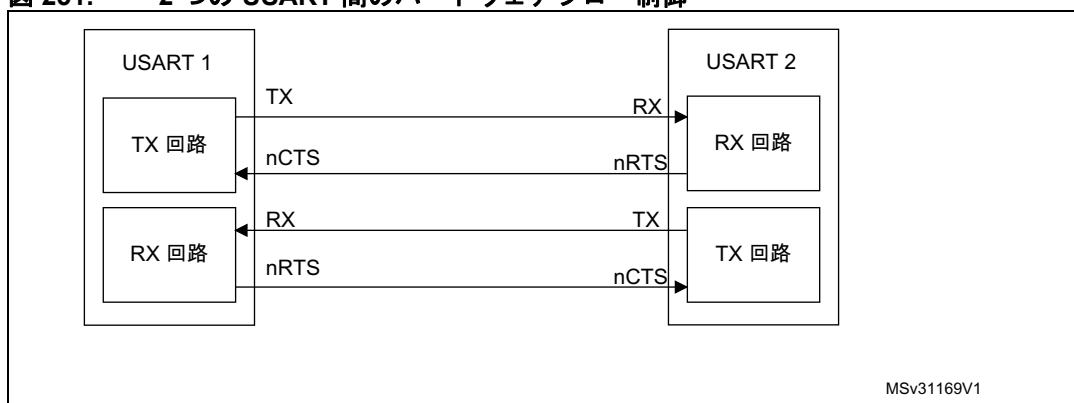
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信でトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り込み有効ビット (USARTx_CR3 レジスタの EIE ビット) があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

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

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。[図 251](#) に、このモードで 2 つのデバイスを接続する方法を示します。

図 251. 2 つの USART 間のハードウェアフロー制御

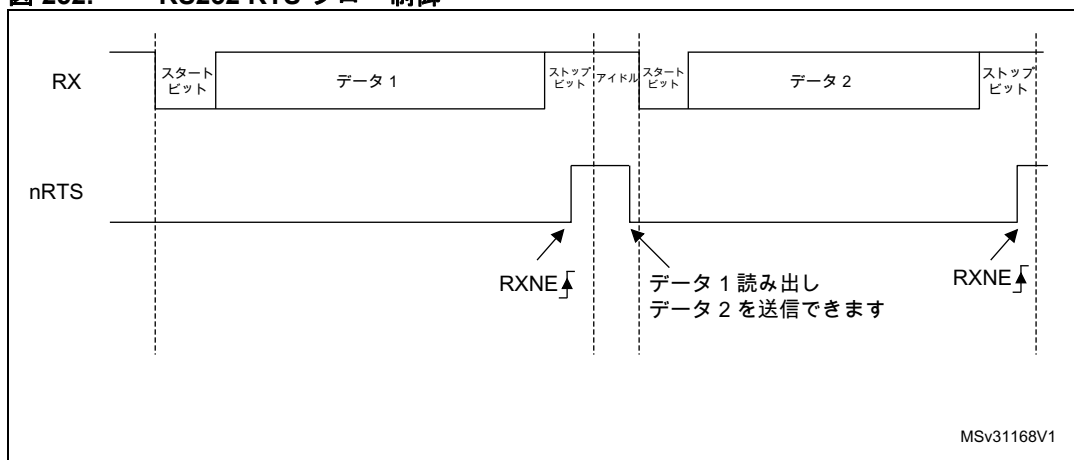


RTS と CTS のフロー制御は、USARTx_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合 (RTSE=1)、USART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます (ローレベル接続)。受信レジスタが満杯になると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。[図 252](#) に、RTS フロー制御が有効な場合の通信例を示します。

図 252. RS232 RTS フロー制御

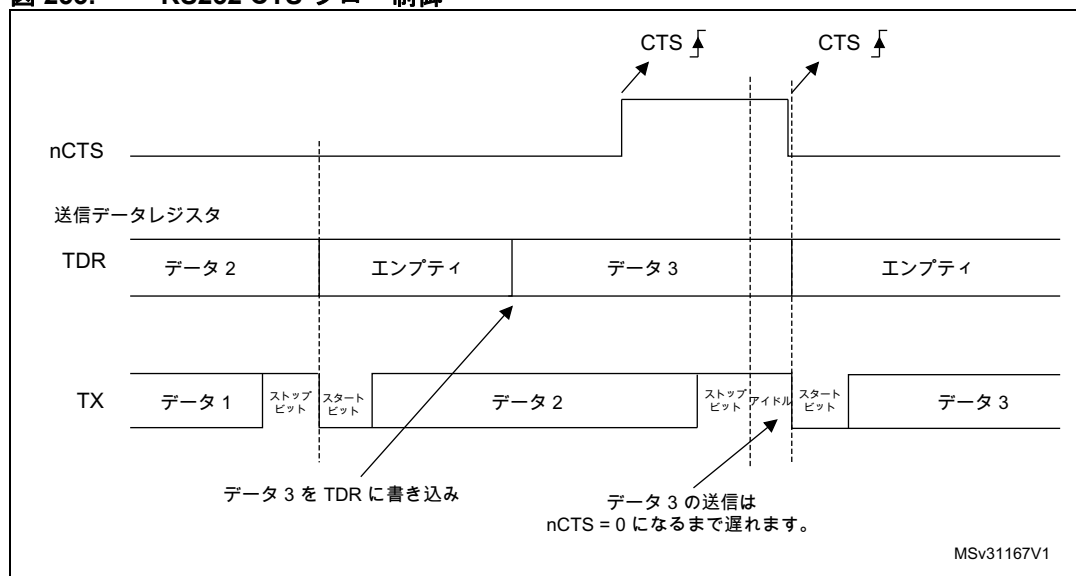


RS232 CTS フロー制御

CTS フロー制御が有効な場合 (CTSE=1)、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合 (ローレベル接続)、次のデータが送信されます (データが送信されると想定、つまり TXE=0 の場合)。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、nCTS 入力が入力がトグルすると、CTSIF ステータスビットはハードウェアによって自動的にセットされます。このビットは、レシーバの通信準備ができていているかどうかを示します。USARTx_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。図 253 に、CTS フロー制御が有効な場合の通信例を示します。

図 253. RS232 CTS フロー制御



注： 正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 USART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。

RS485 ドライバ有効

ドライバ有効機能を有効にするには、USARTx_CR3 制御レジスタのビット DEM をセットします。これにより、DE (Driver Enable) 信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から START ビットの開始までの時間です。USARTx_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。USARTx_CR1 制御レジスタの DEDT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、USARTx_CR3 制御レジスタの DEP ビットを使用して設定できます。

USART では、DEAT および DEDT はサンプル時間単位 (オーバーサンプリングレートに応じて 1/8 または 1/16 ビット時間) で表されます。

29.5.17 STOP モードからのウェイクアップ

USART は、UESM ビットがセットされ、USART クロックが HSI16 または LSE にセットされているとき、MCU を STOP モードからウェイクアップできます（[セクション 7：リセットおよびクロック制御（RCC）](#)を参照）。

STOP モードからの MCU のウェイクアップ は、標準の RXNE 割り込みを使用して行うことができます。この場合、STOP モードに入る前に RXNEIE ビットをセットする必要があります。

または、WUS ビットフィールドによって、特定の割り込みを選択することもできます。

MCU を STOP モードからウェイクアップできるようにするためには、STOP モードに入る前に、USARTx_CR1 制御レジスタの UESM ビットをセットする必要があります。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合はウェイクアップ割り込みが生成されます。

注： **STOP モードに入る前に、ユーザは USART が転送を行っていないことを確認する必要があります。BUSY フラグでは、受信中に STOP モードに入らないことを保証できません。**

WUF フラグは、MCU が STOP モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に STOP モードに入るときには、REACK ビットをチェックして、USART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、STOP モードに入る前に無効化し、STOP モードの終了時に再び有効にする必要があります。

STOP モードからのウェイクアップは、すべてのモードで使用できるわけではありません。たとえば、SPI はマスタモードでのみ動作するので、SPI モードでは機能しません。

STOP モードでのミュートモードの使用

USART は、STOP モードに入る前にミュートモードになります。

- アイドル検出は STOP モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、STOP モードからのウェイクアップのソースもアドレス一致でなければなりません。STOP モードに入るときに RXNE フラグがセットされる場合、アドレス一致によって STOP からウェイクアップしても、インタフェースはミュートモードのままです。
- USART が START ビット検出時に MCU を STOP モードからウェイクアップするように設定された場合、WUF フラグはセットされますが、RXNE フラグはセットされません。

29.6 USART 割り込み

表 121. USART 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|-------------------------------|---------|------------|
| 送信データレジスタエンプティ | TXE | TXEIE |
| CTS 割り込み | CTSIF | CTSIE |
| 送信完了 | TC | TCIE |
| 受信データレジスタノットエンプティ（データの読み出し可能） | RXNE | RXNEIE |
| オーバーランエラー検出 | ORE | |

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

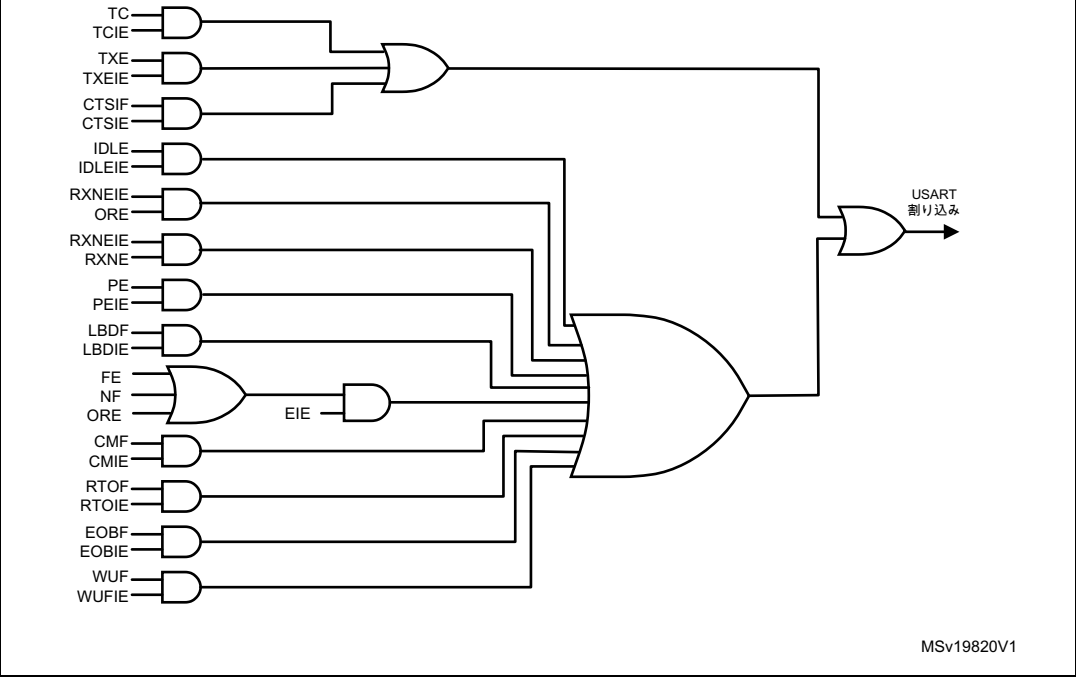
表 121. USART 割り込みリクエスト (続き)

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|---|----------------------|------------|
| アイドルライン検出 | IDLE | IDLEIE |
| パリティエラー | PE | PEIE |
| LIN ブレーク | LBDF | LBDIE |
| マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。 | NF または ORE または FE | EIE |
| キャラクター致 | CMF | CMIE |
| レシーバタイムアウトエラー | RTOF | RTOIE |
| ブロックの終了 | EOBF | EOBIE |
| STOP モードからのウェイクアップ | WUF ⁽¹⁾ | WUFIE |

1. WUF 割り込みは、STOP モードでのみアクティブです。

- USART 割り込みイベントは、同じ割り込みベクトルに接続されます (図 254 を参照)。
- 送信時: 送信完了、Clear to Send、送信データレジスタエンプティ、またはフレーミングエラー (スマートカードモード) 割り込み。
 - 受信時: アイドルライン検出、オーバーランエラー、受信データレジスタノットエンプティ、パリティエラー、LIN ブレーク検出、ノイズフラグ、フレーミングエラー、キャラクター致など。
- これらのイベントは、対応する有効制御ビットがセットされている場合に割り込みを生成します。

図 254. USART 割り込みマッピング図



29.7 USART レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

29.7.1 制御レジスタ 1 (USARTx_CR1)

アドレスオフセット : 0x00
リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|-------|------|------|----|-------|-------|-----------|------|-------|------|--------|-----------|----|----|------|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | M1 | EOBIE | RTOIE | DEAT[4:0] | | | | | DEDT[4:0] | | | | |
| | | | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| OVER8 | CMIE | MME | M0 | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | UESM | UE |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:29 予約済みであり、リセット値のままにしておかなければなりません。

ビット 28 **M1** : ワード長

このビットとビット 12 (M0) によって、ワード長が決まります。ソフトウェアによってセット/クリアされます。

M[1:0] = 00:スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = 01:スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = 10:スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート (0x7F および 0x55 フレーム検出) はサポートされません。

ビット 27 **EOBIE** : ブロック終了割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの EOBIF フラグがセットされると、USART 割り込みが生成されます。

注: USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

ビット 26 **RTOIE** : レシーバタイムアウト割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの RTOF フラグがセットされると、USART 割り込みが生成されます。

注: USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4 : USART の実装 \(690 ページ\)](#) .

ビット 25:21 **DEAT[4:0]** : ドライバ有効アサーション時間

この 5 ビット値は、DE (Driver Enable) 信号の有効化からスタートビットの開始までの時間を定義します。サンプル時間単位 (オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間) で表されます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

ビット 20:16 **DEDT[4:0]** : ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE (Driver Enable) 信号の無効化までの時間を定義します。サンプル時間単位 (オーバーサンプリングレートに応じて、1/8 または 1/16 ビット時間) で表されます。

DEDT 時間中に USARTx_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: *ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 29.4 : USART の実装 (690 ページ) を参照してください。*

ビット 15 **OVER8** : オーバーサンプリングモード

0 : 16 倍のオーバーサンプリング

1 : 8 倍のオーバーサンプリング

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: *LIN、IrDA、およびスマートカードモードでは、このビットは常にクリア状態に保つ必要があります。*

ビット 14 **CMIE** : キャラクター致割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの CMF ビットがセットされると、USART 割り込みが生成されます。

ビット 13 **MME** : ミュートモード有効

このビットは、USART のミュートモード機能を有効にします。セットされると、USART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット/クリアされます。

0 : レシーバは永続的にアクティブモードです。

1 : レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 **M0** : ワード長

このビットとビット 28 (M1) によって、ワード長が決まります。ソフトウェアによってセット/クリアされます。ビット 28 (M1) の説明を参照してください。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 11 **WAKE** : レシーバウェイクアップ方式

このビットによって、ミュートモードからの USART のウェイクアップ方式が決まります。ソフトウェアによってセット/クリアされます。

0 : アイドルライン

1 : アドレスマーク

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 10 **PCE** : パリティ制御有効

このビットは、ハードウェアのパリティ制御 (生成と検出) を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置 (M=1 の場合はビット 9、M=0 の場合はビット 8) に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット/クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0 : パリティ制御は無効です。

1 : パリティ制御は有効です。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 9 **PS** : パリティ選択

このビットは、パリティの生成/検出が有効である (PCE ビットがセットされている) とき、奇数パリティ/偶数パリティを選択します。ソフトウェアによってセット/クリアされます。パリティは、現在のバイトの後で選択されます。

0 : 偶数パリティ

1 : 奇数パリティ

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。



ビット 8 PEIE : PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの PE=1 のときには、USART 割り込みが生成されます。

ビット 7 TXEIE : 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの TXE=1 のときには、USART 割り込みが生成されます。

ビット 6 TCIE : 転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの TC=1 のときには、USART 割り込みが生成されます。

ビット 5 RXNEIE : RXNE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの ORE=1 または RXNE=1 のときには、USART 割り込みが生成されます。

ビット 4 IDLEIE : IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの IDLE=1 のときには、USART 割り込みが生成されます。

ビット 3 TE : トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0 : トランスミッタは無効です。

1 : トランスミッタは有効です。

注： 送信時、TE ビットに 0 パルス (0 の後に 1) を与えると、現在のワードの後にプリアンブル (アイドルライン) を送信します (スマートカードモードの場合を除きます)。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは USARTx_ISR レジスタの TEACK ビットをポーリングできます。

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 RE : レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0 : レシーバは無効です。

1 : レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 **UESM** : STOP モードでの USART 有効

このビットがクリアされると、USART は MCU を STOP モードからウェイクアップできません。
このビットがセットされると、USART は MCU を STOP モードからウェイクアップできますが、USART クロック選択が RCC で HSI 16 または LSE となります。
このビットは、ソフトウェアによってセット／クリアされます。
0 : USART は STOP モードから MCU をウェイクアップできません。
1 : USART は STOP モードから MCU をウェイクアップできます。この機能がアクティブなとき、USART のクロックソースは HSI 16 または LSE でなければなりません ([セクション 7 : リセットおよびクロック制御 \(RCC\)](#) を参照)。

注 : **STOP モードに入る直前に UESM ビットをセットし、STOP モードの終了時にクリアすることが推奨されます。**
USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

ビット 0 **UE** : USART 有効

このビットがクリアされると、USART プリスケールと出力はただちに停止され、現在の操作は破棄されます。USART の設定は保たれますが、USARTx_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。
0 : USART プリスケールと出力は無効であり、低電力モードです。
1 : USART は有効です。

注 : **ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをリセットする必要があります、ソフトウェアは USARTx_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。**
UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャンネルを無効にする必要があります。

注 : **デバイスがスマートカードモードで動作する場合 (SCEN = 1)、SCLK は UE ビット値にかかわらず、CLKEN = 1 の場合に常に使用可能です。**

29.7.2 制御レジスタ 2 (USARTx_CR2)

アドレスオフセット : 0x04
リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|----------|-------|-----------|-----|----------|------|------|------|-------|-------------|------|-------|----------|---------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| ADD[7:4] | | | | ADD[3:0] | | | | RTOEN | ABRMOD[1:0] | | ABREN | MSBFIRST | DATAINV | TXINV | RXINV |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SWAP | LINEN | STOP[1:0] | | CLKEN | CPOL | CPHA | LBCL | Res. | LBDIE | LBDL | ADDM7 | Res. | Res. | Res. | Res. |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | | r/w | r/w | r/w | | | | |

ビット 31:28 ADD[7:4]:USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、7 ビットアドレスマーク検出によるウェイクアップのために、ミュータモードまたは STOP モード時に使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュータモードが無効なときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のために使用することができます。この場合、受信されたキャラクタ全体 (8 ビット) が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。

このビットフィールドは、受信が無効のとき (RE=0) または USART が無効のとき (UE=0) のみ、書き込むことができます。

ビット 27:24 ADD[3:0]:USART ノードのアドレス

このビットフィールドは、認識される USART ノードのアドレスまたはキャラクタコードを指定します。

これは、マルチプロセッサ通信において、アドレスマーク検出によるウェイクアップのために、ミュータモードまたは STOP モード時に使用されます。

このビットフィールドは、受信が無効のとき (RE=0) または USART が無効のとき (UE=0) のみ、書き込むことができます。

ビット 23 RTOEN: レシーバタイムアウト有効

このビットは、ソフトウェアによってセット/クリアされます。

0: レシーバタイムアウト機能は無効です。

1: レシーバタイムアウト機能は有効です。

この機能が有効なとき、RTOR (レシーバタイムアウトレジスタ) でプログラムされた時間にわたって RX ラインがアイドル (受信なし) であった場合、USARTx_ISR レジスタの RTOF フラグがセットされます。

注: USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4: USART の実装 \(690 ページ\)](#) を参照してください。

ビット 22:21 ABRMOD[1:0]: 自動ボーレートモード

これらのビットは、ソフトウェアによってセット/クリアされます。

00: スタートビットの測定がボーレートの検出に使用されます。

01: 立ち下がりエッジから立ち下がりエッジまでの測定。(受信されたフレームはシングルビット = 1 で始まらなければならない、その場合、フレーム = Start10xxxxxx)

10: 0x7F フレーム検出

11: 0x55 フレーム検出

このビットフィールドは、ABREN=0 または USART が無効 (UE=0) のときのみ書き込むことができます。

注: DATAINV=1 および/または MSBFIRST=1 の場合、パターンはライン上で同じである必要があります (たとえば、MSBFIRST の場合は 0xAA)。

USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4: USART の実装 \(690 ページ\)](#) を参照してください。

ビット 20 ABREN: 自動ボーレート有効

このビットは、ソフトウェアによってセット/クリアされます。

0: 自動ボーレート検出は無効です。

1: 自動ボーレート検出は有効です。

注: USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4: USART の実装 \(690 ページ\)](#) を参照してください。

ビット 19 MSBFIRST: 最上位ビットファースト

このビットは、ソフトウェアによってセット/クリアされます。

0: スタートビットに続いて、データはデータビット 0 から順に送受信されます。

1: スタートビットに続いて、データは MSB (ビット 7/8/9) から順に送受信されます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 18 DATAINV : バイナリデータ反転

このビットは、ソフトウェアによってセット/クリアされます。

0 : データレジスタからの論理データは正/ダイレクトロジックで送受信されます。(1=H、0=L)

1 : データレジスタからの論理データは、負/インバースロジックで送受信されます。(1=L、0=H) パリティビットも反転されます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 17 TXINV : TX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット/クリアされます。

0 : TX ピン信号は標準ロジックレベルを使用して機能します (V_{DD} =1/アイドル、Gnd=0/マーク)。

1 : TX ピン信号値は反転されます。(V_{DD} =0/マーク、Gnd=1/アイドル)。

これにより、TX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 16 RXINV : RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット/クリアされます。

0 : RX ピン信号は標準ロジックレベルを使用して機能します (V_{DD} =1/アイドル、Gnd=0/マーク)。

1 : RX ピン信号値は反転されます。(V_{DD} =0/マーク、Gnd=1/アイドル)。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 15 SWAP : TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット/クリアされます。

0 : TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1 : TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 14 LINEN : LIN モード有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : LIN モードは無効です。

1 : LIN モードは有効です。

LIN モードでは、USARTx_RQR レジスタの SBKRQ ビットを使用して LIN 同期ブレイク (下位 13 ビット) を送信し、LIN 同期ブレイクを検出することができます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : **USART がLIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 13:12 STOP[1:0] : STOP ビット

このビットは、ストップビットのプログラミングに使用します。

00 : 1 個のストップビット

01 : 予約済み。

10 : 2 個のストップビット

11 : 1.5 個のストップビット

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 11 CLKEN : クロック有効

このビットによって、SCLK ピンを有効にできます。

0 : SCLK ピンは無効です。

1 : SCLK ピンは有効です。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : **同期モードまたはスマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に0に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

注： スマートカードに SCLK クロックを正しく供給するには、次のステップを順守する必要があります。

- UE = 0
- SCEN = 1
- GTPR 設定
- CLKEN = 1
- UE = 1

ビット 10 CPOL : クロック極性

このビットによって、同期モードにおける SCLK ピンのクロック出力の極性を選択できます。CPHA ビットと連携して動作し、希望するクロック／データ関係になるようにします。

0 : 送信ウィンドウの外で、SCLK ピンはローレベルを維持します。

1 : 送信ウィンドウの外で、SCLK ピンはハイレベルを維持します。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 9 CPHA : クロック位相

このビットは、同期モードでの SCLK ピンのクロック出力の位相を選択するために使用されます。CPOL ビットと連携して動作し、希望するクロック／データ関係になるようにします (図 242 および 図 243 を参照)。

0 : 最初のクロック遷移が最初のデータキャプチャエッジです。

1 : 2 番目のクロック遷移が最初のデータキャプチャエッジです。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 8 LBCL : 最終ビットのクロックパルス

このビットは、同期モードで送信される最終データビット (MSB) に関連するクロックパルスを、SCLK ピンに出力する必要があるかどうかを選択するために使用されます。

0 : 最終データビットのクロックパルスは、SCLK ピンに出力されません。

1 : 最終データビットのクロックパルスは、SCLK ピンに出力されます。

注意： 最終ビットは、USARTx_CR1 レジスタの M ビットによって選択された 7 または 8 または 9 ビットフォーマットに応じて送信された 7 番目または 8 番目または 9 番目のデータビットです。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： 同期モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 LBDIE : LIN ブレーク検出割り込み有効

ブレーク割り込みマスクです (ブレークデリミタを使用したブレーク検出)。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの LBDF=1 のときには、割り込みが生成されます。

注： LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 5 LBDL : LIN ブレーク検出長

このビットでは、10 ビットと 11 ビットのブレーク検出を選択します。

0 : 10 ビットブレーク検出

1 : 11 ビットブレーク検出

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 4 ADDM7 : 7 ビットアドレス検出 / 4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0 : 4 ビットアドレス検出

1 : 7 ビットアドレス検出 (8 ビットデータモード)

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : 7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス (ADD[5:0] および ADD[7:0]) に対して行われます。

ビット 3:0 予約済みであり、リセット値のままにしておかなければなりません。

注 : トランスミッタが有効なときには、3 つのビット (CPOL、CPHA、LBCL) に書き込まないでください。

29.7.3 制御レジスタ 3 (USARTx_CR3)

アドレスオフセット : 0x08

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|---------|---------|-------|------|------|------|-------|----------|------|-------------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUFIE | WUS[2:0] | | SCARCNT2:0] | | | Res. |
| | | | | | | | | | rw | rw | rw | rw | rw | rw | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DEP | DEM | DDRE | OVR DIS | ONE BIT | CTSIE | CTSE | RTSE | DMAT | DMAR | SCEN | NACK | HD SEL | IRLP | IREN | EIE |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **WUFIE** : STOP モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : Interrupt is inhibited

1 : USARTx_ISR レジスタの WUF=1 のときには、USART 割り込みが生成されます。

注： **WUFIE は、STOP モードに入る前にセットする必要があります。**

WUF 割り込みは、STOP モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 21:20 **WUS[1:0]** : STOP モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します (STOP モードからのウェイクアップフラグ)。

00WUF はアドレス一致時に有効になります (ADD[7:0] および ADDM7 による定義に従って)。

01 : 予約済み。

10 : WUF はスタートビット検出時に有効になります。

11 : WUF は RXNE 時に有効になります。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： **USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。**

ビット 19:17 **SCARCNT[2:0]** : スマートカード自動再試行カウンタ

このビットフィールドは、スマートカードにおける送受信の再試行回数を指定します。

送信モードでは、送信エラーが生成されるまでの送信の自動再試行回数を指定します (FE ビットをセット)。

受信モードでは、受信エラーが生成されるまでの受信の試行エラー回数を指定します (RXNE および PE ビットをセット)。

このビットフィールドは、USART が無効 (UE=0) のときのみプログラムする必要があります。

USART が有効になると (UE=1)、このビットフィールドは再送信を停止するために 0x0 にのみ書き込み可能です。

0x0 : 再送信無効 - 送信モードでの自動再送信禁止

0x1 から 0x7 : 自動再送信試行回数 (信号エラーの生成前)

注： **スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15 **DEP** : ドライバ有効極性選択

0 : DE 信号はアクティブハイです。

1 : DE 信号はアクティブローです。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： **ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 14 **DEM** : ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0 : DE 機能は無効です。

1 : DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： **ドライバ有効機能がサポートされない場合、このビットは予約済みであり、クリア状態に保つ必要があります。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 13 DDRE : 受信エラー時 DMA 無効

0 : 受信エラーの場合、DMA は無効になります。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないため、エラーのあるデータは転送されず (DMA リクエストなし)、次の正しい受信データが転送されます。(スマートカードモードで使用されます)

1 : 受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか (DMAR=0)、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 OVRDIS : オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0 : オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1 : オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが USARTx_RDR レジスタの前の内容に上書きされます。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 ONEBIT : 1 サンプルビット方式有効

このビットによって、サンプル方式を選択できます。1 サンプルビット方式が選択されると、ノイズ検出フラグ (NF) が無効になります。

0 : 3 サンプルビット方式

1 : 1 サンプルビット方式

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 10 CTSIE : CTS 割り込み有効

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタの CTSIF=1 のときには、割り込みが生成されます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 9 CTSE : CTS 有効

0 : CTS ハードウェアフロー制御が無効です。

1 : CTS モードが有効です。データは nCTS 入力のアサート (0 に関係) されている場合にのみ転送されます。データの送信中に nCTS 入力 negate された場合、送信は停止前に完了します。nCTS が negate されている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 8 RTSE : RTS 有効

0 : RTS ハードウェアフロー制御が無効です。

1 : RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できるとき、nRTS 出力がアサートされます (0 にプルされます)。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 7 DMAT : DMA 有効トランスミッタ

このビットは、ソフトウェアでセット/リセットされます。

1 : DMA モードは送信に有効

0 : DMA モードは送信に無効です。

ビット 6 DMAR : DMA 有効レシーバ

このビットは、ソフトウェアでセット/リセットされます。

1 : DMA モードが受信に有効です。

0 : DMA モードが受信に無効です。

ビット 5 SCEN : スマートカードモード有効

このビットはスマートカードモードを有効にするために使用します。

0 : スマートカードモードが無効です。

1 : スマートカードモードが有効です。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 4 NACK : スマートカード NACK 有効

0 : パリティエラーの際の NACK 転送が無効です。

1 : パリティエラー時の NACK 転送が有効です。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注： USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 3 HDSEL : 半二重選択

単線半二重モードの選択です。

0 : 半二重モードは選択されません。

1 : 半二重モードが選択されます。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

ビット 2 IRLP : IrDA 低電力

このビットは、通常と低電力の IrDA モードの選択に使用されます。

0 : 通常モード

1 : 低電力モード

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 1 IREN : IrDA モード有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : IrDA は無効です。

1 : IrDA は有効です。

このビットは、USART が無効 (UE=0) のときのみ書き込むことができます。

注 : IrDA モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 0 EIE : エラー割り込み有効

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ (USARTx_ISR レジスタの FE=1 または ORE=1 または NF=1) の場合に割り込み生成を有効にするために必要です。

0 : 割り込みは禁止されています。

1 : USARTx_ISR レジスタで FE=1、ORE=1、または NF=1 になると、割り込みが生成されます。

29.7.4 ボーレートレジスタ (USARTx_BRR)

このレジスタは、USART が無効 (UE=0) のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット : 0x0C

リセット値 : 0x0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:4 **BRR[15:4]**

BRR[15:4] = USARTDIV[15:4]

ビット 3:0 **BRR[3:0]**

OVER8 = 0 のとき、BRR[3:0] = USARTDIV[3:0]。

OVER8 = 1 のとき、

BRR[2:0] = USARTDIV[3:0] であり、右に 1 ビットシフトされます。

BRR[3] は、クリアされたままにする必要があります。

29.7.5 ガード時間およびプリスケアラレジスタ (USARTx_GTPR)

アドレスオフセット : 0x10

リセット値 : 0x0000

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

ビット 31:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15:8 **GT[7:0]** : ガード時間値

このビットフィールドは、ガード時間値をポーロック周期数でプログラムするために使用します。
これはスマートカードモードで使用します。このガード時間値の後は転送完了フラグがセットされます。

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 7:0 **PSC[7:0]** : プリスケアラ値

IrDA 低電力および IrDA 通常モード :

PSC[7:0] = IrDA 通常および低電力ポーレート

USART クロックソースを分周して低電力周波数を得るためのプリスケアラのプログラミングに使用します。

クロックソースは、レジスタに与えられた値 (上位 8 ビット) で分周されます。

00000000 : 予約済み - この値はプログラミングしないでください。

00000001 : クロックソースは 1 で分周されます。

00000010 : クロックソースは 2 で分周されます。

...

スマートカードモード :

PSC[4:0] : プリスケアラ値

USART クロックソースを分周してスマートカードのクロックを提供するプリスケアラのプログラミングに使用します。

レジスタで指定された値 (上位 5 ビット) を 2 倍して、クロックソース周波数の分周比を求めます。

00000 : 予約済み - この値はプログラミングしないでください。

00001 : クロックソースは 2 で分周されます。

00010 : クロックソースは 4 で分周されます。

00011 : クロックソースは 6 で分周されます。

...

このビットフィールドは、USART が無効 (UE=0) のときのみ書き込むことができます。

注: スマートカードモードが使用される場合、ビット [7:5] はクリアされたままにする必要があります。

スマートカードモードや IrDA モードがサポートされない場合、このビットフィールドは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

29.7.6 レシーバタイムアウトレジスタ (USARTx_RTOR)

アドレスオフセット : 0x14

リセット値 : 0x0000

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

ビット 31:24 **BLEN[7:0]** : ブロック長

このビットフィールドは、受信時のスマートカード T=1 のブロック長を指定します。この値は、情報文字数とエピソードフィールド長の和 (1-LEC/2-CRC) から 1 を引いた値と等しくなります。

例 :

BLEN = 0 -> 0 情報文字 + LEC

BLEN = 1 -> 0 情報文字 + CRC

BLEN = 255 -> 254 情報文字 + CRC (合計 256 文字)

スマートカードモードでは、TXE=0 のときにブロック長カウンタがリセットされます。

このビットフィールドは、他のモードでも使用できます。この場合、RE=0 (受信無効) のとき、および/または EOBCF ビットが 1 に書き込まれたときに、ブロック長カウンタがリセットされます。

注 : この値は、ブロック受信の開始後にプログラムできます (プロローグフィールドの LEN 文字のデータを使用)。必ず受信したブロックにつき一度だけプログラムするようにしてください。

ビット 23:0 **RTO[23:0]** : レシーバタイムアウト値

このビットフィールドは、レシーバタイムアウト値をポークロック数で指定します。

標準モードでは、最後の受信キャラクタの後、RTO 値を超える間、新しいスタートビットが検出されなかった場合、RTOF フラグがセットされます。

スマートカードモードでは、この値は CWT および BWT を実装するために使用されます。詳細については、スマートカードの章を参照してください。

この場合、タイムアウト測定は最後の受信キャラクタのスタートビットから始めて行われます。

注 : この値は、受信キャラクタごとにプログラムされる必要があります。

注 : **RTOR は、動作中に書き込むことができます。新しい値がカウンタ以下の場合、RTOF フラグがセットされます。**

レシーバタイムアウト機能がサポートされない場合、このレジスタは予約済みであり、ハードウェアによって 0x00000000 に強制的に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

29.7.7 リクエストレジスタ (USARTx_RQR)

アドレスオフセット : 0x18

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|-------|-------|------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXFRQ | RXFRQ | MMRQ | SBKRQ | ABRRQ |
| | | | | | | | | | | | w | w | w | w | w |

ビット 31:5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **TXFRQ** : 送信データー掃リクエスト

このビットに 1 を書き込むと、TXE フラグがクリアされます。

これにより、送信データを破棄できます。このビットは、エラー (NACK) によりデータが送信されなかった場合、および USARTx_ISR レジスタで FE フラグがアクティブである場合に、スマートカードモードでのみ使用する必要があります。

USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

ビット 3 **RXFRQ** : 受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ** : ミュートモードリクエスト

このビットに 1 を書き込むと、USART はミュートモードになり、RWU フラグがセットされます。

ビット 1 **SBKRQ** : ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注 : アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 **ABRRQ** : 自動ボーレートリクエスト

このビットに 1 を書き込むと、USARTx_ISR の ABRF フラグがリセットされ、次の受信データフレームでの自動ボーレート測定をリクエストします。

注 : USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。[セクション 29.4 : USART の実装 \(690 ページ\)](#) を参照してください。

29.7.8 割り込みおよびステータスレジスタ (USARTx_ISR)

アドレスオフセット : 0x1C

リセット値 : 0x00C0

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-------|------|------|--------|--------|------|-----|------|-----|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RE ACK | TE ACK | WUF | RWU | SBKF | CMF | BUSY |
| | | | | | | | | | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ABRF | ABRE | Res | EOBF | RTOF | CTS | CTSIF | LBDF | TXE | TC | RXNE | IDLE | ORE | NF | FE | PE |
| r | r | | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **REACK** : 受信有効確認応答フラグ

このビットは、受信有効値が USART によって考慮されるときに、ハードウェアによってセット/リセットされます。

これを使用して、STOP モードに入る前に、USART が受信できる状態であることを確認できます。

注 : *USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。*

ビット 21 **TEACK** : 送信有効確認応答フラグ

このビットは、送信有効値が USART によって考慮されるときに、ハードウェアによってセット/リセットされます

USARTx_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 **WUF** : STOP モードからのウェイクアップフラグ

USARTx_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。イベントは、WUS ビットフィールドによって定義されます。USARTx_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR3 レジスタの WUFIE=1 である場合、割り込みが生成されます。

注 : *UESM がクリアされると、WUF フラグもクリアされます。*

WUF 割り込みは、STOP モードでのみアクティブです。

USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19 **RWU** : レシーバのミュートモードからのウェイクアップ

このビットは、USART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア/セットされます。ミュートモード制御シーケンス (アドレスまたは IDLE) は、USARTx_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは USARTx_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0 : レシーバはアクティブモードです。

1 : レシーバはミュートモードです。

ビット 18 **SBKF** : ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。USARTx_RQR レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0 : ブレークキャラクタは送信されません。

1 : ブレークキャラクタは送信されます。

ビット 17 CMF : キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。USARTx_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR1 レジスタの CMIE=1 の場合、割り込みが生成されます。

0 : キャラクター致は検出されていません。

1 : キャラクター致が検出されました。

ビット 16 BUSY : ビジーフラグ

このビットは、ハードウェアによってセット/リセットされます。RX ラインで通信中（スタートビットの検出時）はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0 : USART はアイドルです（受信なし）。

1 : 受信中です。

ビット 15 ABRF : 自動ボーレートフラグ

このビットは、自動ボーレートがセットされたときにハードウェアによってセットされ（RXNE もセットされ、RXNEIE=1 の場合は割り込みが生成されます）、または、自動ボーレート操作が成功せずに完了したときにセットされます（ABRE=1）（この場合、ABRE、RXNE、および FE もセットされます）。新しい自動ボーレート検出をリクエストするために、USARTx_RQR レジスタの ABRRQ に 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： *USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。*

ビット 14 ABRE : 自動ボーレートエラー

このビットは、ボーレート測定が失敗した場合に、ハードウェアによってセットされます（範囲外のボーレートまたはキャラクタ比較の失敗）。

USARTx_CR3 レジスタの ABRRQ ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

注： *USART が自動ボーレート機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。*

ビット 13 予約済みであり、リセット値のままにしておかなければなりません。**ビット 12 EOBF** : ブロック終了フラグ

このビットは、完全なブロックが受信されたときに、ハードウェアによってセットされます（たとえば、T=1 スマートカードモード）。検出は、受信バイト数が BLEN + 4 以上である場合に行われます（ブロックの開始時から、プロローグを含む）。

USARTx_CR2 レジスタの EOBIE=1 である場合、割り込みが生成されます。

USARTx_ICR レジスタの EOBCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0 : ブロック終了に達していません。

1 : ブロック終了（文字数）に達しました。

注： *スマートカードモードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。*

ビット 11 RTOF : レシーバタイムアウト

このビットは、RTOR レジスタでプログラムされたタイムアウト値が通信なしで経過したときに、ハードウェアによってセットされます。USARTx_ICR レジスタの RTOCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR2 レジスタの RTOIE=1 の場合、割り込みが生成されます。

スマートカードモードでは、タイムアウトは CWT または BWT タイミングに対応します。

0 : タイムアウト値に達していません。

1 : データを受信せずにタイムアウト値に達しました。

注 : 時間が RTOR レジスタでプログラムされた値に等しい場合、2 つのキャラクタが分離され、RTOF はセットされません。この時間がこの値に 2 サンプル時間 (オーバーサンプリング方式によって 2/16 または 2/8) を加えた値を超える場合、RTOF フラグがセットされます。

カウンタは RE=0 の場合でもカウントしますが、RTOF は RE=1 のときのみセットされます。RE がセットされたときにタイムアウトがすでに経過していた場合、RTOF はセットされます。

USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 10 CTS : CTS フラグ

このビットは、ハードウェアによってセット/リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0 : nCTS ラインはセットされました。

1 : nCTS ラインはリセットされました。

注 : ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 CTSIF : CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力が入力トグルしたときにハードウェアによってセットされます。USARTx_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、USARTx_CR3 レジスタで CTSIE=1 であれば、割り込みが生成されます。

0 : nCTS ステータスラインでの変更はありません。

1 : nCTS ステータスラインで変更がありました。

注 : ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 LBDF : LIN ブレーク検出フラグ

このビットは、LIN ブレークが検出されると、ハードウェアによってセットされます。USARTx_ICR レジスタの LBDCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR2 レジスタの LBDIE=1 である場合、割り込みが生成されます。

0 : LIN ブレークは検出されませんでした。

1 : LIN ブレークが検出されました。

注 : USART が LIN モードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。

ビット 7 TXE : 送信データレジスタエンプティ

このビットは、USARTx_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。USARTx_TDR レジスタへの書き込みによってクリアされます。

TXE フラグは、USARTx_RQR レジスタの TXFRQ に 1 を書き込んでクリアし、データを破棄することもできます (スマートカード T=0 モードでの送信失敗の場合のみ)。

USARTx_CR1 レジスタの TXEIE ビット=1 の場合、割り込みが生成されます。

0 : データはシフトレジスタに転送されません。

1 : データはシフトレジスタに転送されます。

注 : このビットは、シングルバッファ送信時に使用されます。

ビット 6 TC : 送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。USARTx_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。USARTx_ICR レジスタの TCCF に 1 を書き込むことによって、または USARTx_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。

0 : 送信は完了していません。

1 : 送信は完了しています。

注 : TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 RXNE : 読み出しデータレジスタノットエンプティ

このビットは、RDR シフトレジスタの内容が USARTx_RDR レジスタに転送されると、ハードウェアによってセットされます。USARTx_RDR レジスタの読み出しによってクリアされます。RXNE フラグは、USARTx_RQR レジスタの RXFRQ に 1 を書き込むことによってクリアすることもできます。

USARTx_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。

0 : データは受信されていません。

1 : 受信データを読み出すことができます。

ビット 4 IDLE : アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。USARTx_CR1 レジスタの IDLEIE=1 の場合、割り込みが生成されます。USARTx_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0 : アイドルラインは検出されていません。

1 : アイドルラインが検出されました。

注 : RXNE ビットがセットされるまで (新しいアイドルラインが発生するまで)、IDLE ビットは再びセットされません。

ミュートモードが有効な場合 (MME=1)、USART がミュートでない場合 (RWU=0)、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 ORE : オーバーランエラー

このビットは、RXNE=1 のときに、シフトレジスタで現在受信中のデータを RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。USARTx_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割り込みが生成されます。

0 : オーバーランエラーはありません。

1 : オーバーランエラーが検出されました。

注 : このビットがセットされると、RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。

USARTx_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます (オーバーラン検出なし)。

ビット 2 NF : START ビットノイズ検出フラグ

このビットは、受信フレームでノイズが検出されるとハードウェアによってセットされます。USARTx_ICR レジスタの NFCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0 : ノイズは検出されていません。

1 : ノイズが検出されました。

注 : このビットは、割り込みを生成する RXNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NF フラグがセットされると、割り込みが生成されます。

注 : ラインがノイズフリーであるとき、NF フラグを無効にして、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増加させることができます (セクション 29.5.5 : クロック偏差に対する USART レシーバの許容誤差 (706 ページ) を参照)。

ビット 1 FE : フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。USARTx_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

スマートカードモードでは、送信時、成功せずに (カードがデータフレームをNACK) 最大送信試行回数に達すると、このビットがセットされます。

USARTx_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0 : フレーミングエラーは検出されていません。

1 : フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 PE : パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。USARTx_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

USARTx_CR1 レジスタの PEIE=1 である場合、割り込みが生成されます。

0 : パリティエラーはありません。

1 : パリティエラー

29.7.9 割り込みフラグクリアレジスタ (USARTx_ICR)

アドレスオフセット : 0x20

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|-------|-------|------|-------|-------|------|------|------|--------|-------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUCF | Res. | Res. | CMCF | Res. |
| | | | | | | | | | | | W | | | W | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | EOBCF | RTOCF | Res. | CTSCF | LBDCF | Res. | TCCF | Res. | IDLECF | ORECF | NCF | FECF | PECF |
| | | | W | W | | W | W | | W | | W | W | W | W | W |

ビット 31:21 予約済みであり、リセット値のままにしておかなければなりません。

ビット 20 **WUCF** : STOP モードからのウェイクアップクリアフラグ

このビットに 1 を書き込むと、USARTx_ISR レジスタの WUF フラグがクリアされます。

注 : **USART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。**

ビット 19:18 予約済みであり、リセット値のままにしておかなければなりません。

ビット 17 **CMCF** : キャラクター致フラグクリア

このビットに 1 を書き込むと、USARTx_ISR レジスタの CMF フラグがクリアされます。

ビット 16:13 予約済みであり、リセット値のままにしておかなければなりません。

ビット 12 **EOBCF** : ブロック終了クリアフラグ

このビットに 1 を書き込むと、USARTx_ISR レジスタの EOBFF フラグがクリアされます。

注 : **USART がスマートカードモードをサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 11 **RTOCF** : レシーバタイムアウトフラグクリア

このビットに 1 を書き込むと、USARTx_ISR レジスタの RTOF フラグがクリアされます。

注 : **USART がレシーバタイムアウト機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **CTSCF** : CTS フラグクリア

このビットに 1 を書き込むと、USARTx_ISR レジスタの CTSIF フラグがクリアされます。

注 : **ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 8 **LBDCF** : LIN ブレーク検出クリアフラグ

このビットに 1 を書き込むと、USARTx_ISR レジスタの LBDF フラグがクリアされます。

注 : **LIN モードがサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。セクション 29.4 : USART の実装 (690 ページ) を参照してください。**

ビット 7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **TCCF** : 送信完了フラグクリア

このビットに 1 を書き込むと、USARTx_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値のままにしておかなければなりません。

- ビット 4 **IDLECF** : アイドルライン検出フラグクリア
このビットに 1 を書き込むと、USARTx_ISR レジスタの IDLE フラグがクリアされます。
- ビット 3 **ORECF** : オーバーランエラーフラグクリア
このビットに 1 を書き込むと、USARTx_ISR レジスタの ORE フラグがクリアされます。
- ビット 2 **NCF** : ノイズ検出フラグクリア
このビットに 1 を書き込むと、USARTx_ISR レジスタの NF フラグがクリアされます。
- ビット 1 **FE CF** : フレーミングエラーフラグクリア
このビットに 1 を書き込むと、USARTx_ISR レジスタの FE フラグがクリアされます。
- ビット 0 **PE CF** : パリティエラーフラグクリア
このビットに 1 を書き込むと、USARTx_ISR レジスタの PE フラグがクリアされます。

29.7.10 受信データレジスタ (USARTx_RDR)

アドレスオフセット : 0x24
リセット値 : 未定義

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

- ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。
- ビット 8:0 **RDR[8:0]** : 受信データ値
受信データキャラクタを含みます。
RDR レジスタは、入カシフトレジスタと内部バスとの間にパラレルインタフェースを提供します ([図 230](#) を参照)。
パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

29.7.11 送信データレジスタ (USARTx_TDR)

アドレスオフセット : 0x28

リセット値 : 未定義

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

ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8:0 **TDR[8:0]** : 送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出力シフトレジスタとの間にパラレルインタフェースを提供します ([図 230](#) を参照)。

パリティを有効にして (USARTx_CR1 レジスタの PCE ビットに 1 をセット) 送信しているとき、MSB (データ長に応じてビット 7 または 8) に書き込まれた値は、パリティによって置き換えられるため、無効です。

注 : このレジスタは、TXE=1 のときのみ書き込むことができます。

29.7.12 USART レジスタマップ

次の表に、USART のレジスタマップとリセット値を示します。

表 122. USART レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|-------------|-----------|------|------|------|----------|-------|-------|-------|-----------|---------|----------|-------|--------------|---------|-------|-----------|-------|-------|-----------|--------|-------|------|----------|-------|-------|-------|--------|--------|-------|------|-------|-------|
| 0x00 | USARTx_CR1 | Res. | Res. | Res. | M1 | EOBIE | RTOIE | DEAT4 | DEAT3 | DEAT2 | DEAT1 | DEAT0 | DEDI4 | DEDI3 | DEDI2 | DEDI1 | DEDT0 | OVER8 | CMIE | MME | M0 | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | UESM | UE |
| | リセット値 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | USARTx_CR2 | ADD[7:4] | | | | ADD[3:0] | | | | RTOEN | ABRMOD1 | ABRMOD0 | ABREN | MSBFIRST | DATAINV | TXINV | RXINV | SWAP | LINEN | STOP[1:0] | | CLKEN | CPOL | CPHA | LBCL | Res. | LBDIE | LBDL | ADDM7 | Res. | Res. | Res. | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | | | | |
| 0x08 | USARTx_CR3 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUFIE | WUS[1:0] | | SCAR CNT2:0] | | Res. | DEP | DEM | DDRE | OVRDIS | ONEBIT | CTSIE | CTSE | RTSE | DMAT | DMAR | SCEN | NACK | HDSSEL | IRLP | IREN | FIF | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | USARTx_BRR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BRR[15:0] | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x10 | USARTx_GTPR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | GT[7:0] | | | | | | | PSC[7:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x14 | USARTx_RTOR | BLEN[7:0] | | | | | | | | RTO[23:0] | | | | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x18 | USARTx_RQR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXFRQ | RXFRQ | MMRQ | SBKRQ | ABRRQ |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x1C | USARTx_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | REACK | TEACK | WUF | RWU | SBKF | CMF | BUSY | ABRF | ABRE | Res. | EOBF | RTOF | CTS | CTSIF | LBDF | TXE | TC | RXNE | IDLE | ORE | NF | FE | PE |
| | リセット値 | | | | | | | | | | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x20 | USARTx_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUCF | Res. | Res. | CMCF | Res. | Res. | Res. | Res. | EOBCF | RTOCF | Res. | CTSCF | LBDCF | Res. | TCOF | Res. | IDLECF | ORECF | NCF | FCF | PECF |
| | リセット値 | | | | | | | | | | | | 0 | | | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x24 | USARTx_RDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RDR[8:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | X | X | X | X | X | X | X | X | X |
| 0x28 | USARTx_TDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TDR[8:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | X | X | X | X | X | X | X | X | X |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

30 低電力ユニバーサル非同期レシーバトランスミッタ (LPUART)

30.1 概要

低電力ユニバーサル非同期レシーバトランスミッタ (LPUART) は、限られた消費電力で双方向 UART 通信が可能な UART です。わずか 32.768 kHz LSE クロックで最大 9600 baud/s の UART 通信が可能です。LSE クロックとは別のクロックソースによって LPUART にクロック供給すると、さらに高いボーレートを達成できます。

マイクロコントローラがストップモードのときでも、LPUART は UART フレームの着信を待つことができ、その間のエネルギー消費は非常に低くなっています。LPUART には、最小の消費電力で非同期シリアル通信を可能にするために必要なすべてのハードウェアサポートが含まれています。

半二重単線通信とモデム操作 (CTS/RTS) をサポートします。

マルチプロセッサ通信もサポートします。

データの送受信に DMA (直接メモリアクセス) を使用できます。

30.2 LPUART の主な機能

- 全二重非同期通信
- NRZ 標準フォーマット (マーク/スペース)
- プログラム可能なボーレート
- 32.768 kHz クロックソースを使用して、300 baud/s から 9600 baud/s まで。
- より高い周波数のクロックソースを使用することにより、より高いボーレートを達成可能
- デュアルクロックドメイン可能
 - UART 機能と STOP モードからのウェイクアップ
 - PCLK 再プログラミングから独立した便利なボーレートプログラミング
- プログラム可能なデータワード長 (7 または 8 または 9 ビット)
- データ順序をプログラム可能 (MSB ファーストまたは LSB ファースト シフト)
- 設定可能なストップビット (1 または 2 個のストップビット)
- 単線半二重通信
- DMA を使用した連続通信
- 送受信バイトは集中型 DMA を使用して予約済み SRAM にバッファ
- トランスミッタとレシーバ用に個別の有効ビット
- 送信と受信の信号極性を個別に制御
- スワップ可能な Tx/Rx ピン設定
- モデムと RS-485 トランシーバのハードウェアフロー制御
- 転送検出フラグ :
 - 受信バッファフル
 - 送信バッファエンプティ
 - ビジーおよび送信終了フラグ
- パリティ制御 :
 - パリティビットの送信
 - 受信したデータバイトのパリティ検査
- 4 つのエラー検出フラグ :
 - オーバーランエラー
 - ノイズ検出
 - フレームエラー
 - パリティエラー
- 14 の割り込みソース (フラグ付き)
- マルチプロセッサ通信
アドレスが一致しない場合、LPUART はミュートモードに入ります。
- ミュートモードからのウェイクアップ (アイドルライン検出またはアドレスマーク検出による)

30.3 LPUART の実装

STM32L0x3 デバイスには 1 個の LPUART が埋め込まれています。LPUART によってサポートされる機能については、[セクション 29.4 : USART の実装](#) を参照してください。

30.4 LPUART 機能の詳細

LPUART の双方向通信には、少なくとも 2 本のピンが必要です。すなわち、受信データ入力 (RX) と送信データ出力 (TX) です。

- **RX** : 受信データ入力。
これはシリアルデータ入力です。
- **TX** : 送信データ出力。
トランスミッタが無効なときは、出力ピンは入出力ポート設定に戻ります。トランスミッタが有効で、送信すべきデータがないとき、TX ピンはハイレベルになります。単線モードでは、この I/O はデータの送受信に使用されます。

通常の LPUART モードでは、フレームは以下の部分で構成され、これらのピンを通じてシリアルデータが送受信されます。

- 送受信前のアイドルライン
- スタートビット
- データワード (7、8、または 9 ビット) LSBファースト
- フレームの完了を示す 1 個、2 個のストップビット
- LPUART インタフェースはボーレートジェネレータを使用
- ステータスレジスタ (LPUARTx_ISR)
- 受信および送信データレジスタ (LPUARTx_RDR、LPUARTx_TDR)
- ボーレートレジスタ (LPUARTx_BRR)

各ビットの定義については、[セクション 30.6 : LPUART レジスタ](#) を参照してください。

RS232 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **nCTS** : Clear To Send は、ハイレベルのとき、現在の転送の終わりにデータ送信をブロックします。
- **nRTS** : Request To Send は、LPUART がデータを受信する準備ができた (ローである) ことを示します。

RS485 ハードウェアフロー制御モードでは、以下のピンが必要です。

- **DE** : Driver Enable は、外部トランシーバの送信モードを有効にします。

注 : **DE と nRTS は同じピンを共有します。**

[illegible]

30.4.1 LPUART キャラクタの説明

ワード長は、LPUARTx_CR1 レジスタの M[1:0] ビットをプログラムすることによって、7、8、または 9 ビットとして選択できます (図 256 を参照)。

- 7 ビットのキャラクタ長 : M[1:0] = 10
- 8 ビットのキャラクタ長 : M[1:0] = 00
- 9 ビットのキャラクタ長 : M[1:0] = 01

注 : 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ボーレート (0x7F および 0x55 フレーム検出) はサポートされません。

デフォルト設定では、信号 (TX または RX) はスタートビットの処理中ではロー状態です。また、ストップビットの処理中にはハイ状態です。

これらの値は、極性設定制御により、各信号について個別に反転できます。

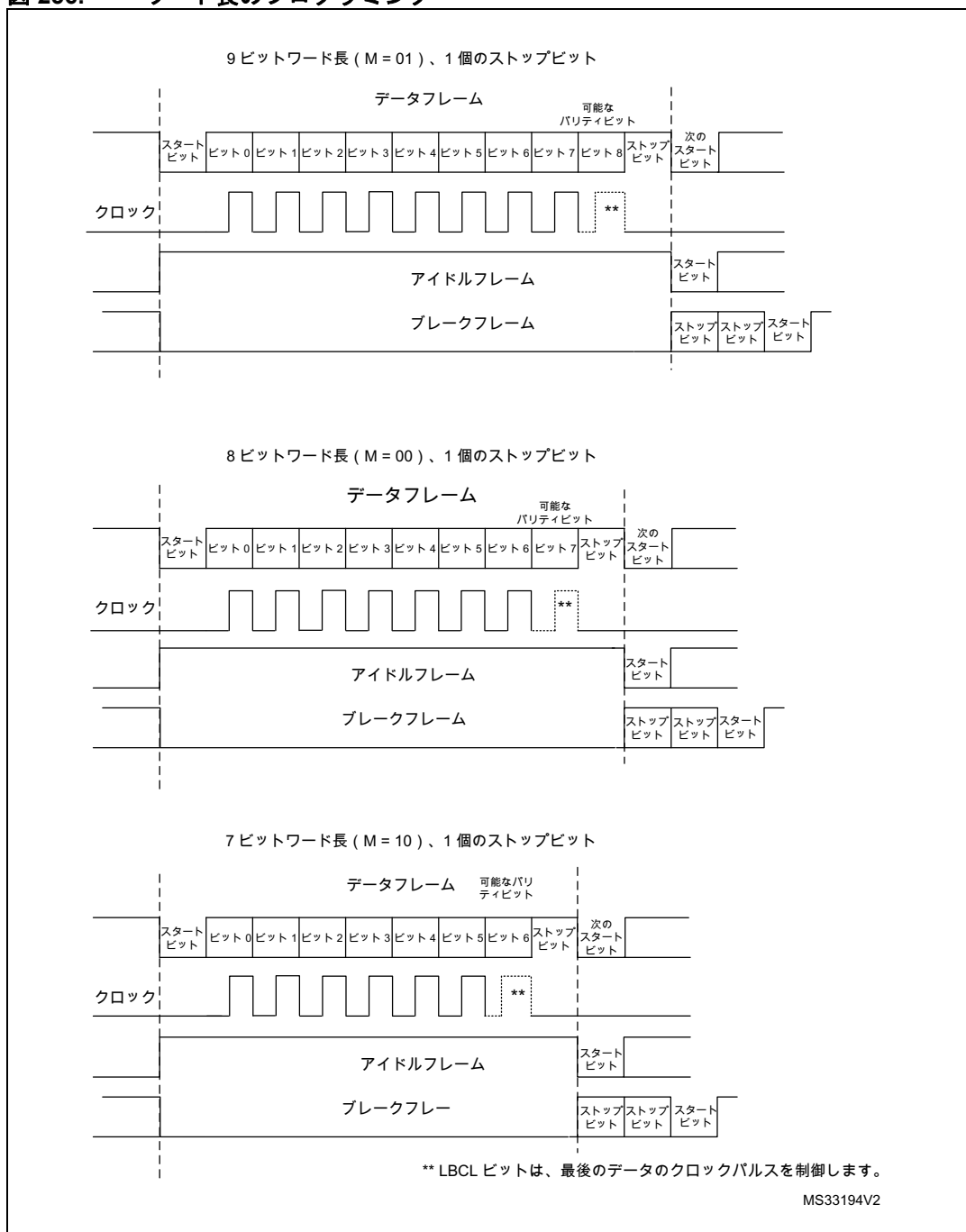
アイドルキャラクタは、すべてが「1」のフレームとして解釈されます (「1」の数にはストップビットの数が含まれます)。

ブレイクキャラクタは、フレーム周期中に「0」を受信することと解釈されます。ブレイクフレームの終了時、トランスミッタは 2 個のストップビットを挿入します。

送信と受信は、共通のボーレートジェネレータによって駆動され、それぞれのクロックは、トランスミッタとレシーバの有効ビットがそれぞれセットされたときに生成されます。

各ブロックの詳細を次に示します。

図 256. ワード長のプログラミング



30.4.2 トランスミッタ

トランスミッタは、M ビットのステータスに応じて、7または 8 または 9 ビットのデータワードを送信できます。トランスミッタ機能を有効にするには、送信イネーブルビット (TE) をセットする必要があります。送信シフトレジスタ内のデータは、TX ピンで出力されます。

キャラクタ送信

LPUART 送信時、データはLSBファースト (デフォルト設定) で TX ピンにシフトアウトされます。このモードでは、LPUARTx_TDR レジスタは、内部バスと送信シフトレジスタの間のバッファ (TDR) で構成されます (図 230 を参照)。

各キャラクタの前には、スタートビット (1 ビット周期、論理レベルロー) があります。キャラクタは、設定可能な数のストップビットで終端されます。

LPUART では、1 個および 2 個のストップビットがサポートされます。

注： **送信データを LPUARTx_TDR に書き込む前に、TE ビットをセットする必要があります。**
データの送信中に TE ビットをリセットしないでください。送信中に TE ビットをリセットすると、ボーレートカウンタが停止されるため、TX ピンのデータが破壊されます。送信中のデータは失われます。
TE ビットが有効になると、アイドルフレームが送信されます。

設定可能なストップビット

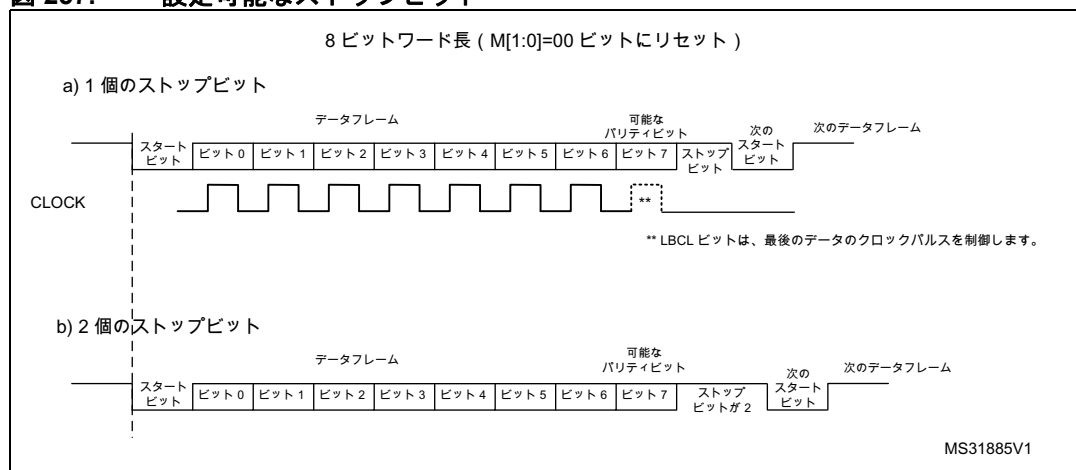
各キャラクタとともに送信されるストップビットの数は、制御レジスタ 2 のビット 13、12 でプログラミングできます。

- **1 個のストップビット：** ストップビット数のデフォルト値です。
- **2 個のストップビット：** これは、通常の LPUART モード、単線モード、およびモデムモードでサポートされます。

アイドルフレームの送信にはストップビットが含まれます。

ブ레이크送信は、10 個のロービット (M[1:0] = 00 のとき) または 11 個のロービット (M[1:0] = 01 のとき) または 9 個のロービット (M[1:0] = 10 のとき) の後に 2 個のストップビットが続きます。長いブ레이크 (9/10/11 個のロービットを超える長さのブ레이크) を送信することはできません。

図 257. 設定可能なストップビット



キャラクタ送信手順

1. LPUARTx_CR1 の M ビットをプログラムして、ワード長を定義します。
2. LPUARTx_BRR レジスタを使用して、目的のボーレートを選択します。
3. LPUARTx_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUARTx_CR1 レジスタの UE ビットに 1 を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合は、LPUARTx_CR3 レジスタの DMA 有効 (DMAT) を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUARTx_CR1 の TE ビットをセットして、最初の送信としてアイドルフレームを送信します。
7. 送信するデータを LPUARTx_TDR レジスタに書き込みます (これによって TXE ビットがクリアされます)。シングルバッファの場合、送信される各データにこれを繰り返します。
8. LPUARTx_TDR レジスタに最後のデータを書き込んだ後、TC=1 になるまで待ちます。これは、最後のフレームの送信が完了したことを示します。これは、たとえば最後の送信内容の破壊を避けるために、LPUART が無効になったり HALT モードに入ったりする場合に必要です。

1 バイト通信

TXE ビットは、常に送信データレジスタへの書き込みによってクリアされます。

TXE ビットは、ハードウェアによってセットされ、次を示します。

- データは LPUARTx_TDR レジスタからシフトレジスタへ移動され、データ送信が開始されます。
- LPUARTx_TDR レジスタは空です。
- 次のデータを、前のデータに上書きせずに、LPUARTx_TDR レジスタに書き込みます。

TXEIE ビットがセットされている場合、このフラグは割り込みを生成します。

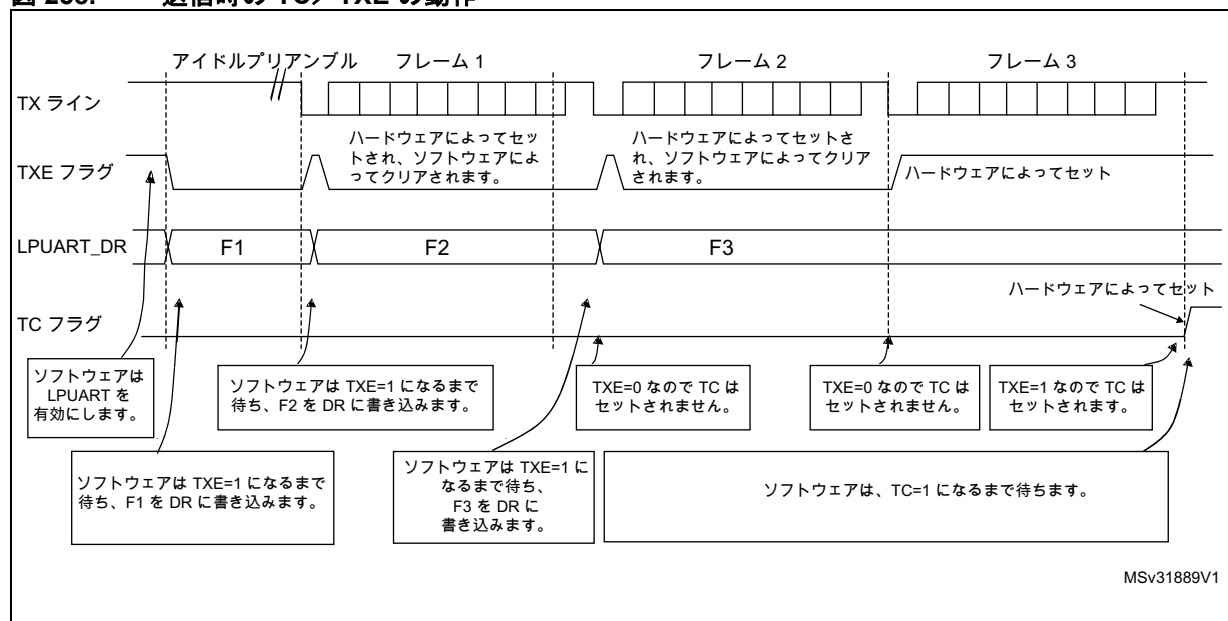
送信が行われている場合、LPUARTx_TDR レジスタへの書き込み命令によってデータが TDR レジスタに格納され、さらに、現在の送信の最後にシフトレジスタにコピーされます。

送信が行われていないときには、LPUARTx_TDR レジスタへの書き込み命令によってデータがシフトレジスタに格納され、データ送信が開始され、TXE ビットがセットされます。

フレームが送信され (ストップビットの後)、TXE ビットがセットされると、TC ビットはハイレベルになります。LPUARTx_CR1 レジスタの TCIE ビットがセットされると、割り込みが生成されます。

LPUARTx_TDR レジスタに最後のデータを書き込んだ後は、LPUART を無効にしたり、マイクロコントローラを低電力モードにする前に TC=1 になるまで待つ必要があります ([図 233 : 送信時の TC/TXE の動作](#) を参照)。

図 258. 送信時の TC/TXE の動作



ブ레이크キャラクタ

SBKRQ ビットをセットすると、ブ레이크キャラクタが送信されます。ブ레이크フレーム長は、M ビットに依存します (図 256 を参照)。

SBKRQ ビットに 1 が書き込まれた場合、現在のキャラクタ送信の完了後、TX ラインにブ레이크キャラクタが送信されます。SBKF ビットは書き込み操作によってセットされ、ブ레이크キャラクタが完了すると (ブ레이크キャラクタの後のストップビット時に) ハードウェアによってリセットされます。LPUART は、次のフレームのスタートビットの認識を保証するために、ブ레이크フレームの終わりに 2 ビットの長さの間、ロジック 1 信号 (STOP) を挿入します。

アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブ레이크キャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

アイドルキャラクタ

TE ビットをセットすると、LPUART は、最初のデータフレームの前にアイドルフレームを送信します。

30.4.3 レシーバ

LPUART は、LPUARTx_CR1 レジスタの M ビットに応じて、7 または 8 または 9 ビットのデータワードを受信できます。

スタートビット検出

LPUART では、START ビット検出については、まず、立ち下がリエッジが Rx ラインで検出された後、スタートビットの中央でサンプルが採取され、まだ「0」であることが確認されます。スタートサンプルが「1」であった場合、ノイズエラーフラグ (NF) がセットされ、START ビットが破棄され、レシーバは新しい START ビットを待ちます。そうでない場合、レシーバは通常通り、着信するすべてのビットのサンプリングを続けます。

キャラクタの受信

LPUART の受信時には、データは RX ピンを通じて LSB ファーストでシフトインされます。このモードでは、LPUARTx_RDR レジスタは、内部バスと受信シフトレジスタの間のバッファ (RDR) で構成されます。

キャラクタ受信手順

1. LPUARTx_CR1 の M ビットをプログラムして、ワード長を定義します。
2. ボーレートレジスタ LPUARTx_BRR を使用して、目的のボーレートを選択します。
3. LPUARTx_CR2 レジスタでストップビットの数をプログラミングします。
4. LPUARTx_CR1 レジスタの UE ビットに 1 を書き込んで、LPUART を有効にします。
5. マルチバッファ通信を行う場合には、LPUARTx_CR3 レジスタの DMA 有効 (DMAR) を選択します。マルチバッファ通信の説明に基づいて、DMA レジスタを設定します。
6. LPUARTx_CR1 レジスタの RE ビットをセットします。これによってレシーバが有効になり、スタートビットの検索を開始します。

キャラクタが受信されると、

- RXNE ビットがセットされます。これは、シフトレジスタの内容が RDR レジスタに転送されたことを示します。言い換えると、データは受信され、読み出し可能です (関連するエラーフラグも同様です)。
- RXNEIE ビットがセットされていた場合、割り込みが生成されます。
- 受信中にフレームエラー、ノイズまたはオーバーランエラーが検出された場合、エラーフラグをセットできます。RXNE によって PE フラグもセットできます。
- マルチバッファでは、RXNE はバイト受信ごとにセットされ、受信データレジスタの DMA 読み出しによってクリアされます。
- シングルバッファモードでは、RXNE ビットのクリアは、ソフトウェアによる LPUARTx_RDR レジスタの読み出しによって行われます。RXNE フラグは、LPUARTx_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。オーバーランエラーを避けるには、次のキャラクタの受信が終了する前に、RXNE ビットをクリアする必要があります。

ブレークキャラクタ

ブレークキャラクタを受信すると、LPUART はブレークキャラクタをフレーミングエラーとして処理します。

アイドルキャラクタ

アイドルフレームが検出されると、受信データキャラクタの場合と同じ手順が行われ、IDLEIE ビットがセットされていた場合は割り込みが生成されます。

オーバーランエラー

オーバーランエラーは、RXNE がリセットされていないときにキャラクタを受信したときに発生し、RXNE ビットがクリアされるまで、データはシフトレジスタから RDR レジスタに転送できません。

RXNE フラグは、バイトを受信するたびにセットされます。次のデータを受信したときに RXNE フラグがセットされていた場合、または前回の DMA リクエストがまだ処理されていない場合、オーバーランエラーが発生します。オーバーランエラーが発生すると、

- ORE ビットがセットされます。
- RDR の内容は失われません。LPUARTx_RDR の読み出しが行われると、前のデータが使用できません。
- シフトレジスタは上書きされます。その後、オーバーラン中に受信されたデータは失われます。
- RXNEIE ビットがセットされているか、EIE ビットがセットされている場合、割り込みが生成されます。
- ORE ビットは、ICR レジスタの ORECF ビットをセットすることによってリセットされます。

注： ORE ビットがセットされた場合、少なくとも 1 個のデータが失われています。この場合、次の 2 つの可能性がありま

- RXNE=1 の場合、最後の有効なデータが受信レジスタ RDR に格納され、読み出すことができます。

- RXNE=0 の場合、最後の有効なデータはすでに読み出されたので、RDR には読み出すべきものがないことを意味します。このケースは、有効な最後のデータが RDR で読み出されると同時に新しい（そして失われた）データが受信されると発生します。

クロックソースの選択

クロックソースの選択は、リセットおよびクロック制御システム (RCC) を通じて行われます。クロックソースは、LPUART を有効にする (UE ビットをセットする) 前に選ぶ必要があります。

クロックソースの選択は、2 つの基準に従って行われる必要があります。

- LPUART を低電力モードで使用できること
- 通信速度

クロックソース周波数は、 f_{CK} です。

デュアルクロックドメインと STOP モードからのウェイクアップ機能がサポートされるとき、クロックソースは、 f_{PCLK} (デフォルト)、 f_{LSE} 、 f_{HSI} 、または f_{SYS} のいずれかにできます。そうでない場合、LPUART のクロックソースは f_{PCLK} です。

f_{LSE} 、 f_{HSI} をクロックソースとして選択すると、LPUART は MCU が低電力モードのときにデータを受信できます。受信データとウェイクアップモードの選択に応じて、LPUART は必要なときに MCU をウェイクアップして、ソフトウェアが LPUARTx_RDR レジスタを読み出すことによって、または DMA によって受信データを転送します。

他のクロックソースの場合、LPUART 通信を可能にするためには、システムを有効にする必要があります。

通信速度の範囲 (特に最大通信速度) もクロックソースによって決まります。

レシーバは各着信ポーをポー周期のできるだけ中央でサンプリングします。各着信ポーについて 1 つのサンプルだけが採取されます。

注： データのノイズ検出は行われません。

フレーミングエラー

フレーミングエラーは、次の場合に検出されます。

非同期化または過剰なノイズのため、受信時に予想されたタイミングでストップビットが認識されない場合。

フレーミングエラーが検出された場合：

- FE ビットがハードウェアによってセットされます。
- 無効なデータがシフトレジスタから LPUARTx_RDR レジスタへ転送されます。
- シングルバイト通信の場合、割り込みは生成されません。ただし、このビットは、割り込みを生成する RXNE ビットと同時に立ち上がります。マルチバッファ通信の場合、LPUARTx_CR3 レジスタの EIE ビットがセットされている場合に割り込みが発行されます。

FE ビットは、LPUARTx_ICR レジスタの FECF に 1 を書き込むことによってリセットされます。

受信時の設定可能なストップビット

受信するストップビット数は、制御レジスタ 2 の制御ビットを通じて設定でき、通常モードでは 1 または 2 にできます。

- **1 個のストップビット**: ストップビット 1 個のサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
- **2 個のストップビット**: 2 個のストップビットのサンプリングは、2 番目のストップビットの中央で行われます。RXNE および FE フラグは、このサンプルの直後、すなわち 2 番目のストップビット中にセットされます。最初のストップビットでは、フレーミングエラーの検査は行われません。

30.4.4 ボーレート生成

レシーバとトランスミッタ (Rx と Tx) のボーレートは、両方とも、LPUARTx_BRR レジスタでプログラムされたものと同じ値にセットされます。

レシーバとトランスミッタ (Rx と Tx) のボーレートは、両方とも、LPUARTx_BRR レジスタでプログラムされたものと同じ値にセットされます。

$$Tx/Rx \text{ ボー} = \frac{256 \times f_{CK}}{LPUARTDIV}$$

LPUARTDIV は、LPUARTx_BRR レジスタでコード化されます。

注： ボーカウンタは、LPUARTx_BRR への書き込み操作後、ボーレジスタの新しい値によって更新されます。したがって、通信中はボーレートレジスタの値を変更しないでください。

LPUARTx_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。

fck は、[3 x ボーレート, 4096 x ボーレート] の範囲内であればなりません。

表 123. $f_{ck} = 32,768 \text{ KHz}$ でプログラムされたボーレートのエラー計算

| ボーレート | | $f_{ck} = 32,768 \text{ KHz}$ | | |
|-------|-----------|-------------------------------|-----------------------------|--|
| S.No | 目標 | 実際 | ボーレートレジスタに プログラミングされている値 | 誤差 (%) = (計算値 - 目標値) B レート / 目標のB レート |
| 1 | 0.3 Kbps | 0.3 Kbps | 0x6D3A | 0 |
| 2 | 0.6 Kbps | 0.6 Kbps | 0x369D | 0 |
| 3 | 1200 Bps | 1200.087 Bps | 0x1B4E | 0.007 |
| 4 | 2400 Bps | 2400.17 Bps | 0x7 | 0.007 |
| 5 | 4800 Bps | 4801.72 Bps | 0x6D3 | 0.035 |
| 6 | 9600 Kbps | 9608.94 Bps | 0x369 | 0.093 |

30.4.5 マルチプロセッサ通信

LPUART とのマルチプロセッサ通信が可能です（ネットワーク内で複数の LPUART を接続して）。たとえば、1 つの LPUART をマスタとして、その TX 出力を別の LPUART の RX 入力に接続することができます。別の USART はスレーブであり、それぞれの TX 出力の論理積をとった上でマスタの RX 入力に接続します。

マルチプロセッサ設定では、多くの場合、メッセージの本来の受信者のみがメッセージ内容の全体を能動的に受信することが望ましく、これによって対象外の受信者に対する LPUART サービスの余分なオーバーヘッドを減らすことができます。

対象外のデバイスは、ミュート機能によってミュートモードにできます。ミュートモード機能を使用するためには、LPUARTx_CR1 レジスタの MME ビットをセットする必要があります。

ミュートモードでは、

- 受信ステータスビットはセットできません。
- 受信割り込みはすべて禁止されます。
- LPUARTx_ISR レジスタの RWU ビットは 1 にセットされます。特定の条件下では、LPUARTx_RQR レジスタの MMRQ ビットを通じて、RWU をハードウェアまたはソフトウェアによって自動的に制御できます。

LPUART は、LPUART_CR1 レジスタの WAKE ビットの設定に応じて、次のいずれかの方法でミュートモードに入ったり終了したりできます。

- WAKE ビットがリセットされている場合は、アイドルライン検出
- WAKE ビットがセットされている場合は、アドレスマーク検出

アイドルライン検出 (WAKE=0)

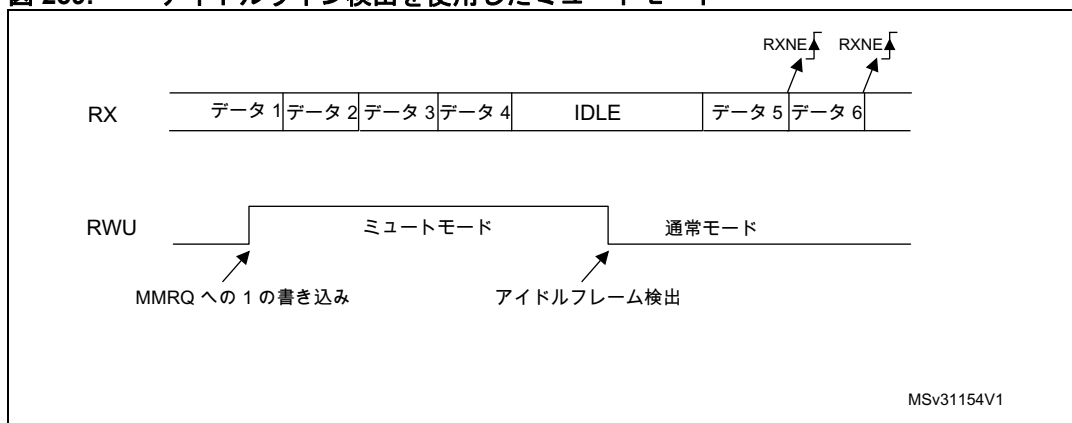
MMRQ ビットに 1 が書き込まれ、RWU が自動的にセットされたときには、LPUART はミュートモードに入ります。

USART は、アイドルフレームを検出するとウェイクアップします。その後、RWU ビットはハードウェアによってクリアされますが、LPUARTx_ISR レジスタの IDLE ビットはセットされません。アイドルライン検出を使用したミュートモードの動作例を [図 237](#) に示します。



この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

図 259. アイドルライン検出を使用したミュートモード



注： IDLE キャラクタがすでに経過しているときに MMRQ がセットされた場合は、ミュートモードに入りません (RWU はセットされません)。

ラインが IDLE のときに LPUART が有効にされた場合、1 IDLE フレーム後にアイドル状態が検出されます (1 キャラクタフレームの受信後だけでなく)。

4 ビット/7 ビットアドレスマーク検出 (WAKE=1)

このモードでは、MSB が 1 の場合、バイトはアドレスとして認識され、そうでない場合はデータとみなされます。アドレスバイトのうち、ターゲットレシーバのアドレスは 4 または 7 LSB です。7 または 4 ビットアドレス検出の選択は、ADDm7 ビットを使用して行われます。この 4 ビット/7 ビットワードは、レシーバによって、LPUARTx_CR2 レジスタの ADD ビットでプログラムされたレシーバの自己アドレスと比較されます。

注： 7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス (ADD[5:0] および ADD[7:0]) で行われます。

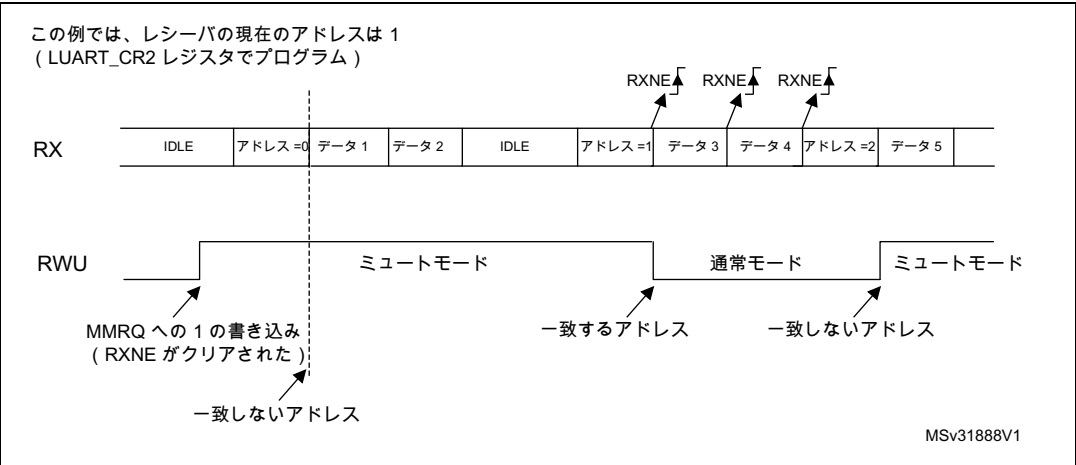
プログラミングされたアドレスと一致しないアドレスキャラクタを受信すると、LPUART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。LPUART がミュートモードに入ったときには、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

MMRQ ビットに 1 が書き込まれたときにも、LPUART はミュートモードに入ります。この場合、RWU ビットも自動的にセットされます。

プログラムされたアドレスに一致するアドレスキャラクタを受信すると、LPUART はミュートモードを終了します。続いて RWU ビットがクリアされ、それ以降のバイトは通常どおりに受信されます。RWU ビットはクリアされているので、アドレスキャラクタに対して RXNE ビットがセットされます。

アドレスマーク検出を使用したミュートモードの動作例を [図 238](#) に示します。

図 260. アドレスマーク検出を使用したミュートモード



30.4.6 パリティ制御

パリティ制御（送信中のパリティビット生成と受信中のパリティ検査）を有効にするには、LPUARTx_CR1 レジスタの PCE ビットをセットします。M ビットによって定義されたフレーム長に応じて、可能な LPUART フレームフォーマットを [表 120](#) に示します。

表 124. フレームフォーマット

| M ビット | PCE ビット | LPUART フレーム ⁽¹⁾ |
|-------|---------|----------------------------|
| 00 | 0 | SB 8 ビットデータ STB |
| 00 | 1 | SB 7 ビットデータ PB STB |
| 01 | 0 | SB 9 ビットデータ STB |
| 01 | 1 | SB 8 ビットデータ PB STB |
| 10 | 0 | SB 7 ビットデータ STB |
| 10 | 1 | SB 6 ビットデータ PB STB |

- 凡例：SB：スタートビット、STB：ストップビット、PB：パリティビット。
- データレジスタでは、PB は常に MSB 位置を取ります（M ビットの値に応じて、9、8、または 7 番目）。

偶数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が偶数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、偶数パリティが選択された場合（LPUARTx_CR1 の PS ビット = 0）、パリティビットは 0 になります。

奇数パリティ

パリティビットは、6、7、または 8 LSB ビット（M ビットの値に応じて）とパリティビットで構成されるフレーム内で「1」の数が奇数になるように計算されます。

たとえば、データ = 00110101 であり、4 ビットがセットされた場合、奇数パリティが選択された場合（LPUARTx_CR1 の PS ビット = 1）、パリティビットは 1 になります。

受信中のパリティチェック

パリティチェックに失敗した場合、LPUARTx_ISR レジスタの PE フラグがセットされ、LPUARTx_CR1 レジスタの PEIE ビットがセットされている場合は割り込みが生成されます。PE フラグは、LPUARTx_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによりクリアされます。

送信中のパリティ生成

LPUARTx_CR1 の PCE ビットがセットされている場合、データレジスタに書き込まれたデータの MSB ビットは送信されますが、パリティビットによって変更されます（偶数パリティが選択された場合（PS=0）は偶数個の「1」、奇数パリティが選択された場合（PS=1）は奇数個の「1」）。

30.4.7 単線半二重通信

単線半二重モードを選択するには、LPUARTx_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットをクリアされた状態に保つ必要があります。

- LPUARTx_CR2 レジスタの LINEN および CLKEN ビット
- LPUARTx_CR3 レジスタの SCEN および IREN ビット

LPUART は、単線半二重のプロトコルに従うように設定できます。この場合、TX ラインと RX ラインは内部接続されます。半二重通信と全二重通信の選択は、LPUART_CR3 レジスタの制御ビット HDSEL で行います。

HDSEL ビットに 1 が書き込まれると、

- TX ラインと RX ラインが内部接続されます。
- RX ピンは使用されなくなります。
- データが送信されないときには、TX ピンは常に解放されます。したがって、アイドル時や受信時には標準入出力として機能します。つまり、TX が外部プルアップ付きの代替機能オープンドレインとして設定されるように、I/O を設定する必要があります。

この点を除くと、通信プロトコルは通常の LPUART モードと同じです。ラインの競合はソフトウェアによって管理する必要があります（たとえば、集中型アービタを使用して）。特に、送信がハードウェアによってブロックされることはなく、TE ビットがセットされている間は、データレジスタにデータが書き込まれるとすぐに、送信が続行されます。

注： LPUART では、1 個のストップビット設定の場合、ストップビットの中央で RXNE フラグがセットされます。

30.4.8 DMA を使用した連続通信

LPUART は、DMA を使用して連続通信を行うことができます。Rx バッファと Tx バッファに対する DMA リクエストは、それぞれ独立して生成できます。

注： [セクション 30.4.3](#) で説明されているように、LPUART を使用します。連続通信を行うには、LPUARTx_ISR レジスタの TXE/ RXNE フラグをクリアします。

DMA を使用した送信

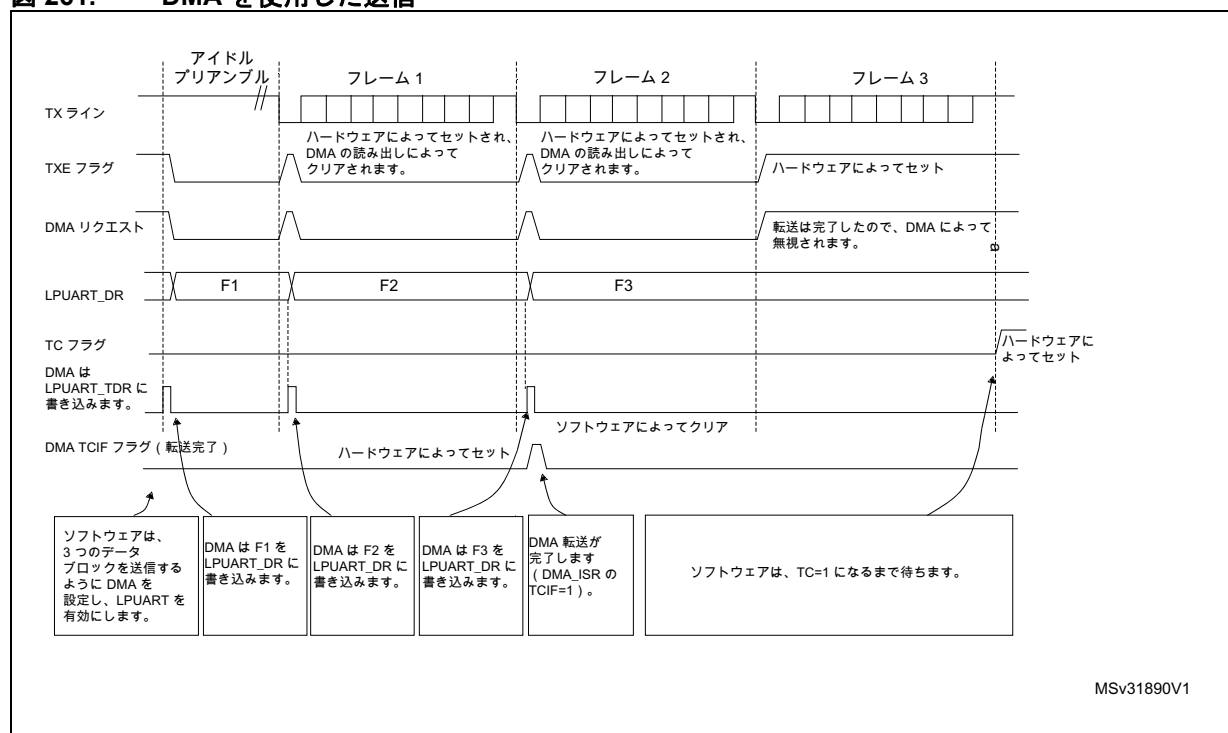
DMA モードでの送信を有効にするには、LPUARTx_CR3 レジスタの DMAT ビットをセットします。TXE ビットがセットされるたびに、データは、DMA ペリフェラル（[セクション 11：ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)）を使用して設定された SRAM 領域から LPUARTx_TDR レジスタにロードされます。DMA チャンネルを LPUART 送信用に割り付けるには、次の手順を実行します（x はチャンネル番号を示します）

1. DMA 制御レジスタに LPUARTx_TDR レジスタのアドレスを書き込み、これを転送先として設定します。データは、各 TXE イベント後にメモリからこのアドレスに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送元として設定します。データは、各 TXE イベント後に、このメモリ領域から LPUARTx_TDR レジスタにロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャンネル優先順位を DMA レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の DMA 割り込み生成を設定します。
6. LPUARTx_ICR レジスタの TCCF ビットをセットすることによって、LPUARTx_ISR レジスタの TC フラグをクリアします。
7. DMA レジスタのチャンネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャンネルの割り込みベクタに基づいて割り込みを生成します。

送信モードでは、送信すべきすべてのデータを DMA が書き込むと (DMA_ISR レジスタの TCIF フラグがセットされます)、TC フラグを観察して LPUART 通信の完了を確認することができます。これは、LPUART を無効にしたり STOP モードに入ったりする前に、最後の送信が壊れないようにするために必要です。ソフトウェアは、TC=1 になるまで待つ必要があります。TC フラグは、すべてのデータ転送中、クリアされたままであり、最後のフレームの送信終了時にハードウェアによってセットされます。

図 261. DMA を使用した送信



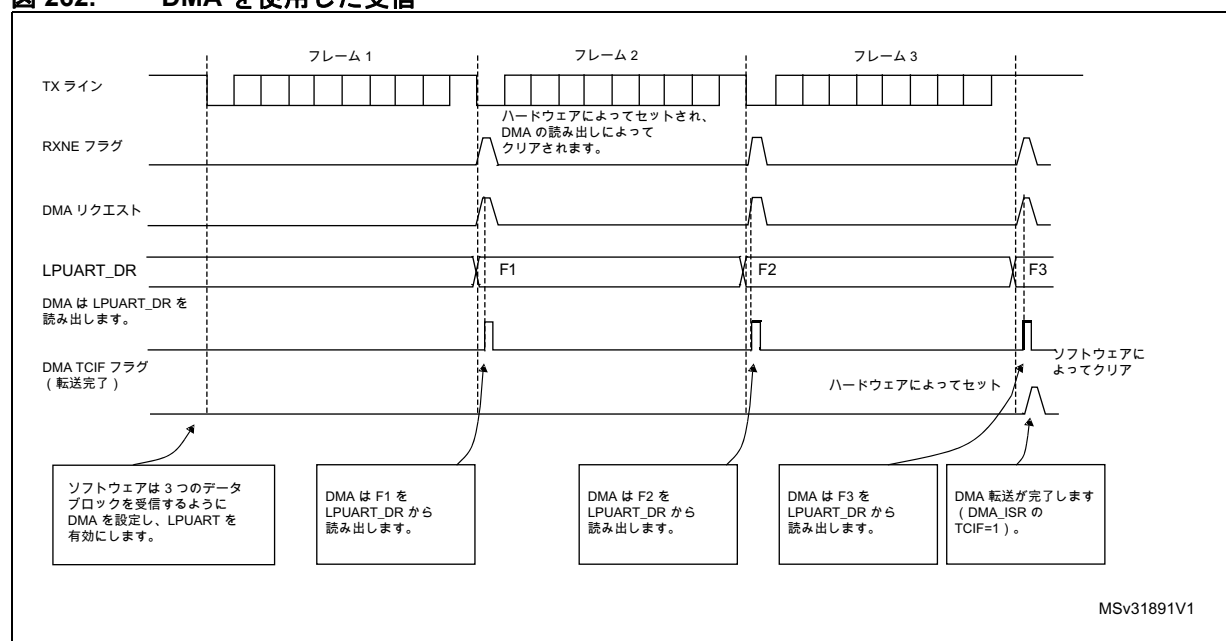
DMA を使用した受信

DMA モードでの受信を有効にするには、LPUARTx_CR3 レジスタの DMAR ビットをセットします。データは、データバイトが受信されると、LPUARTx_RDR レジスタから、DMA ペリフェラルを使用して設定された SRAM 領域（[セクション 11：ダイレクトメモリアクセスコントローラ \(DMA\) \(232 ページ\)](#)）を参照）にロードされます。DMA チャンネルを LPUART 受信用に割り付けるには、次の手順を実行します。

1. DMA 制御レジスタに LPUARTx_RDR レジスタのアドレスを書き込み、これを転送元として設定します。データは、各 RXNE イベント後に、このアドレスからメモリに移動されます。
2. DMA 制御レジスタにメモリアドレスを書き込み、これを転送先として設定します。データは、各 RXNE イベント後に、LPUARTx_RDR からこのメモリ領域にロードされます。
3. 転送すべきバイト総数を DMA 制御レジスタに設定します。
4. チャンネル優先順位を DMA 制御レジスタで設定します。
5. アプリケーションで必要とされる 1/2 転送終了、転送完了後の割り込み生成を設定します。
6. DMA 制御レジスタのチャンネルを有効にします。

DMA コントローラにプログラミングされたデータ転送数に達すると、DMA コントローラは、DMA チャンネルの割り込みベクタに基づいて割り込みを生成します。

図 262. DMA を使用した受信



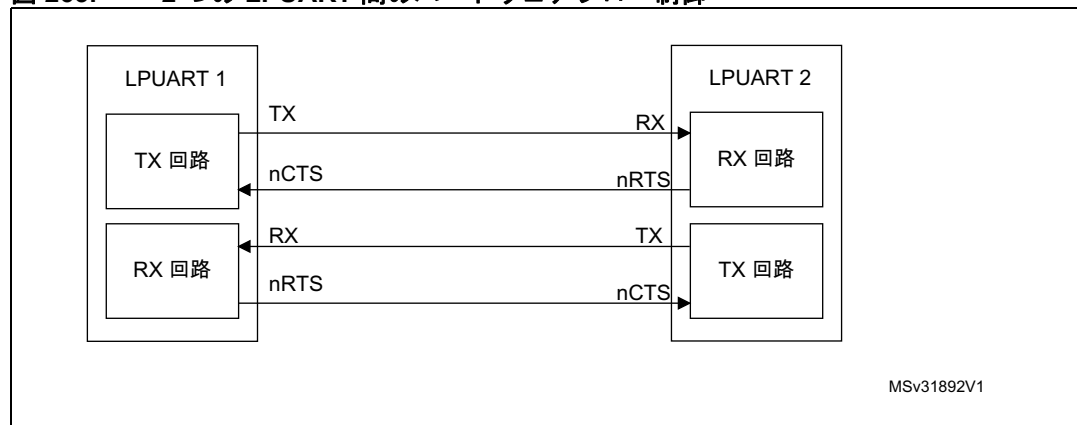
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信でトランザクション中にエラーが発生した場合、現在のバイトの後でエラーフラグがアサートされます。割り込み有効フラグがセットされている場合、割り込みが生成されます。1 バイト受信において RXNE とともにアサートされるフレーミングエラー、オーバーランエラー、およびノイズフラグに関しては、別のエラーフラグ割り込み有効ビット (LPUARTx_CR3 レジスタの EIE ビット) があり、これがセットされている場合、いずれかのエラーが発生すると、現在のバイトの後で割り込みが有効になります。

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

nCTS 入力と nRTS 出力を使用すると、2 つのデバイス間でシリアルデータフローを制御できます。[図 251](#) に、このモードで 2 つのデバイスを接続する方法を示します。

図 263. 2 つの LPUART 間のハードウェアフロー制御

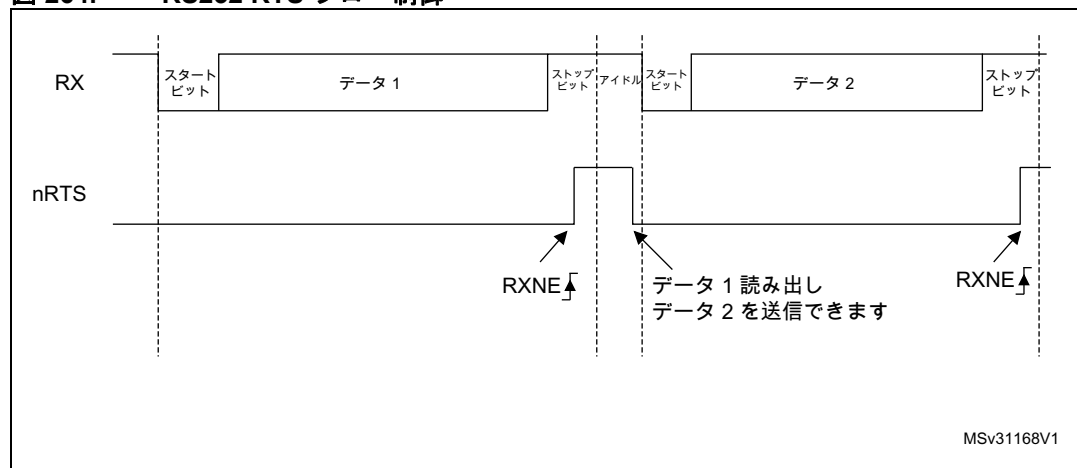


RS232 RTS と CTS のフロー制御は、LPUARTx_CR3 レジスタの RTSE ビットと CTSE ビットにそれぞれ 1 を書き込むことによって、個別に有効にできます。

RS232 RTS フロー制御

RTS フロー制御が有効な場合 (RTSE=1)、LPUART レシーバが新しいデータを受信可能である限り、nRTS がアサートされます (ローレベル接続)。受信レジスタが満杯になると nRTS がネゲートされ、現在のフレームの終わりに送信が停止する予定であることを示します。[図 252](#) に、RTS フロー制御が有効な場合の通信例を示します。

図 264. RS232 RTS フロー制御

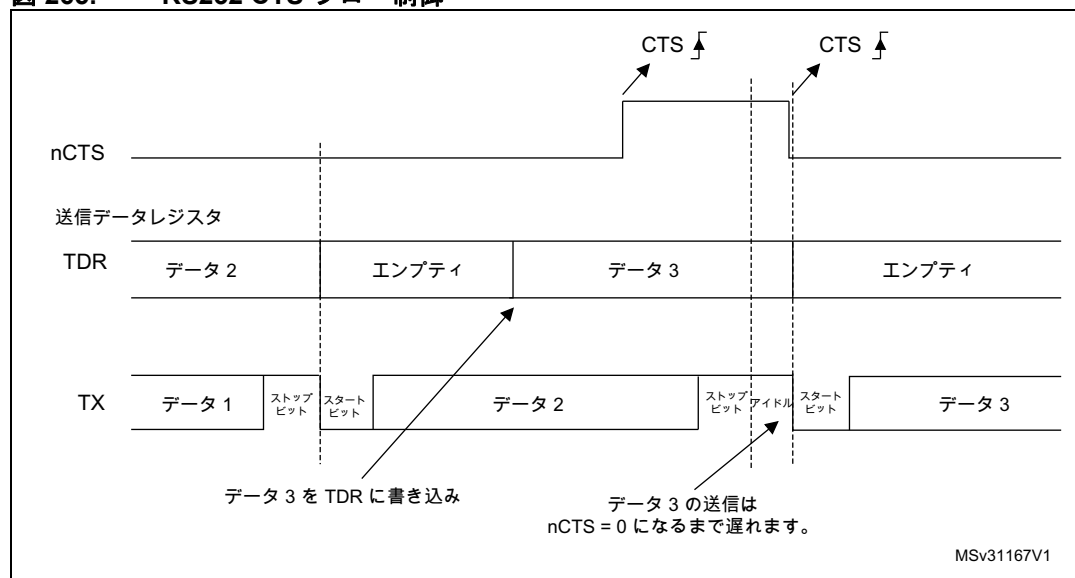


RS232 CTS フロー制御

CTS フロー制御が有効な場合 (CTSE=1)、トランスミッタは、nCTS 入力を検査してから、次のフレームを送信します。nCTS がアサートされた場合 (ローレベル接続)、次のデータが送信されます (データが送信されると想定、つまり TXE=0 の場合)。そうでない場合、送信は行われません。送信中に nCTS がネゲートされると、現在の送信が完了してから、トランスミッタが停止します。

CTSE=1 の場合、nCTS 入力が入力がトグルすると、CTSIF ステータスビットはハードウェアによって自動的にセットされます。このビットは、レシーバの通信準備ができているかどうかを示します。LPUARTx_CR3 レジスタの CTSIE ビットがセットされている場合、割り込みが生成されます。[図 253](#) に、CTS フロー制御が有効な場合の通信例を示します。

図 265. RS232 CTS フロー制御



注： 正しい動作のために、nCTS は、現在のキャラクタの終了の少なくとも 3 LPUART クロックソース周期前にアサートする必要があります。さらに、2 x PCLK 周期より短いパルスでは CTSCF フラグがセットされない場合があることに注意してください。

RS485 ドライバ有効

ドライバ有効機能を有効にするには、LPUARTx_CR3 制御レジスタのビット DEM をセットします。これにより、DE (Driver Enable) 信号によって外部トランシーバ制御を有効にできます。アサーション時間は、DE 信号の有効化から START ビットの開始までの時間です。LPUARTx_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。ネゲート時間は、送信メッセージの最後のストップビットの終了から DE 信号の無効化までの時間です。LPUARTx_CR1 制御レジスタの DEAT [4:0] ビットフィールドを使用してプログラムされます。DE 信号の極性は、LPUARTx_CR3 制御レジスタの DEP ビットを使用して設定できます。

LPUART では、DEAT および DEDT は USART クロックソース (f_{CK}) サイクルで表されます。

- ドライバ有効アサーション時間 =
 - $(1 + (DEAT \times P)) \times f_{CK}$, if $P \neq 0$
 - $(1 + DEAT) \times f_{CK}$, if $P = 0$
- ドライバ有効ネゲート時間 =
 - $(1 + (DEDT \times P)) \times f_{CK}$, if $P \neq 0$
 - $(1 + DEDT) \times f_{CK}$, if $P = 0$

$P = BRR[14:11]$

30.4.10 STOP モードからのウェイクアップ

LPUART は、UESM ビットがセットされ、LPUART クロックが HSI または LSE にセットされているとき、MCU を STOP モードからウェイクアップできます（リセットおよびクロック制御 (RCC) のセクションを参照してください）。

STOP モードからの MCU のウェイクアップは、標準 RXNE 割り込みを使用して行うことができます。この場合、STOP モードに入る前に RXNEIE ビットをセットする必要があります。

または、WUS ビットフィールドによって、特定の割り込みを選択することもできます。

MCU を STOP モードからウェイクアップできるようにするためには、STOP モードに入る前に、LPUARTx_CR1 制御レジスタの UESM ビットをセットする必要があります。

ウェイクアップイベントが検出されると、ハードウェアによって WUF フラグがセットされ、WUFIE ビットがセットされていた場合はウェイクアップ割り込みが生成されます。

注： STOP モードに入る前に、ユーザは LPUART が転送を行っていないことを確認する必要があります。BUSY フラグでは、受信中に STOP モードに入らないことを保証できません。

WUF フラグは、MCU が STOP モードか、アクティブモードかに関係なく、ウェイクアップイベントが検出されたときにセットされます。

初期化とレシーバの有効化の直後に STOP モードに入るときには、REACK ビットをチェックして、LPUART が実際に有効であることを確認する必要があります。

受信に DMA が使用されるときには、STOP モードに入る前に無効化し、STOP モードの終了時に再び有効にする必要があります。

STOP モードからのウェイクアップは、すべてのモードで使用できるわけではありません。たとえば、SPI はマスタモードでのみ動作するので、SPI モードでは機能しません。

STOP モードでのミュートモードの使用

LPUART は、STOP モードに入る前にミュートモードになります。

- アイドル検出は STOP モードでは機能しないので、アイドル検出時にミュートモードからウェイクアップすることはできません。
- アドレス一致によるミュートモードからのウェイクアップが使用される場合、STOP モードからのウェイクアップのソースもアドレス一致でなければなりません。STOP モードに入るときに RXNE フラグがセットされる場合、アドレス一致によって STOP からウェイクアップしても、インタフェースはミュートモードのままです。
- LPUART が START ビット検出時に MCU を STOP モードからウェイクアップするように設定された場合、WUF フラグはセットされますが、RXNE フラグはセットされません。

30.5 LPUART 割り込み

表 125. LPUART 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|-------------------------------|---------|------------|
| 送信データレジスタエンプティ | TXE | TXEIE |
| CTS 割り込み | CTSIF | CTSIE |
| 送信完了 | TC | TCIE |
| 受信データレジスタノットエンプティ（データの読み出し可能） | RXNE | RXNEIE |
| オーバーランエラー検出 | ORE | |



表 125. LPUART 割り込みリクエスト（続き）

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|---|----------------------|------------|
| アイドルライン検出 | IDLE | IDLEIE |
| パリティエラー | PE | PEIE |
| LIN ブレーク | LBDF | LBDIE |
| マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー。 | NF または ORE または FE | EIE |
| キャラクター致 | CMF | CMIE |
| STOP モードからのウェイクアップ | WUF ⁽¹⁾ | WUFIE |

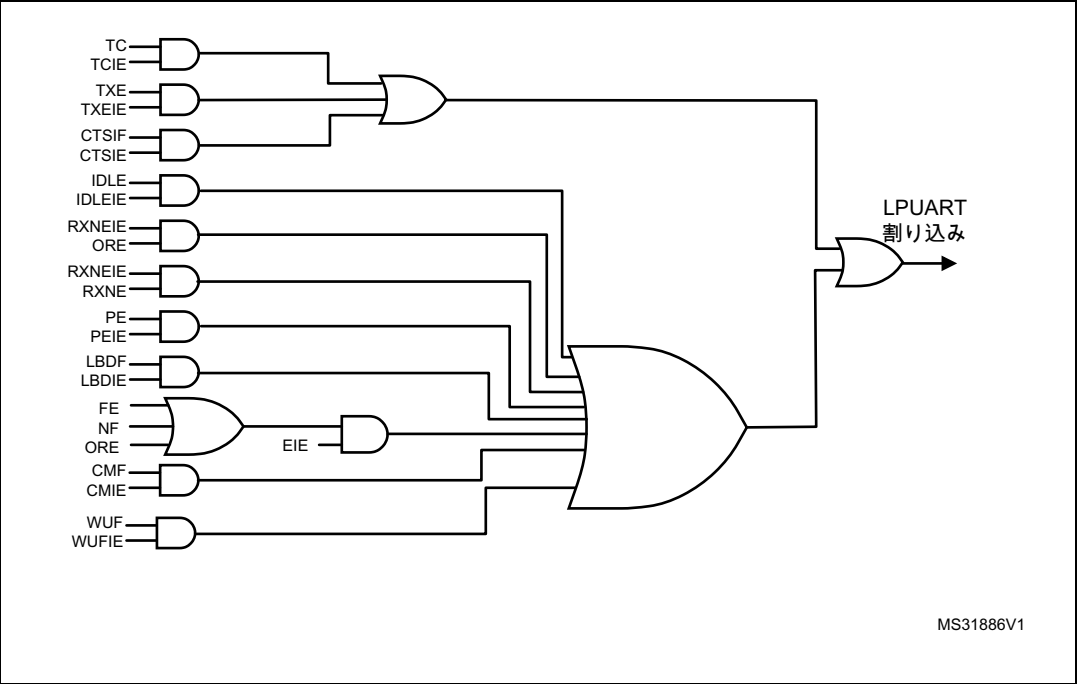
1. WUF 割り込みは、STOP モードでのみアクティブです。

LPUART 割り込みイベントは、同じ割り込みベクターに接続されます（[図 254](#) を参照）。

- 送信時：送信完了、Clear to Send、送信データレジスタエンプティ、またはフレーミングエラー割り込み。
- 受信時：アイドルライン検出、オーバーランエラー、受信データレジスタノットエンプティ、パリティエラー、LIN ブレーク検出、ノイズフラグ、フレーミングエラー、キャラクター致など。

これらのイベントは、対応する有効制御ビットがセットされている場合に割り込みを生成します。

図 266. LPUART 割り込みマッピング図



この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

30.6 LPUART レジスタ

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

30.6.1 制御レジスタ 1 (LPUARTx_CR1)

アドレスオフセット : 0x00

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|----|------|------|-----------|------|-------|------|--------|-----------|----|----|------|----|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | M1 | Res. | Res. | DEAT[4:0] | | | | | DEDT[4:0] | | | | |
| | | | rW | | | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | CMIE | MME | M0 | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | UESM | UE |
| | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW |

ビット 31:29 予約済みであり、リセット値のままにしておかなければなりません。

ビット 28 **M1** : ワード長

このビットとビット 12 (M0) によって、ワード長が決まります。ソフトウェアによってセット/クリアされます。

M[1:0] = 00: スタートビット 1 個、データビット 8 個、ストップビット n 個

M[1:0] = 01: スタートビット 1 個、データビット 9 個、ストップビット n 個

M[1:0] = 10: スタートビット 1 個、データビット 7 個、ストップビット n 個

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

注 : 7 ビットデータ長モードでは、スマートカードモード、LIN マスタモード、および自動ポーレート (0x7F および 0x55 フレーム検出) はサポートされません。

ビット 27 予約済みであり、リセット値のままにしておかなければなりません。

ビット 26 予約済みであり、リセット値のままにしておかなければなりません。

ビット 25:21 **DEAT[4:0]** : ドライバ有効アサーション時間

この 5 ビット値は、DE (Driver Enable) 信号の有効化からスタートビットの開始までの時間を定義します。UCLK (USART クロック) クロックサイクル数で表されます。詳細については、RS485 ドライバ有効の段落を参照してください。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 20:16 **DEDT[4:0]** : ドライバ有効ネゲート時間

この 5 ビット値は、送信メッセージの最後のストップビットの終了から DE (Driver Enable) 信号の無効化までの時間を定義します。UCLK (USART クロック) クロックサイクル数で表されます。詳細については、RS485 ドライバ有効の段落を参照してください。

DEDT 時間中に LPUARTx_TDR レジスタに書き込みが行われた場合、DEDT 時間と DEAT 時間の両方が経過するまで、新しいデータは送信されません。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 15 予約済みであり、リセット値のままにしておかなければなりません。

ビット 14 **CMIE** : キャラクター一致割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの CMF ビットがセットされると、LPUART 割り込みが生成されます。



ビット 13 MME : ミュートモード有効

このビットは、LPUART のミュートモード機能を有効にします。セットされると、LPUART は、WAKE ビットの定義に従って、アクティブモードとミュートモードを切り替えることができます。ソフトウェアによってセット／クリアされます。

0 : レシーバは永続的にアクティブモードです。

1 : レシーバはミュートモードとアクティブモードを切り替えることができます。

ビット 12 M0 : ワード長

このビットとビット 28 (M1) によって、ワード長が決まります。ソフトウェアによってセット／クリアされます。ビット 28 (M1) の説明を参照してください。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 11 WAKE : レシーバウェイクアップ方式

このビットによって、ミュートモードからの LPUART のウェイクアップ方式が決まります。ソフトウェアによってセット／クリアされます。

0 : アイドルライン

1 : アドレスマーク

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 10 PCE : パリティ制御有効

このビットは、ハードウェアのパリティ制御 (生成と検出) を選択します。パリティ制御が有効なとき、算出されたパリティは MSB 位置 (M=1 の場合はビット 9、M=0 の場合はビット 8) に挿入され、受信されたデータではパリティが検査されます。このビットは、ソフトウェアによってセット／クリアされます。このビットがセットされると、送受信において現在のバイトの後で PCE が有効になります。

0 : パリティ制御は無効です。

1 : パリティ制御は有効です。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 9 PS : パリティ選択

このビットは、パリティの生成／検出が有効である (PCE ビットがセットされている) とき、奇数パリティ／偶数パリティを選択します。ソフトウェアによってセット／クリアされます。パリティは、現在のバイトの後で選択されます。

0 : 偶数パリティ

1 : 奇数パリティ

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 8 PEIE : PE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタで PE=1 のときには LPUART 割り込みが生成されます。

ビット 7 TXEIE : 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの TXE=1 のときには LPUART 割り込みが生成されます。

ビット 6 TCIE : 転送完了割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの TC=1 のときには LPUART 割り込みが生成されます。

ビット 5 RXNEIE : RXNE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの ORE=1 または RXNE=1 のときには、LPUART 割り込みが生成されます。

ビット 4 IDLEIE : IDLE 割り込み有効

このビットは、ソフトウェアによってセット／クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの IDLE=1 のときには LPUART 割り込みが生成されます。

ビット 3 TE : トランスミッタ有効

このビットは、トランスミッタを有効にします。ソフトウェアによってセット／クリアされます。

0 : トランスミッタは無効です。

1 : トランスミッタは有効です。

注： 送信中に TE ビットに「0」パルスを与える（「0」に続けて「1」を書き込む）と、現在のワードの後にブリアンブル（アイドルライン）が送信されます。アイドルキャラクタを生成するためには、すぐには TE に 1 を書き込まないでください。必要な時間を確保するために、ソフトウェアは LPUARTx_ISR レジスタの TEACK ビットをポーリングできます。

TE がセットされると、送信が開始されるまでに 1 ビット時間の遅れが生じます。

ビット 2 RE : レシーバ有効

このビットは、レシーバを有効にします。ソフトウェアによってセット／クリアされます。

0 : レシーバは無効です。

1 : レシーバは有効であり、スタートビットの検索が開始されます。

ビット 1 UESM : STOP モードでの LPUART 有効

このビットがクリアされると、LPUART は MCU を STOP モードからウェイクアップできません。

このビットがセットされると、LPUART は MCU を STOP モードからウェイクアップできますが、LPUART クロック選択が RCC にて HSI または LSE であることが条件です。

このビットは、ソフトウェアによってセット／クリアされます。

0 : LPUART は STOP モードから MCU をウェイクアップできません。

1 : LPUART は STOP モードから MCU をウェイクアップできます。この機能がアクティブなとき、LPUART のクロックソースは HSI または LSE でなければなりません（[セクション 7：リセットおよびクロック制御 \(RCC\)](#) を参照）。

注： STOP モードに入る直前に UESM ビットをセットし、STOP モードの終了時にクリアすることが推奨されます。

ビット 0 UE : LPUART 有効

このビットがクリアされると、LPUART プリスケアラと出力はただちに停止され、現在の操作は破棄されます。LPUART の設定は保たれますが、LPUARTx_ISR のステータスフラグはすべてリセットされます。このビットは、ソフトウェアによってセット／クリアされます。

0 : LPUART プリスケアラと出力は無効であり、低電力モードです。

1 : LPUART は有効です。

注： ラインにエラーを生成せずに低電力モードに入るためには、TE ビットをリセットする必要があります。ソフトウェアは LPUARTx_ISR の TC ビットがセットされるのを待ってから、UE ビットをリセットする必要があります。

UE=0 のときには DMA リクエストもリセットされるので、UE ビットをリセットする前に DMA チャンネルを無効にする必要があります。

30.6.2 制御レジスタ 2（LPUARTx_CR2）

アドレスオフセット：0x04

リセット値：0x0000

| | | | | | | | | | | | | | | | |
|----------|------|-----------|----|----------|------|------|------|------|------|------|-------|----------|---------|-------|-------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| ADD[7:4] | | | | ADD[3:0] | | | | Res. | Res. | Res. | Res. | MSBFIRST | DATAINV | TXINV | RXINV |
| rw | rw | rw | rw | rw | rw | rw | rw | | | | | rw | rw | rw | rw |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| SWAP | Res. | STOP[1:0] | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADDM7 | Res. | Res. | Res. | Res. |
| rw | | rw | rw | | | | | | | | rw | | | | |

ビット 31:28 ADD[7:4]:LPUART ノードのアドレス
このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。
これは、マルチプロセッサ通信において、7 ビットアドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。トランスミッタによって送信されるキャラクタの MSB は 1 でなければなりません。通常の受信時、ミュートモードが無効なときに（たとえば、ModBus プロトコルのブロック終了検出）、キャラクタ検出のために使用することができます。この場合、受信されたキャラクタ全体（8 ビット）が ADD[7:0] 値と比較され、一致した場合は CMF フラグがセットされます。
このビットフィールドは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 27:24 ADD[3:0]:LPUART ノードのアドレス
このビットフィールドは、認識される LPUART ノードのアドレスまたはキャラクタコードを指定します。
これは、マルチプロセッサ通信において、アドレスマーク検出によるウェイクアップのために、ミュートモードまたは STOP モード時に使用されます。
このビットフィールドは、受信が無効のとき（RE=0）または LPUART が無効のとき（UE=0）のみ、書き込むことができます。

ビット 23:20 予約済みであり、リセット値のままにしておかなければなりません。

ビット 19 MSBFIRST：最上位ビットファースト
このビットは、ソフトウェアによってセット／クリアされます。
0：スタートビットに続いて、データはデータビット 0 から順に送受信されます。
1：スタートビットに続いて、データは MSB（ビット 7/8/9）から順に送受信されます。
このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 18 DATAINV：バイナリデータ反転
このビットは、ソフトウェアによってセット／クリアされます。
0：データレジスタからの論理データは正／ダイレクトロジックで送受信されます。（1=H、0=L）
1：データレジスタからの論理データは、負／インバースロジックで送受信されます。（1=L、0=H）パリティビットも反転されます。
このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 17 TXINV：TX ピンアクティブレベル反転
このビットは、ソフトウェアによってセット／クリアされます。
0：TX ピン信号は標準ロジックレベルを使用して機能します（V_{DD}=1/アイドル、Gnd=0/マーク）。
1：TX ピン信号値は反転されます。（V_{DD}=0/マーク、Gnd=1/アイドル）。
これにより、TX ラインで外部インバータを使用できます。
このビットフィールドは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 16 RXINV : RX ピンアクティブレベル反転

このビットは、ソフトウェアによってセット／クリアされます。

0 : RX ピン信号は標準ロジックレベルを使用して機能します ($V_{DD} = 1$ /アイドル、Gnd=0/マーク)。

1 : RX ピン信号値は反転されます。 ($V_{DD} = 0$ /マーク、Gnd=1/アイドル)。

これにより、RX ラインで外部インバータを使用できます。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 15 SWAP : TX/RX ピンのスワップ

このビットは、ソフトウェアによってセット／クリアされます。

0 : TX/RX ピンは標準ピンアウトでの定義に従って使用されます。

1 : TX および RX ピンの機能はスワップされます。これにより、別の USART へのクロスワイヤ接続の場合に動作できます。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 14 予約済みであり、リセット値のままにしておかなければなりません。**ビット 13:12 STOP[1:0]** : STOP ビット

このビットは、ストップビットのプログラミングに使用します。

00 : 1 個のストップビット

01 : 予約済み。

10 : 2 個のストップビット

11 : 予約済み

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 11:5 予約済みであり、リセット値のままにしておかなければなりません。**ビット 4 ADDM7** : 7 ビットアドレス検出／4 ビットアドレス検出

このビットは、4 ビットアドレス検出と 7 ビットアドレス検出の選択に使用されます。

0 : 4 ビットアドレス検出

1 : 7 ビットアドレス検出 (8 ビットデータモード)

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

注 : 7 ビットおよび 9 ビットデータモードでは、アドレス検出は、それぞれ 6 ビットおよび 8 ビットアドレス (ADD[5:0] および ADD[7:0]) に対して行われます。

ビット 3:0 予約済みであり、リセット値のままにしておかなければなりません。

30.6.3 制御レジスタ 3 (LPUARTx_CR3)

アドレスオフセット : 0x08

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|---------|------|-------|------|------|-------|-------|----------|------|--------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UCESM | WUFIE | WUS[2:0] | | Res. | Res. | Res. | Res. |
| | | | | | | | | rw | rw | rw | rw | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DEP | DEM | DDRE | OVR DIS | Res. | CTSIE | CTSE | RTSE | DMAT | DMAR | Res. | Res. | HD SEL | Res. | Res. | EIE |
| rw | rw | rw | rw | | rw | rw | rw | rw | rw | | | rw | | | rw |

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

ビット 23 UCESM : STOP モードでの LPUART クロック有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : STOP モードでの LPUART/USART クロック無効

1 : STOP モードでの LPUART/USART クロック有効

注 : この制御ビットは、LPUART/USART クロックソースが HSI 16 または LSE の場合にのみ使用されます。

ビット 22 WUFIE : STOP モードからのウェイクアップ割り込み有効

このビットは、ソフトウェアによってセット/クリアされます。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの WUF=1 のときには LPUART 割り込みが生成されます。

注 : WUFIE は、STOP モードに入る前にセットする必要があります。

WUF 割り込みは、STOP モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 21:20 WUS[1:0] : STOP モードからのウェイクアップ割り込みフラグ選択

このビットフィールドは、WUF を有効にするイベントを指定します (STOP モードからのウェイクアップフラグ)。

00 : WUF はアドレス一致時に有効になります (ADD[7:0] および ADDM7 による定義に従って)。

01 : 予約済み

10 : WUF はスタートビット検出時に有効になります。

11 : WUF は RXNE 時に有効になります。

このビットフィールドは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

注 : LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19:16 予約済みであり、リセット値のままにしておかなければなりません。

ビット 15 DEP : ドライバ有効極性選択

0 : DE 信号はアクティブハイです。

1 : DE 信号はアクティブローです。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 14 DEM : ドライバ有効モード

このビットにより、DE 信号によって外部トランシーバ制御を有効にできます。

0 : DE 機能は無効です。

1 : DE 機能は有効です。DE 信号は RTS ピンで出力されます。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 13 DDRE : 受信エラー時 DMA 無効

0 : 受信エラーの場合、DMA は無効になります。対応するエラーフラグはセットされますが、RXNE は 0 に保たれ、オーバーランを防ぎます。結果として、DMA リクエストはアサートされないで、エラーのあるデータは転送されず (DMA リクエストなし)、次の正しい受信データが転送されます。

1 : 受信エラーの後、DMA は無効化されます。対応するエラーフラグと RXNE がセットされます。エラーフラグがクリアされるまで、DMA リクエストはマスクされます。つまり、ソフトウェアは、まず、DMA リクエストを無効にするか (DMAR=0)、RXNE をクリアしてから、エラーフラグをクリアする必要があります。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

注： 受信エラーは、パリティエラー、フレーミングエラー、またはノイズエラーです。

ビット 12 OVRDIS : オーバーラン無効

このビットは、受信オーバーラン検出を無効にするために使用されます。

0 : オーバーランエラーフラグ、ORE は、受信データが読み出される前に新しいデータを受信したときにセットされます。

1 : オーバーラン機能は無効です。RXNE フラグがまだセットされている間に新しいデータを受信した場合、

ORE フラグはセットされず、新しく受信されたデータが LPUARTx_RDR レジスタの前の内容に上書きされます。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

注： この制御ビットにより、データを読み出さずに通信フローをチェックできます。

ビット 11 予約済みであり、リセット値のままにしておかなければなりません。**ビット 10 CTSIE : CTS 割り込み有効**

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの CTSIF=1 のときには、割り込みが生成されます。

ビット 9 CTSE : CTS 有効

0 : CTS ハードウェアフロー制御が無効です。

1 : CTS モードが有効です。データは nCTS 入力のアサート (0 に関係) されている場合にのみ転送されます。データの送信中に nCTS 入力にネグートされた場合、送信は停止前に完了します。nCTS がネグートされている間にデータがデータレジスタに書き込まれた場合、nCTS がアサートされるまで送信は延期されます。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 8 RTSE : RTS 有効

0 : RTS ハードウェアフロー制御が無効です。

1 : RTS 出力は有効であり、レシーババッファにスペースがあるときにのみ、データがリクエストされます。現在の文字が転送された後、データの転送は停止すると期待されます。データを受信できる場合、nRTS 出力がアサートされます (0 にプルされます)。

このビットは、LPUART が無効 (UE=0) のときのみ書き込むことができます。

ビット 7 DMAT : DMA 有効トランスミッタ

このビットは、ソフトウェアでセット/リセットされます。

1 : DMA モードは送信に有効

0 : DMA モードは送信に無効です。

ビット 6 DMAR : DMA 有効レシーバ

このビットは、ソフトウェアでセット/リセットされます。

1 : DMA モードが受信に有効です。

0 : DMA モードが受信に無効です。

ビット 5:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **HDSEL** : 半二重選択

単線半二重モードの選択です。

0 : 半二重モードは選択されません。

1 : 半二重モードが選択されます。

このビットは、LPUART が無効（UE=0）のときのみ書き込むことができます。

ビット 2:1 予約済みであり、リセット値のままにしておかなければなりません。

ビット 0 **EIE** : エラー割り込み有効

エラー割り込み有効ビットは、フレーミングエラー、オーバーランエラー、またはノイズフラグ（LPUARTx_ISR レジスタの FE=1 または ORE=1 または NF=1）の場合に割り込み生成を有効にするために必要です。

0 : 割り込みは禁止されています。

1 : LPUARTx_ISR レジスタの FE=1、ORE=1、または NF=1 のときには、割り込みが生成されます。

30.6.4 **ボーレートレジスタ（LPUARTx_BRR）**

このレジスタは、LPUART が無効（UE=0）のときのみ書き込むことができます。自動ボーレート検出モードでハードウェアによって自動的に更新されます。

アドレスオフセット : 0x0C

リセット値 : 0x0000

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

ビット 31:20 予約済みであり、リセット値のままにしておかなければなりません。

ビット 19:0 **BRR[19:0]**

注 : **LPUARTx_BRR レジスタに 0x300 未満の値を書き込むことは禁じられています。**

ただし、LPUARTx_BRR は >= 0x300 でなければならず、LPUARTx_BRR は 20 ビットであり、高い fck 値を使用して高いボーレートを生成するときには、注意が必要です。fck は [3 x ボーレート , 4096 x ボーレート] の範囲内でなければなりません。

30.6.5 リクエストレジスタ (LPUARTx_RQR)

アドレスオフセット : 0x18

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|------|------|------|-------|------|-------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RXFRQ | MMRQ | SBKRQ | Res. |
| | | | | | | | | | | | | w | w | w | |

ビット 31:4 予約済みであり、リセット値のままにしておかなければなりません。

ビット 3 **RXFRQ** : 受信データー掃リクエスト

このビットに 1 を書き込むと、RXNE フラグがクリアされます。

これにより、受信したデータを読み出さずに破棄して、オーバーラン条件を避けることができます。

ビット 2 **MMRQ** : ミュートモードリクエスト

このビットに 1 を書き込むと、LPUART はミュートモードになり、RWU フラグはリセットされます。

ビット 1 **SBKRQ** : ブレーク送信リクエスト

このビットに 1 を書き込むと、SBKF フラグがセットされ、送信マシンが使用可能になるとすぐに、ラインで BREAK を送信するリクエストが発行されます。

注 : アプリケーションが、まだ送信されていないものも含めて、以前に挿入されたすべてのデータに続いてブレークキャラクタを送信する必要がある場合、ソフトウェアは SBKRQ ビットをセットする前に、TXE フラグのアサートを待つ必要があります。

ビット 0 予約済みであり、リセット値のままにしておかなければなりません。

30.6.6 割り込みおよびステータスレジスタ (LPUARTx_ISR)

アドレスオフセット : 0x1C

リセット値 : 0x00C0

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-------|------|------|--------|--------|------|-----|------|-----|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RE ACK | TE ACK | WUF | RWU | SBKF | CMF | BUSY |
| | | | | | | | | | r | r | r | r | r | r | r |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | CTS | CTSIF | Res. | TXE | TC | RXNE | IDLE | ORE | NF | FE | PE |
| | | | | | r | r | | r | r | r | r | r | r | r | r |

ビット 31:23 予約済みであり、リセット値のままにしておかなければなりません。

ビット 22 **REACK** : 受信有効確認応答フラグ

このビットは、受信有効値が LPUART によって考慮されるときに、ハードウェアによってセット／リセットされます。

これを使用して、STOP モードに入る前に、LPUART が受信できる状態であることを確認できます。

注 : LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。



ビット 21 TEACK : 送信有効確認応答フラグ

このビットは、送信有効値が LPUART によって考慮されるときに、ハードウェアによってセット/リセットされます。

LPUARTx_CR1 レジスタで TE=0 を書き込んだ後、TE=1 を書き込むことによってアイドルフレームリクエストが生成されるとき、TE=0 の最小周期を満たすために使用できます。

ビット 20 WUF : STOP モードからのウェイクアップフラグ

このビットは、ウェイクアップイベントが検出されたときに、ハードウェアによってセットされます。イベントは、WUS ビットフィールドによって定義されます。LPUARTx_ICR レジスタの WUCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR3 レジスタの WUFIE=1 である場合、割り込みが生成されます。

注： UESM がクリアされると、WUF フラグもクリアされます。

WUF 割り込みは、STOP モードでのみアクティブです。

LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 19 RWU : レシーバのミュートモードからのウェイクアップ

このビットは、LPUART がミュートモードかどうかを示します。ウェイクアップ/ミュートシーケンスが認識されたときに、ハードウェアによってクリア/セットされます。ミュートモード制御シーケンス (アドレスまたは IDLE) は、LPUARTx_CR1 レジスタの WAKE ビットによって選択されます。

IDLE モードでのウェイクアップが選択されたとき、このビットは LPUARTx_RQR レジスタの MMRQ ビットに 1 を書き込むことによって、ソフトウェアによってのみセットできます。

0 : レシーバはアクティブモードです。

1 : レシーバはミュートモードです。

注： LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 18 SBKF : ブレーク送信フラグ

このビットは、ブレークキャラクタ送信がリクエストされたことを示します。LPUARTx_CR3 レジスタの SBKRQ ビットに 1 を書き込むことによって、ソフトウェアによってセットされます。ブレーク送信のストップビット時に、ハードウェアによって自動的にリセットされます。

0 : ブレークキャラクタは送信されません。

1 : ブレークキャラクタは送信されます。

ビット 17 CMF : キャラクター致フラグ

このビットは、ADD[7:0] によって定義されたキャラクタが受信されたときに、ハードウェアによってセットされます。LPUARTx_ICR レジスタの CMCF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR1 レジスタの CMIE=1 の場合、割り込みが生成されます。

0 : キャラクター致は検出されていません。

1 : キャラクター致が検出されました。

ビット 16 BUSY : ビジーフラグ

このビットは、ハードウェアによってセット/リセットされます。RX ラインで通信中 (スタートビットの検出時) はアクティブです。成否にかかわらず、受信終了時にリセットされます。

0 : LPUART はアイドルです (受信なし)。

1 : 受信中です。

ビット 15:11 予約済みであり、リセット値のままにしておかなければなりません。

ビット 10 CTS : CTS フラグ

このビットは、ハードウェアによってセット/リセットされます。nCTS 入力ピンのステータスの反転コピーです。

0 : nCTS ラインはセットされました。

1 : nCTS ラインはリセットされました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 9 CTSIF : CTS 割り込みフラグ

このビットは、CTSE ビットがセットされていた場合、nCTS 入力がトグルしたときにハードウェアによってセットされます。LPUARTx_ICR レジスタの CTSCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

また、LPUARTx_CR3 レジスタで CTSIE=1 であれば、割り込みが生成されます。

0 : nCTS ステータスラインでの変更はありません。

1 : nCTS ステータスラインで変更がありました。

注： ハードウェアフロー制御機能がサポートされない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。

ビット 8 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 TXE : 送信データレジスタエンプティ

このビットは、LPUARTx_TDR レジスタの内容がシフトレジスタに転送されると、ハードウェアによってセットされます。LPUARTx_TDR レジスタへの書き込みによってクリアされます。

LPUARTx_CR1 レジスタの TXEIE ビット =1 の場合、割り込みが生成されます。

0 : データはシフトレジスタに転送されません。

1 : データはシフトレジスタに転送されます。

注： このビットは、シングルバッファ送信時に使用されます。

ビット 6 TC : 送信完了

データを含むフレームの送信が完了し、TXE がセットされている場合、このビットはハードウェアによってセットされます。LPUARTx_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。LPUARTx_ICR レジスタの TCCF に 1 を書き込むことによって、または LPUARTx_TDR レジスタに書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。

0 : 送信は完了していません。

1 : 送信は完了しています。

注： TE ビットがリセットされ、送信中でなかった場合、TC ビットはただちにセットされます。

ビット 5 RXNE : 読み出しデータレジスタノットエンプティ

このビットは、RDR シフトレジスタの内容が LPUARTx_RDR レジスタに転送されると、ハードウェアによってセットされます。LPUARTx_RDR レジスタの読み出しによってクリアされます。RXNE フラグは、LPUARTx_RQR レジスタの RXFRQ に 1 を書き込むことによってもクリアすることもできます。

LPUARTx_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。

0 : データは受信されていません。

1 : 受信データを読み出すことができます。

ビット 4 IDLE : アイドルライン検出

このビットは、アイドルラインが検出されたときに、ハードウェアによってセットされます。LPUARTx_CR1 レジスタの IDLEIE=1 である場合、割り込みが生成されます。LPUARTx_ICR レジスタの IDLECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

0 : アイドルラインは検出されていません。

1 : アイドルラインが検出されました。

注： RXNE ビットがセットされるまで (新しいアイドルラインが発生するまで)、IDLE ビットは再びセットされません。

ミュートモードが有効な場合 (MME=1)、LPUART がミュートでない場合 (RWU=0)、WAKE ビットによって選択されたミュートモードに関係なく、IDLE はセットされます。RWU=1 の場合、IDLE はセットされません。

ビット 3 ORE : オーバーランエラー

このビットは、RXNE=1 のときに、シフトレジスタで現在受信中のデータを RDR レジスタに転送する準備ができたときに、ハードウェアによってセットされます。LPUARTx_ICR レジスタの ORECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR1 レジスタの RXNEIE=1 または EIE=1 の場合、割り込みが生成されます。

0 : オーバーランエラーはありません。

1 : オーバーランエラーが検出されました。

注 : このビットがセットされると、RDR レジスタの内容は失われませんが、シフトレジスタは上書きされます。EIE ビットがセットされている場合、マルチバッファ通信中に ORE フラグがセットされた場合、割り込みが生成されます。

LPUARTx_CR3 レジスタの OVRDIS ビットがセットされると、このビットは永続的に 0 に強制設定されます (オーバーラン検出なし)。

ビット 2 NF : START ビットノイズ検出フラグ

このビットは、受信フレームの START ビットでノイズが検出されるとハードウェアによってセットされます。LPUARTx_ICR レジスタの NFCF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

0 : ノイズは検出されていません。

1 : ノイズが検出されました。

注 : このビットは、割り込みを生成する RXNE ビットと同時に出現するため、割り込みを生成しません。EIE ビットがセットされている場合、マルチバッファ通信中に NF フラグがセットされると、割り込みが生成されます。

ビット 1 FE : フレーミングエラー

このビットは、非同期化、過度なノイズ、またはブレークキャラクタが検出されたときに、ハードウェアによってセットされます。LPUARTx_ICR レジスタの FECF ビットに 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR1 レジスタの EIE=1 の場合、割り込みが生成されます。

0 : フレーミングエラーは検出されていません。

1 : フレーミングエラーまたはブレークキャラクタが検出されました。

ビット 0 PE : パリティエラー

このビットは、レシーバモードでパリティエラーが発生したときに、ハードウェアによってセットされます。LPUARTx_ICR レジスタの PECF に 1 を書き込むことによって、ソフトウェアによってクリアされます。

LPUARTx_CR1 レジスタの PEIE=1 の場合、割り込みが生成されます。

0 : パリティエラーはありません。

1 : パリティエラー

30.6.7 割り込みフラグクリアレジスタ (LPUARTx_ICR)

アドレスオフセット : 0x20

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-------|------|------|------|------|--------|-------|------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUCF | Res. | Res. | CMCF | Res. |
| | | | | | | | | | | | w | | | w | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | CTSCF | Res. | Res. | TCCF | Res. | IDLECF | ORECF | NCF | FECF | PECF |
| | | | | | | w | | | w | | w | w | w | w | w |

ビット 31:21 予約済みであり、リセット値のままにしておかなければなりません。

ビット 20 **WUCF** : STOP モードからのウェイクアップフラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの WUF フラグがクリアされます。

注 : *LPUART が STOP モードからのウェイクアップ機能をサポートしない場合、このビットは予約済みであり、ハードウェアによって強制的に 0 に設定されます。*

ビット 19:18 予約済みであり、リセット値のままにしておかなければなりません。

ビット 17 **CMCF** : キャラクター致フラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの CMF フラグがクリアされます。

ビット 16:10 予約済みであり、リセット値のままにしておかなければなりません。

ビット 9 **CTSCF** : CTS フラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの CTSIF フラグがクリアされます。

ビット 8:7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 7 予約済みであり、リセット値のままにしておかなければなりません。

ビット 6 **TCCF** : 送信完了フラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの TC フラグがクリアされます。

ビット 5 予約済みであり、リセット値のままにしておかなければなりません。

ビット 4 **IDLECF** : アイドルライン検出フラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの IDLE フラグがクリアされます。

ビット 3 **ORECF** : オーバーランエラーフラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの ORE フラグがクリアされます。

ビット 2 **NCF** : ノイズ検出フラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの NF フラグがクリアされます。

ビット 1 **FECF** : フレーミングエラーフラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの FE フラグがクリアされます。

ビット 0 **PECF** : パリティエラーフラグクリア

このビットに 1 を書き込むと、LPUARTx_ISR レジスタの PE フラグがクリアされます。

30.6.8 受信データレジスタ (LPUARTx_RDR)

アドレスオフセット : 0x24

リセット値 : 未定義

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

ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8:0 **RDR[8:0]** : 受信データ値

受信データキャラクタを含みます。

RDR レジスタは、入カシフトレジスタと内部バスとの間にパラレルインタフェースを提供します (図 230 を参照)。

パリティを有効にして受信する場合、MSB ビットで読み出される値が受信したパリティビットです。

30.6.9 送信データレジスタ (LPUARTx_TDR)

アドレスオフセット : 0x28

リセット値 : 未定義

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

ビット 31:9 予約済みであり、リセット値のままにしておかなければなりません。

ビット 8:0 **TDR[8:0]** : 送信データ値

送信されるデータキャラクタを含みます。

TDR レジスタは、内部バスと出カシフトレジスタとの間にパラレルインタフェースを提供します (図 230 を参照)。

パリティを有効にして (LPUARTx_CR1 レジスタの PCE ビットに 1 をセット) 送信しているとき、MSB (データ長に応じてビット 7 または 8) に書き込まれた値は、パリティによって置き換えられるため、無効です。

注 : このレジスタは、TXE=1 のときのみ書き込むことができます。

30.6.10 LPUART レジスタマップ

次の表に、LPUART のレジスタマップとリセット値を示します。

表 126. LPUART レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
|---------------|-----------------|----------|------|------|------|----------|------|-------|-------|-------|-------|-------|----------|-----------|----------|--------|-------|-------|------|------|-----------|------|------|-------|----------|------|-------|------|--------|--------|-------|-------|------|------|
| 0x00 | LPUARTx_ CR1 | Res. | Res. | Res. | M1 | Res. | Res. | DEAT4 | DEAT3 | DEAT2 | DEAT1 | DEAT0 | DEDT4 | DEDT3 | DEDT2 | DEDT1 | DEDT0 | Res. | CMIE | MME | M | | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | UESM | UE |
| | リセット値 | | | | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | LPUARTx_ CR2 | ADD[7:4] | | | | ADD[3:0] | | | | | Res. | Res. | Res. | | MSBFIRST | DATINV | TXINV | RXINV | SWAP | Res. | STOP[1:0] | | Res. | Res. | Res. | Res. | Res. | Res. | Res. | ADDM7 | Res. | Res. | Res. | Res. |
| | リセット値 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | | Res. | Res. | Res. | Res. | | | 0 | | Res. | Res. | Res. |
| 0x08 | LPUARTx_ CR3 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | UCESM | WUFIE | WUS[1:0] | Res. | Res. | Res. | Res. | DEP | DEM | DDRE | OVRDIS | | Res. | CTSIE | CTSE | RTSE | DMAT | DMAR | Res. | Res. | HDSEL | Res. | Res. | EIE |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | | | 0 | | | 0 |
| 0x0C | LPUARTx_ BRR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BRR[19:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x10- 0x14 | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x18 | LPUARTx_ RQR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXFRQ | RXFRQ | MMRQ | SBKRQ | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | |
| 0x1C | LPUARTx_ISR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | REACK | TEACK | WUF | RWU | SBKF | CMF | BUSY | Res. | Res. | Res. | Res. | Res. | CTS | CTSIF | Res. | TXE | TC | RXNE | IDLE | ORE | NF | FE | PE | |
| | リセット値 | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | 0 | 0 | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x20 | LPUARTx_ICR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | WUCF | Res. | Res. | CMCF | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTSCF | Res. | Res. | TC | TC | ORECF | NCF | FE | PECF | |
| | リセット値 | | | | | | | | | | | | 0 | | | 0 | | | | | | | | 0 | | | 0 | | | 0 | 0 | 0 | 0 | |
| 0x24 | LPUARTx_ RDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RDR[8:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x28 | LPUARTx_ TDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TDR[8:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。

31 シリアルペリフェラルインタフェース/I²S (SPI/I2S)

31.1 概要

SPI/I²S インタフェースを使用して、SPI プロトコルまたは I²S オーディオプロトコルに基づき外部デバイスと通信することができます。SPI または I²S モードはソフトウェアによって選択可能です。デバイスのリセット後は、デフォルトで SPI モードが選択されます。

SPI (シリアルペリフェラルインタフェース) プロトコルは、外部デバイスとの半二重、全二重、および単方向の同期シリアル通信をサポートしています。このインタフェースはマスタとして設定することも可能で、その場合、外部スレーブデバイスに通信クロック (SCK) を供給します。このインタフェースは、マルチマスタ設定で動作することもできます。

I²S プロトコルも、同期シリアル通信インタフェースです。スレーブまたはマスタモードで、レシーバまたはトランスミッタとして動作することができます。

フィリップス I²S 規格、MSB/LSB 詰め規格、PCM 規格など、4 つのオーディオ規格に対応できます。

31.1.1 SPI の主な機能

- マスタまたはスレーブ動作
- 3 本のラインでの全二重同期転送
- 2 本のラインでの半二重同期転送 (双方向データライン有り)
- 2 本のラインでの単方向同期転送 (単方向データライン有り)
- 8 または 16 ビットの転送フレームフォーマット選択
- マルチマスタモード機能
- 8 個のマスタモードポーレートプリスケアラ (最大周波数 $f_{PCLK}/2$)
- スレーブモード周波数 (最大周波数 $f_{PCLK}/2$)
- マスタとスレーブの両方に対するハードウェア/ソフトウェアによる NSS 管理: マスタ/スレーブ動作の動的切り替え
- クロックの極性と位相をプログラム可能
- データ順序をプログラム可能 (MSB ファースト/LSB ファーストのシフト)
- 専用の送受信フラグ (割り込み機能付き)
- SPI バスビジステータスフラグ
- SPI モトローラモードをサポート
- ハードウェア CRC 機能による信頼性の高い通信:
 - Tx モードでは CRC 値を最終バイトとして送信可能
 - 最終受信バイトに対する CRC エラーの自動チェック
- マスタモードの障害、オーバーランの各フラグ (割り込み機能付き)
- CRC エラーフラグ
- 1 バイト/ワードの送受信バッファ (DMA 機能付き: 送受信リクエスト)

31.1.2 SPI の拡張機能

- SPI TI モードをサポート

31.1.3 I²Sの機能

- 半二重通信（トランスミッタまたはレシーバのみ）
- マスタまたはスレーブ動作
- 正確なオーディオサンプリング周波数（8~192 kHz）を実現するプログラム可能な 8 ビットのリニアプリスケアラ
- 16、24、または 32 ビットのデータフォーマット
- パケットフレームはオーディオチャネルによって 16 ビット（16 ビットデータフレーム）または 32 ビット（16、24、32 ビットデータフレーム）に固定。
- プログラム可能なクロック極性（定常状態）
- スレーブ送信モードのアンダーランフラグ、受信モード（マスタおよびスレーブ）のオーバーランフラグ、受信モードと送信モード（スレーブの場合のみ）のフレームエラーフラグ
- 送受信用の 16 ビットレジスタ（両チャネルサイドに対して 1 個のデータレジスタ）
- 以下の I²S プロトコルをサポート。
 - フィリップス I²S 規格
 - MSB 詰め規格（左詰め）
 - LSB 詰め規格（右詰め）
 - PCM 規格（16 ビットチャネルフレーム、または 32 ビットチャネルフレームに拡張された 16 ビットデータフレームでの、ショートおよびロングフレーム同期付き）
- データ方向は常に MSB ファースト。
- 送受信（16 ビット幅）用の DMA 機能
- 外部オーディオコンポーネントを駆動するためのマスタクロックを出力可能。周波数比は、 $256 \times F$ （ F_S はオーディオサンプリング周波数）に固定。
- I²S（I2S2）クロックは、I2S_CKIN ピンに配置された外部クロックから取得可能。

31.2 SPI/I2S の実装

このマニュアルでは、SPI1 および SPI2 に実装されているすべての機能について説明しています。

表 127. STM32L0x3 SPI の実装

| SPI の機能 ⁽¹⁾ | SPI1 | SPI2 |
|------------------------|------|------|
| ハードウェア CRC 計算 | X | X |
| I ² S モード | - | X |
| TI モード | X | X |

1. X : サポートされています。

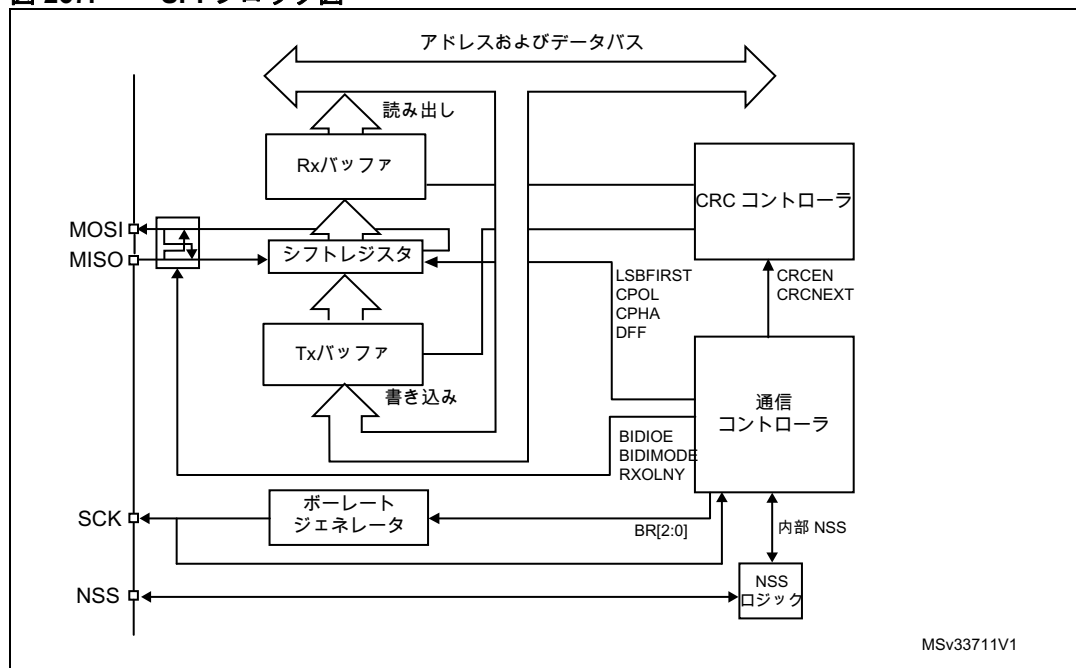
この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

31.3 SPI の機能説明

31.3.1 概要

SPI では、MCU と外部デバイス間の同期シリアル通信が可能です。アプリケーションソフトウェアは、ステータスフラグをポーリングするか、または専用の SPI 割り込みを使用することで、通信を管理することができます。SPI の主要要素およびそれらの相互作用を以下のブロック図 (図 267 に示します。

図 267. SPI ブロック図



4 本の I/O ピンが外部デバイスとの SPI 通信専用に使われます。

- **MISO** : マスターイン/スレーブアウトデータ。一般に、このピンは、スレーブモードではデータの送信に、マスタモードではデータの受信に使われます。
- **MOSI** : マスターアウト/スレーブインデータ。一般に、このピンは、マスタモードではデータの送信に、スレーブモードではデータの受信に使われます。
- **SCK (シリアルクロック)** : SPI マスタではシリアルクロックの出力に、SPI スレーブでは入力に使われます。
- **NSS** : スレーブ選択用のピンです。このピンは、SPI および NSS の設定に応じて、以下のいずれかに使われます。
 - 個々の通信用スレーブデバイスを選択する
 - データフレームを同期させる
 - 複数のマスタ間での競合を検出する

詳細は、[セクション 31.3.4 : スレーブ選択 \(NSS\) ピンの管理](#) を参照してください。

SPI バスを使用することで、1 つのマスタデバイスと 1 つ以上のスレーブデバイスとの間で通信することができます。バスは 2 本以上の線から成り、1 本はクロック信号用、その他はデータの同期転送用です。SPI ノード間でのデータ交換とそれらのスレーブ選択信号管理に応じて、その他の信号を追加することができます。

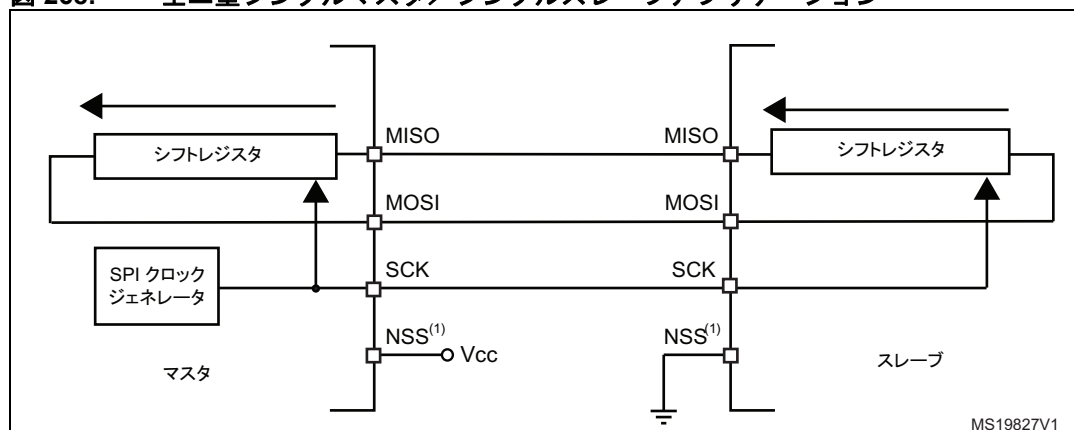
31.3.2 マスタとスレーブの 1 対 1 の通信

SPI を使用することで、MCU は対象となるデバイスやアプリケーション要件に応じたさまざまな設定で通信ができます。これらの設定には、2 または 3 本の線（ソフトウェア NSS 管理あり）、あるいは 3 または 4 本の線（ハードウェア NSS 管理あり）が使われます。通信は常にマスタによって開始されます。

全二重通信

SPI は、デフォルトで全二重通信に設定されます。この設定では、マスタおよびスレーブのシフトレジスタは、MOSI ピンと MISO ピンの間に 2 本の単方向ラインを介してリンクされます。SPI 通信の間、データはマスタから供給される SCK クロックのエッジに同期してシフトされます。マスタは、送信すべきデータを MOSI ライン経由でスレーブに送信し、MISO ライン経由でスレーブからデータを受信します。データフレーム転送が完了した（すべてのビットがシフトされた）時点で、マスタとスレーブの間で情報が交換されます。

図 268. 全二重シングルマスタ/シングルスレーブアプリケーション

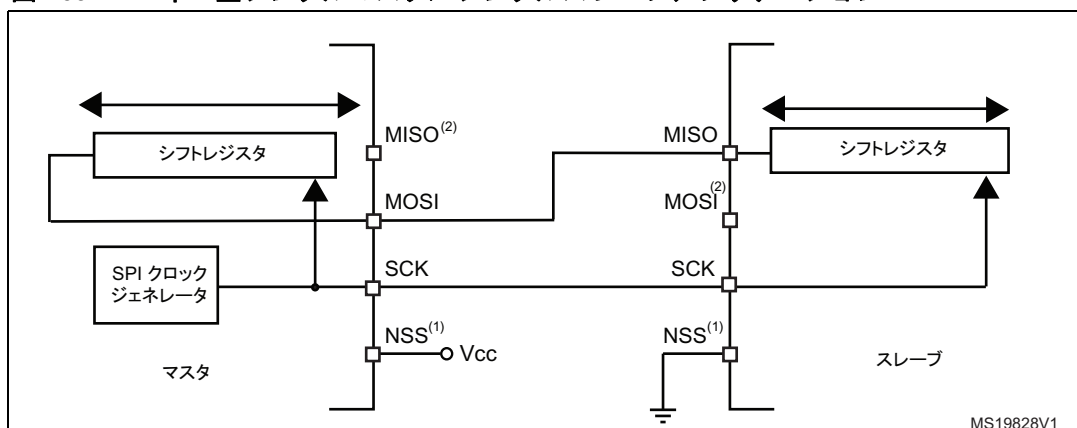


1. この場合、NSS ピンは入力として設定されます。

半二重通信

SPi_x_CR1 レジスタの BIDIMODE ビットをセットすることで、SPI は半二重モードで通信できます。この設定では、1 本の交差接続ラインを使用して、マスタとスレーブのシフトレジスタを互いにリンクさせます。この通信中に、データは SCK クロックのエッジに同期して、シフトレジスタ間でシフトされます。シフトの方向は、マスタとスレーブの両方が SPi_x_CR1 レジスタの BDIOE ビットを使用して相互に選択された転送方向となります。この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、他のアプリケーションで使用でき、GPIO として機能します。

図 269. 半二重シングルマスタ/シングルスレーブアプリケーション



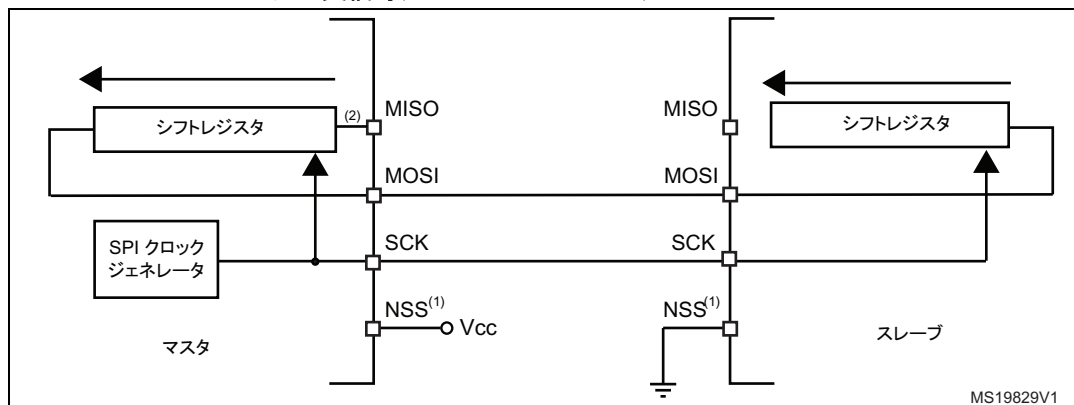
1. この場合、NSS ピンは入力として設定されます。
2. この設定では、マスタの MISO ピンとスレーブの MOSI ピンは、GPIO として使用できます。

単方向通信

SPI は、SPIx_CR2 レジスタの RXONLY ビットを使用して送信専用または受信専用に設定することにより、単方向モードで通信できます。この設定では、マスタとスレーブのシフトレジスタ間の転送に使用するのは 1 ラインのみです。残りの MISO ピンと MOSI ピンのペアは通信には使用されず、標準の GPIO として使用できます。

- **送信専用モード (RXONLY = 0) の場合**：設定は全二重の場合と同じです。アプリケーションは、未使用の入力ピンでキャプチャされた情報を無視する必要があります。このピンは標準の GPIO として使用できます。
- **受信専用モード (RXONLY = 1) の場合**：アプリケーションにて、RXONLY ビットをセットすることによって、SPI 出力機能を無効にできます。スレーブ設定では、MISO 出力が無効化され、ピンを GPIO として使用することができます。スレーブ選択信号がアクティブな間は、スレーブは MOSI ピンからデータを受信し続けます（[31.3.4 : スレーブ選択 \(NSS\) ピンの管理](#)を参照）。データバッファの設定に応じて、受信データイベントが出現します。マスタ設定では、MOSI 出力が無効化され、ピンを GPIO として使用することができます。SPI が有効である間はクロック信号が生成され続けます。クロックを停止させる唯一の方法は、クロックの設定に応じて、RXONLY ビットまたは SPE ビットをクリアし、MISO ピンからの受信パターンが終了し、データバッファ構造への書き込みが行われるまで待つことです。

図 270. 単方向シングルマスタ/シングルスレーブアプリケーション (送信専用モードのマスタ/受信専用モードのスレーブ)



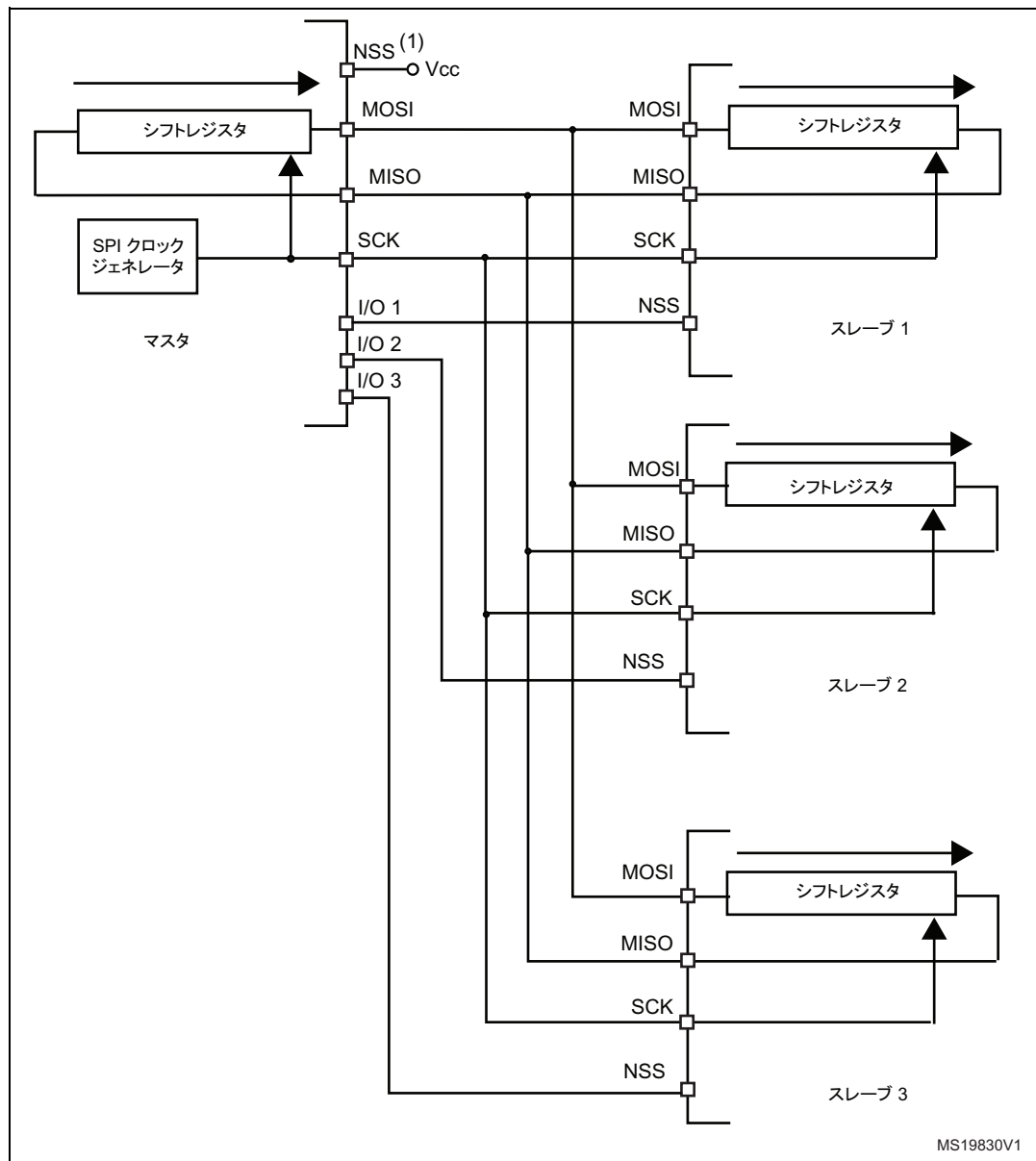
1. この場合、NSS ピンは入力として設定されます。
2. 入力情報はシフトレジスタでキャプチャされますが、標準の送信専用モード (たとえば、OVF フラグ) では無視する必要があります。
3. この設定では、両方のMISO ピンを GPIO として使用できます。

注： すべての単方向通信は、トランザクション方向の設定を固定して (双方向モードはBDIO ビットが変化しない限り有効)、別の半二重通信に置き換えることができます。

31.3.3 標準マルチスレーブ通信

2 つ以上の独立したスレーブがある設定の場合、マスタは GPIO ピンを使用して、各スレーブのチップセレクトラインを管理します (図 271 を参照)。マスタは、スレーブの NSS 入力に接続されている GPIO をローレベルにプルダウンすることによって、スレーブの 1 つを選択する必要があります。これを行うことにより、標準マスタと専用スレーブの通信が確立します。

図 271. マスタと 3 つの独立したスレーブ



1. スレーブの MISO ピンは相互接続されているので、すべてのスレーブにおいて、その MISO ピンの GPIO 設定をオルタネート機能オープンドレインとしてセットする必要があります (セクション 9.3.7: I/O オルタネート機能の入力/出力 (210 ページ) を参照)。

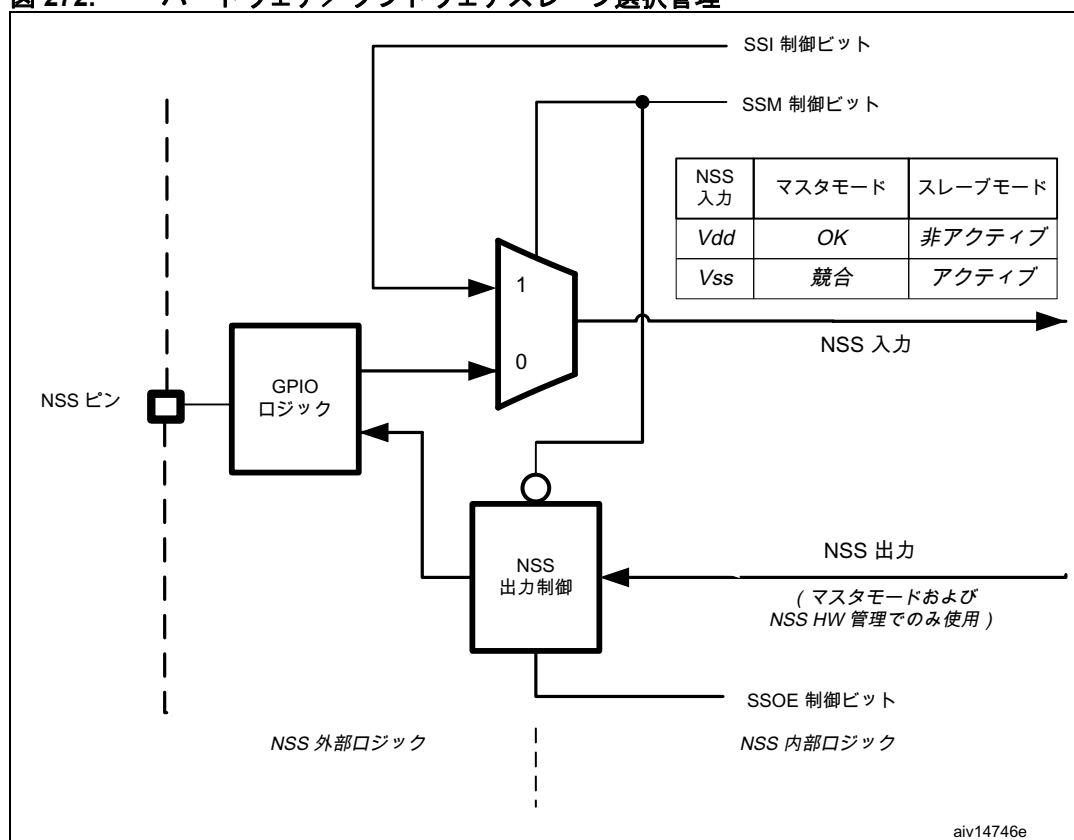
31.3.4 スレーブ選択 (NSS) ピンの管理

スレーブモードでは、NSS は標準の「チップセレクト」入力として機能し、スレーブをマスタと通信させます。マスタモードでは、NSS は出力としても入力としても使用できます。入力として使用する場合は、NSS はマルチマスタのバスの衝突を未然に防ぎ、出力として使用する場合は 1 つのスレーブのスレーブ選択信号を駆動させることができます。

ハードウェアまたはソフトウェアのスレーブ選択管理は、SPIx_CR1 レジスタの SSM ビットを使用して、以下のようにセットすることができます。

- **ソフトウェア NSS 管理 (SSM = 1)** : この設定では、スレーブ選択情報は SPIx_CR1 レジスタの SSI ビットの値によって内部で駆動されます。外部 NSS ピンは他のアプリケーションで使用できます。
- **ハードウェア NSS 管理 (SSM = 0)** : この場合、2 通りの設定が可能です。次のどちらの設定を使用するかは、NSS 出力設定 (SPIx_CR1 レジスタの SSOE ビット) によって決まります。
 - **NSS 出力が有効な場合 (SSM = 0, SSOE = 1)** : この設定は、MCU がマスタとしてセットされている場合にのみ使用します。NSS ピンはハードウェアによって管理されます。NSS 信号は、SPI がマスタモードで有効になる (SPE = 1) とすぐにローレベルに駆動され、SPI が無効化される (SPE = 0) までローレベルに保たれます。
 - **NSS 出力が無効な場合 (SSM = 0, SSOE = 0)** : マイクロコントローラがバスでマスタとして機能している場合、この設定によりマルチマスタ機能が可能になります。このモードで、NSS ピンがローレベルにプルダウンされた場合、SPI はマスタモードのフォールト状態に入り、デバイスは自動的にスレーブモードに再設定されます。スレーブモードでは、NSS ピンは標準の「チップセレクト」入力として機能し、NSS ラインがローレベルの間はスレーブが選択されます。

図 272. ハードウェア/ソフトウェアスレーブ選択管理



31.3.5 通信フォーマット

SPI 通信中は受信と送信の操作が同時に行われます。シリアルクロック (SCK) は、データライン上で行われる情報のシフトとサンプリングを同期させます。通信フォーマットは、クロック位相、クロック極性、およびデータフレームフォーマットに応じて決定されます。マスタデバイスとスレーブデバイスの通信を可能にするには、双方が同じ通信フォーマットに従う必要があります。

クロックの位相および極性の制御

SPIx_CR1 レジスタの CPOL ビットと CPHA ビットを使用することによって、考えられる4つのタイミングの関係をソフトウェアで選択できます。CPOL (クロック極性) ビットは、データが転送されていないときのクロックのアイドル状態の値を制御します。このビットは、マスタモードとスレーブモードの両方に影響を与えます。CPOL がリセットされると、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされると、SCK ピンはハイレベルのアイドル状態になります。

CPHA ビットがセットされると、SCK ピンの2番目のエッジがトランザクションの最初のデータビットをキャプチャします (CPOL ビットがリセットされていれば立ち下がりエッジ、CPOL ビットがセットされていれば立ち上がりエッジ)。データは、この種のクロック遷移が発生するたびにラッチされます。CPHA ビットがリセットされている場合、SCK ピンの1番目のエッジがトランザクションの最初のデータビットをキャプチャします (CPOL ビットがセットされていれば立ち下がりエッジ、CPOL ビットがリセットされていれば立ち上がりエッジ)。データは、この種のクロック遷移が発生するたびにラッチされます。

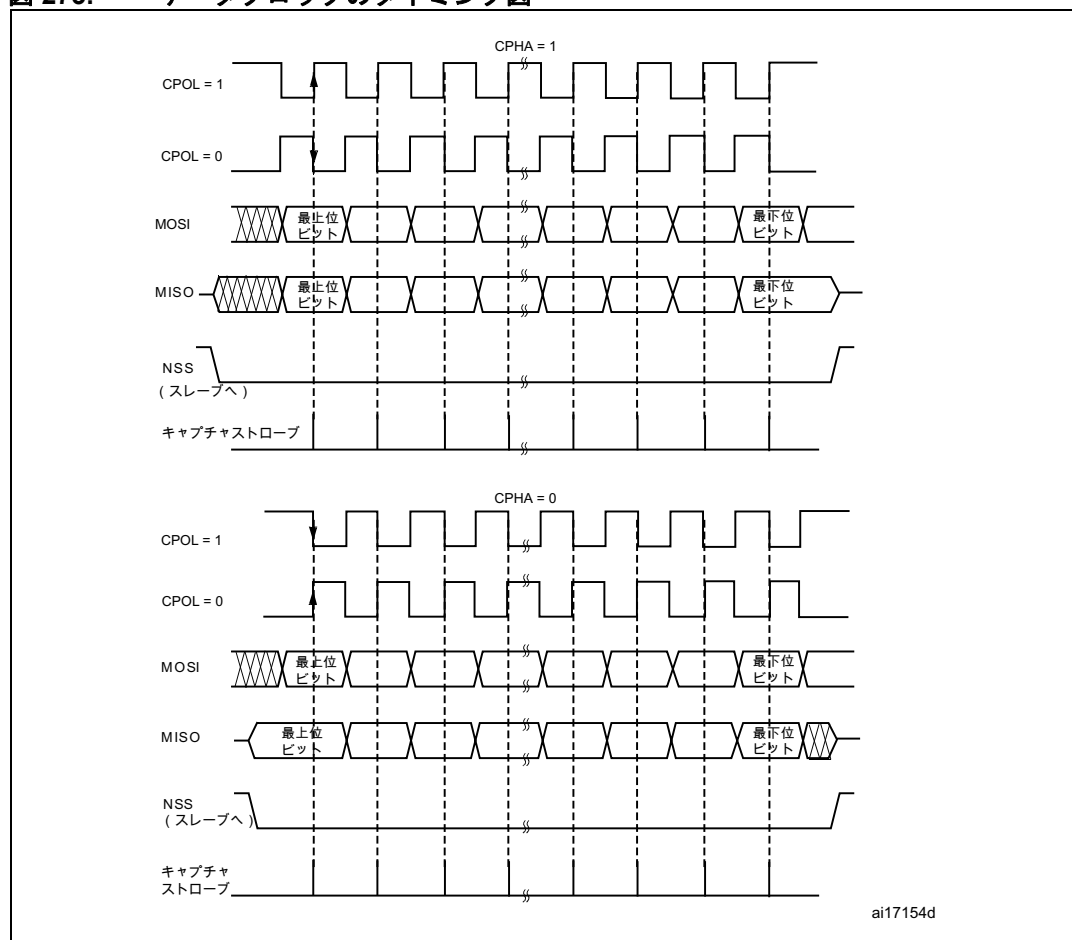
CPOL (クロック極性) ビットと CPHA (クロック位相) ビットの組み合わせによって、データキャプチャのクロックエッジを選択できます。

 273 は、CPHA ビットと CPOL ビットの4つの組み合わせによる SPI 全二重転送を示しています。

注： *CPOL または CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。*

SCK のアイドル状態は、SPIx_CR1 レジスタで (CPOL = 1 なら SCK のプルアップ、CPOL = 0 なら SCK のプルダウンによって) 選択された極性に一致する必要があります。

図 273. データクロックのタイミング図



注： データビットの順序はLSBFIRST ビットの設定値に依存します。

データフレームフォーマット

LSBFIRST ビットの値に応じて、SPI シフトレジスタを設定することで、シフトをMSB ファーストまたはLSB ファーストに設定することができます。各データフレームの長さは、SPI_CR1 レジスタのDFF ビットを使用してデータのサイズをプログラムすることにより、8 または 16 ビットに設定できます。選択されたデータフレームフォーマットは、送受信のどちらにも適用できます。

31.3.6 SPI の設定

設定手順は、マスタとスレーブではほぼ同じです。特定のモードの設定については、それぞれのモードに関する章を参照してください。標準通信を初期化する必要があるときは、以下の手順を実行します。

- 適切な GPIO レジスタに書き込みを行います。MOSIピン、MISOピン、SCKピンの GPIO 設定を行います。
- SPI_CR1 レジスタに書き込みを行います。
 - BR[2:0] ビットを使用して、シリアルクロックボーレートを設定します（[注：3](#) を参照）。
 - CPOL ビットと CPHA ビットの組み合わせを設定して、データ転送とシリアルクロックの 4 つの関係のうちの 1 つを定義します（[注：2](#) を参照）。
 - RXONLY または BIDIMODE、および BIDIOE を設定することによって（RXONLY と BIDIMODE は同時にセットできません）、単方向または半二重モードを選択します。
 - LSBFIRST ビットを設定して、フレームフォーマットを定義します（[注：2](#) を参照）。
 - CRC が必要な場合は（SCK クロック信号がアイドル状態のとき）、CRCEN ビットを設定します。
 - SSM と SSI を設定します（[注：2](#) を参照）。
 - MSTR ビットを設定します（マルチマスタ NSS 設定では、MODF エラーを防ぐためにマスタが設定されている場合、NSS での競合を避けること）。
 - DFF ビットをセットして、データフレームフォーマット（8 または 16 ビット）を設定します。
- 以下のように、SPI_CR2 レジスタに書き込みを行います。
 - SSOE を設定します（[注：1](#) および [2](#) を参照）。
 - TI プロトコルが必要な場合は、FRF ビットをセットします。
- SPI_CRCPR レジスタに書き込みを行います。必要に応じて CRC 多項式を設定します。
- 適切な DMA レジスタに書き込みを行います。DMA ストリームが使用されている場合は、DMA レジスタに SPI Tx および Rx 専用の DMA ストリームを設定します。

注：

- (1) このステップはスレーブモードでは必要ありません。
- (2) このステップは TI モードでは必要ありません。
- (3) このステップは、スレーブモードにて TI モードで動作している場合を除き、スレーブモードでは必要ありません。

31.3.7 SPI を有効にする手順

マスタがクロックを送信する前に、SPI スレーブを有効にすることを推奨します。そうしないと、望ましくないデータ送信が発生することがあります。スレーブデータレジスタは、マスタとの通信を開始する前に、送信データをすでに格納していなければなりません（通信クロックの 1 番目のエッジに、またはクロック信号が連続的なときは現在の通信の最後の前に）。SPI スレーブが有効になる前に、SCK 信号を選択された極性に対応するアイドル状態のレベルに安定させる必要があります。

全二重（または送信専用モード）では、SPI が有効になり、送信するデータが Tx バッファに書き込まれた時点で、マスタは通信を開始します。

あらゆるマスタ受信専用モードにおいて（RXONLY = 1 または BIDIMODE = 1、および BIDIOE = 0）、SPI が有効になるとすぐに、マスタは通信を開始し、クロックは動作を開始します。

スレーブは、マスタから正しいクロック信号を受信した時点で通信を開始します。SPI マスタが転送を開始する前に、スレーブのソフトウェアは送信データを書き込んでおく必要があります。

DMA の処理方法については、[セクション 31.3.10：DMA（ダイレクトメモリアクセス）を使用する通信](#)を参照してください。



31.3.8 データの送受信手順

RxバッファとTxバッファ

受信の場合、データは受信されたあと、内部のRxバッファに格納されます。一方、送信の場合、データはまず内部のTxバッファに格納されてから、送信されます。SPI_DR レジスタへの読み出しアクセスでは、Rxバッファに格納された値が返されます。一方、SPI_DR への書き込みアクセスでは、書き込まれたデータがTxバッファに格納されます。

Txバッファの処理

データフレームは、最初のビットの送信中にTxバッファからシフトレジスタへロードされます。その後、ビットは、LSBFIRST ビットの設定に応じて、シフトレジスタから専用の出力ピンへ順次シフトアウトされます。データがTxバッファからシフトレジスタへ転送された時点で、TXE フラグ (Txバッファエンプティ) がセットされます。このフラグは、内部のTxバッファに次のデータをロードする準備ができていることを示します。SPI_CR2 レジスタの TXEIE ビットがセットされている場合は、割り込みを生成できます。TXE ビットは、SPI_DR レジスタへの書き込みによってクリアされます。

前のフレーム送信がまだ進行中の間は、次に送信されるデータがTxバッファに格納されていれば、連続した送信ストリームを実現できます。TXE フラグがセットされていないときにソフトウェアがTxバッファへの書き込みを行った場合、トランザクション待ちのデータが上書きされます。

Rxバッファの処理

データがシフトレジスタからRxバッファに転送されると、最後のサンプリングクロックエッジでRXNE フラグ (Rxバッファノートエンプティ) がセットされます。このフラグは、SPI_DR レジスタからデータを読み出す準備ができていることを示します。SPI_CR2 レジスタの RXNEIE ビットがセットされている場合は、割り込みを生成できます。RXNE ビットは、SPI_DR レジスタの読み出しによってクリアされます。

前に送信されたデータバイトが原因でデバイスが RXNE ビットをクリアしなかった場合、次の値がバッファされた時点でオーバーランの条件が発生します。OVR ビットがセットされ、ERRIE ビットがセットされている場合は割り込みが生成されます。

データ交換を管理するもう一つの方法は、DMA を使用することです ([セクション 11.2 : DMA の主な機能](#)を参照)。

シーケンス処理

現在のデータフレームトランザクションが進行中の場合に、BSY ビットがセットされます。クロック信号が流れ続けているときは、BSY フラグはマスタ側のデータフレーム間でセットされたままになります。一方、スレーブ側では、BSY フラグは、各データフレーム転送間で最小の SPI クロック1サイクルの間ローレベルになります。

設定によっては、最後のデータ転送時に BSY フラグを使用して、転送の完了まで待つことができます。

マスタ側で、半二重 (BIDIMODE = 1、BIDIOE = 0) または単方向 (BIDIMODE = 0、RXONLY = 1) のいずれかの設定で、受信専用モードが設定されている場合、SPI が有効になるとすぐにマスタは受信シーケンスを開始します。次に、マスタによってクロック信号が供給されますが、この信号はマスタが SPI または受信専用モードを無効にするまで停止しません。マスタは、クロック信号が停止するまでデータフレームを受信し続けます。

マスタはあらゆるトランザクションを連続モードで供給できる (SCK 信号は連続的) 一方で、データフローおよびその内容をいつでも処理できるスレーブ機能を優先する必要があります。必要に応じて、マスタは通信速度を下げ、より低速のクロックか、または十分な遅延を含む個別のフレーム/データセッションを供給する必要があります。次の 2 点に注意してください。SPI モードで動作する

スレーブに対してはアンダーフローエラー信号はありません。また、スレーブからのデータは、たとえスレーブがそれらのデータを時間内に適切に準備できない場合でも、常にマスタによってトランザクション処理されます。スレーブが DMA を使用することが特に望ましいのは、データフレームが短く、バスが高速の場合です。

マルチスレーブシステムにおいて通信用のスレーブを 1 つだけ選択するには、各シーケンスをNSS パルス内に収める必要があります。シングルスレーブシステムでは、スレーブの制御に NSS を使用する必要はありません。ただし、NSS パルスを使用して、スレーブを各データ転送シーケンスの開始と同期させることができます。NSS は、ソフトウェアまたはハードウェアによって管理できます（[セクション 31.3.4：スレーブ選択 \(NSS\) ピンの管理](#)を参照）。

マスタ/全二重モードおよびスレーブ/全二重モードでの連続転送については、[図 274](#) および [図 275](#) を参照してください。

図 274. マスタ/全二重モードでの TXE/RXNE/BSY 動作 (BIDIMODE = 0、RXONLY = 0) 連続転送の場合

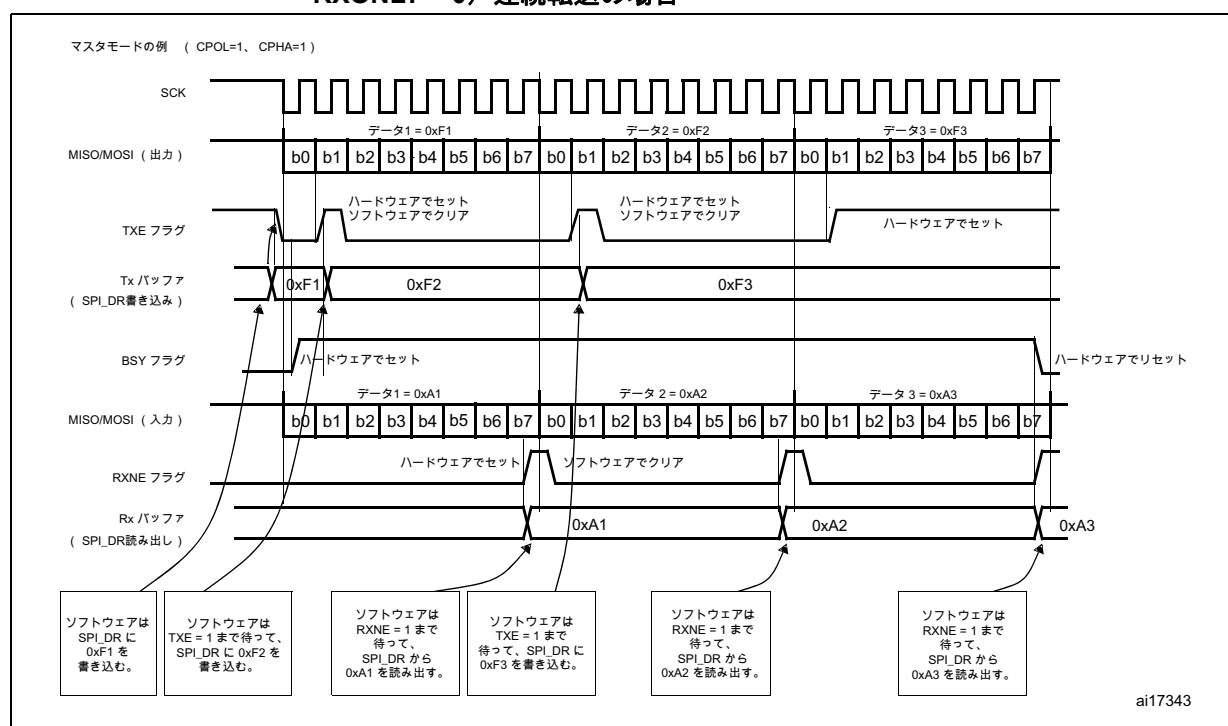
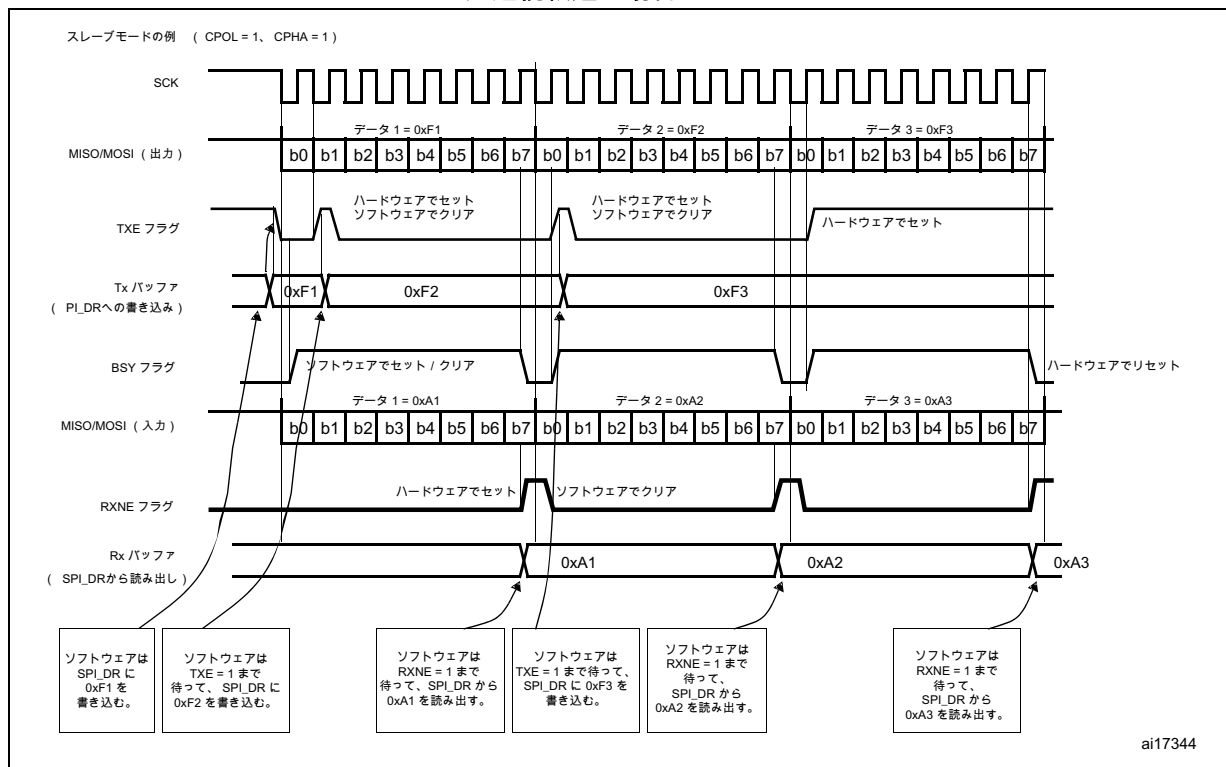


図 275. スレーブ/全二重モードでの TXE/RXNE/BSY 動作 (BIDIMODE = 0、RXONLY = 0) 連続転送の場合



31.3.9 SPI を無効にする手順

SPI を無効にする場合は、本項に記載されている無効化手順に従ってください。この手順は、ペリフェラルクロックが停止し、システムが低電力モードに入る前に行うことが重要です。この場合、進行中のトランザクションが破壊されることがあります。モードによっては、この無効化手順が連続通信を停止させる唯一の方法です。

全二重または送信専用モードでは、マスタは、転送するデータの供給を停止した時点でいかなるトランザクションも終了することができます。この場合、クロックは最後のデータトランザクション後に停止します。

標準的な無効化手順は、送信セッションが完全に終わったかどうかをチェックするために、TXE フラグおよび BSY フラグの状態をポーリングして行われます。このチェックは、たとえば以下に示すように、進行中のトランザクションの終わりを識別する必要があるような特別な場合にも行うことができます。

- NSS 信号が任意の GPIO トグルによって管理されており、マスタはスレーブに適切な NSS パルスの終わりを提供する場合
- 最後のデータフレームまたは CRC フレームのトランザクションがまだペリフェラルバスで進行している間に、DMA からのトランザクションのストリームが完了した場合

正しい無効化手順を以下に示します (受信専用モードが使用されている場合を除く)。

1. RXNE = 1 になるまで待ってから、最後のデータを受信します。
2. TXE = 1 になるまで待ち、さらに BSY = 0 になるまで待ってから、SPI を無効にします。
3. 受信データを読み出します。

注： 不連続通信時には、SPI_DR レジスタへの書き込み動作と BSY ビットが設定される間に、2 APB クロック周期分の遅延があります。従って、最後のデータを書き込んだ後、まず TXE がセットされるまで待ち、さらに BSY がクリアされるまで待つ必要があります。

受信専用モードの場合の正しい無効化手順を以下に示します。

1. 最後のデータフレームの進行中に特定の時間枠内で SPI を無効 (SPE = 0) にすることにより、受信フローへの割り込みを行います。
2. BSY = 0 (最後のデータフレームが処理される) まで待ちます。
3. 受信データを読み出します。

注： 連続受信シーケンスを停止させるには、最後のデータフレームの受信中に、ある特定の時間枠を順守する必要があります。その時間枠は、最初のビットがサンプリングされたときに開始され、最後のビット転送が開始される前に終了します。

31.3.10 DMA (ダイレクトメモリアクセス) を使用する通信

最高速度で動作し、オーバーランを回避するために必要なデータレジスタの読み出し/書き込み処理を容易にするために、SPI は簡単なリクエスト/確認応答プロトコルを実現する DMA 機能を備えています。

SPIx_CR2 レジスタの TXE または RXNE イネーブルビットをセットすると、DMA アクセスがリクエストされます。Tx バッファと Rx バッファには、別々のリクエストを発行する必要があります。

- 送信では、TXE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタに書き込みます。
- 受信では、RXNE が 1 にセットされるたびに DMA リクエストが発行されます。その後、DMA は SPIx_DR レジスタを読み出します。

DMA 送受信波形については、[図 276](#) および [図 277](#) を参照してください。

SPI がデータの送信にのみ使用される場合、SPI Tx DMA チャンネルのみを有効にすることができます。この場合、受信したデータは読み出されないため、OVR フラグがセットされます。SPI がデータの受信にのみ使用される場合、SPI Rx DMA チャンネルのみを有効にすることができます。

送信モードで、DMA がすべての送信データを書き込んだとき (DMA_ISR レジスタの TCIF フラグがセットされます)、BSY フラグを監視することで SPI 通信の完了を確認できます。最後の送信内容の破壊を避けるために、SPI を無効にする前、または STOP モードに入る前にこの操作を行う必要があります。ソフトウェアは、まず TXE = 1 になるまで待ち、さらに BSY = 0 になるまで待つ必要があります。

DMA を使用して通信を開始する場合、DMA チャンネルの管理によるエラーイベントを防ぐために、以下の手順を順に実行する必要があります。

1. DMA Rx が使用される場合、SPI_CR2 レジスタの RXDMAEN ビットの DMA Rx バッファを有効にします。
2. ストリームが使用される場合、DMA レジスタにて Tx および Rx の DMA ストリームを有効にします。
3. DMA Tx が使用される場合、SPI_CR2 レジスタの TXDMAEN ビットにて DMA Tx バッファを有効にします。
4. SPE ビットをセットして、SPI を有効にします。

通信を終了するには、以下の手順を順に実行する必要があります。

1. ストリームが使用される場合、DMA レジスタにて Tx および Rx の DMA ストリームを無効にします。
2. SPI 無効化手順に従って SPI を無効にします。

3. DMA Tx および(または) DMA Rx が使用されている場合、SPI_CR2 レジスタの TXDMAEN ビットおよび RXDMAEN ビットをクリアすることにより、DMA Tx バッファおよび Rx バッファを無効にします。

図 276. DMA を使用した送信

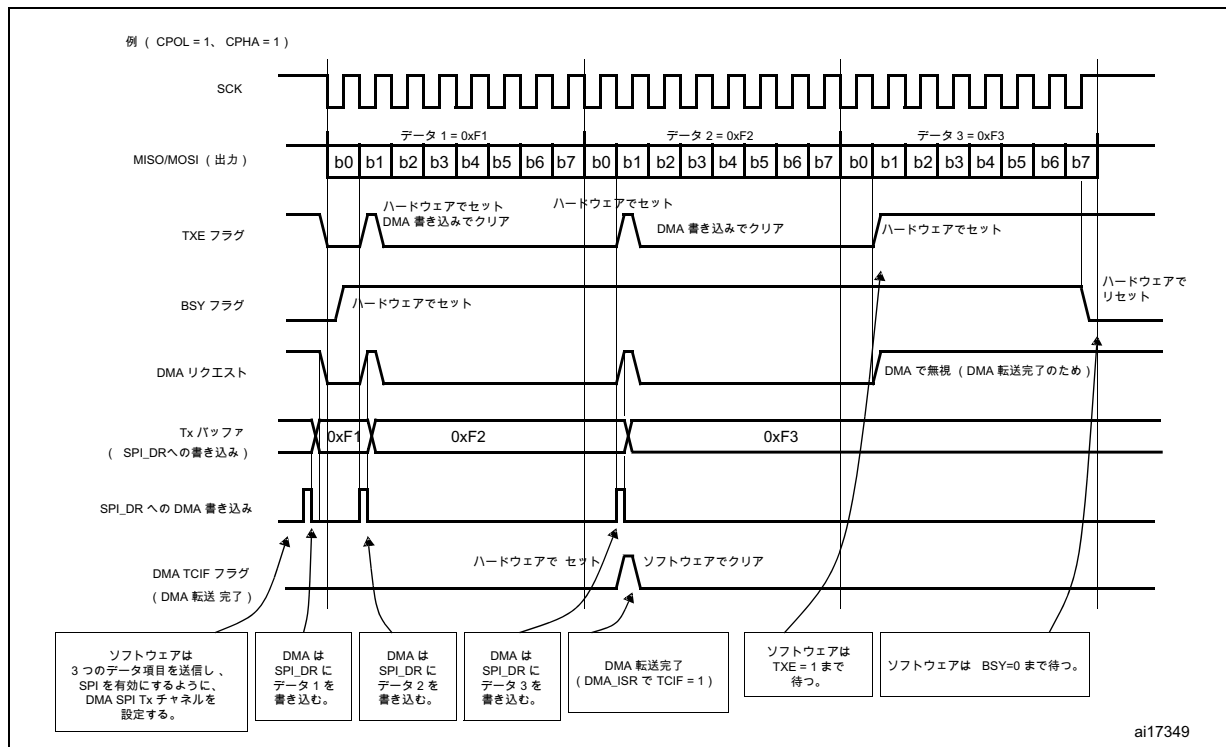
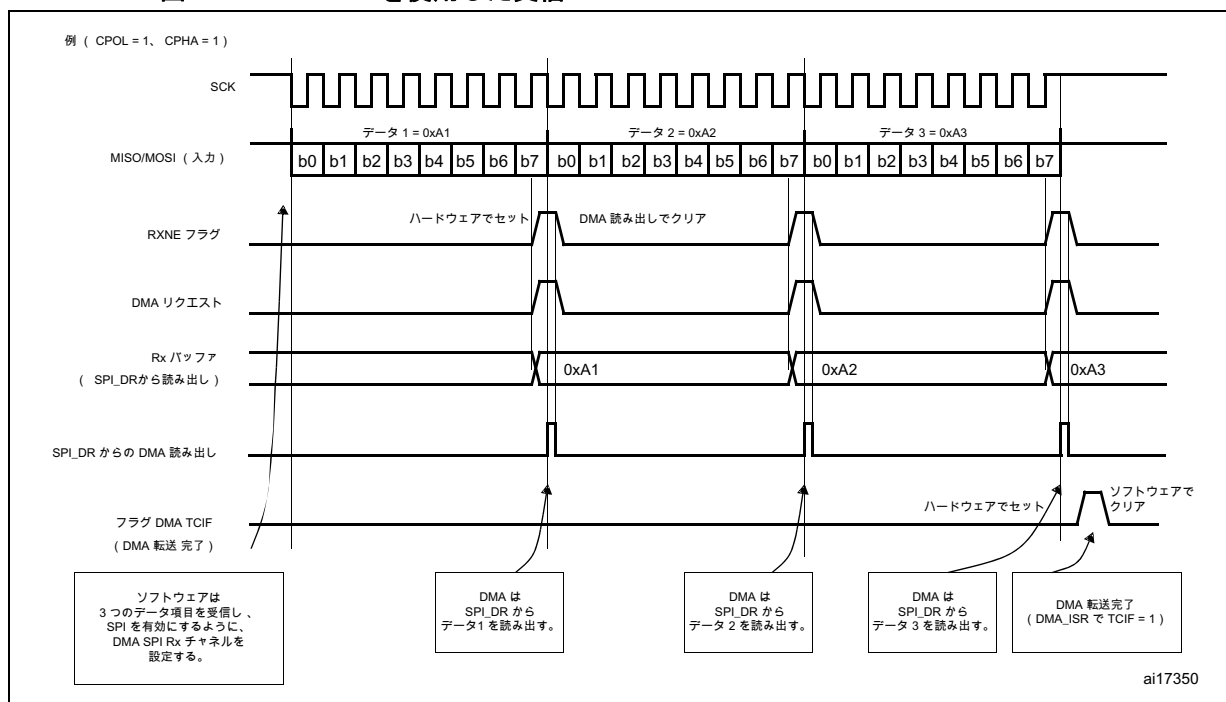


図 277. DMA を使用した受信



31.3.11 SPI ステータスフラグ

アプリケーションが SPI バスの状態を完全に監視できるように、3 つのステータスフラグが用意されています。

Txバッファエンプティフラグ (TXE)

このフラグがセットされると、Txバッファが空であり、次に送信するデータをバッファにロードできることを示します。TXE フラグは、SPI_DR レジスタへの書き込みによってクリアされます。

Rxバッファノットエンプティ (RXNE)

このフラグがセットされると、Rxバッファに有効な受信データがあることを示します。このフラグは、SPI_DR レジスタからの読み出しによってクリアされます。

ビジーフラグ (BSY)

BSY フラグは、ハードウェアによってセット/クリアされます (このフラグへの書き込みは無効)。

BSY フラグがセットされると、SPI上 でデータ転送が進行中であることを示します (SPI バスはビジー)。1つ例外があり、マスタの双方向受信モード (MSTR = 1、BDM = 1、および BDOE = 0) では、BSY フラグが受信時にローレベルに保持されます。

BSY フラグはいくつかのモードで転送の終わりを検出するのに使用できます。それにより、低電力モードに入る前に SPI ペリフェラルクロックが無効化される場合、または NSS パルスの終わりがソフトウェアで処理される場合に、最後の転送の内容が破壊されるのを防ぐことができます。

BSY フラグは、マルチマスタシステムでの書き込み衝突の回避にも役立ちます。

BSY フラグは次のいずれかの条件下でクリアされます。

- SPI が正常に無効にされたとき
- マスタモードで、障害が検出 (MODF ビットが 1 にセットされます) されたとき
- マスタモードで、データ送信が終了し、送信準備ができていない新しいデータがないとき
- スレーブモードで、各データ転送間で少なくとも SPI の 1 クロックサイクルの間、BSY フラグが 0 にセットされているとき

注： *必ず (BSY フラグを使用する代わりに) TXE フラグと RXNE フラグを使用して、データの送受信の処理を行うことを推奨します。*

31.3.12 SPI エラーフラグ

次のいずれかのエラーフラグがセットされていて、ERRIE ビットをセットすることにより割り込みが有効になっている場合、SPI 割り込みが生成されます。

オーバーランフラグ (OVR)

Rxバッファからの前のフレームの読み出し動作が完了していない (RXNE フラグがセットされている) うちに、マスタまたはスレーブが次のデータフレームの受信を完了すると、オーバーラン条件が発生します。

この場合、Rxバッファの内容は新しい受信データによって更新されません。SPI_DR レジスタからの読み出し動作によって、前に受信されたフレームが返されます。その後に送信されたすべてのデータは失われます。

OVR ビットをクリアするには、SPI_DR レジスタを読み出し、続けて SPI_SR レジスタを読み出しアクセスを行います。

モードフォールト (MODF)

モードフォールトは、マスタデバイスが内部 NSS 信号 (NSS ハードウェアモードでは NSS ピン、NSS ソフトウェアモードでは SSI ビット) をローレベルにプルダウンしたときに発生します。これにより、MODF ビットが自動的にセットされます。マスタモードフォールトは、SPI インタフェースに次のような影響を与えます。

- MODF ビットがセットされ、ERRIE ビットがセットされている場合は SPI 割り込みが生成されます。
- SPE ビットがクリアされます。これによって、デバイスからのすべての出力がブロックされ、SPI インタフェースが無効になります。
- MSTR ビットがクリアされ、デバイスは強制的にスレーブモードになります。

MODF ビットをクリアするには、次のソフトウェアシーケンスを実行します。

1. MODF ビットがセットされている間、SPIx_SR レジスタに読み出し/書き込みアクセスを行います。
2. 次に、SPIx_CR1 レジスタに書き込みを行います。

複数の MCU で構成されるシステムでスレーブ間の競合を避けるには、MODF ビットをクリアするシーケンス中、NSS ピンをハイレベルにプルアップする必要があります。このクリアシーケンスの後、SPE ビットと MSTR ビットは、元の状態に戻すことができます。安全のため、MODF ビットがセットされている間、ハードウェアは SPE ビットと MSTR ビットのセットを許可しません。スレーブデバイスでは、MODF ビットはセットできません。ただし、前回のマルチマスタ競合の結果としてセットする場合は例外です。

CRC エラー (CRCERR)

このフラグを使用して、SPIx_CR1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。シフトレジスタに受信された値が、レシーバである SPIx_RXCRC の値と一致しなかった場合、SPIx_SR レジスタの CRCERR フラグがセットされます。フラグはソフトウェアによってクリアされます。

TI モードフレームフォーマットエラー (FRE)

SPI がスレーブモードで動作し、かつ TI モードプロトコルに準拠した設定となっている場合、通信の進行中に NSS パルスが発生すると、TI モードフレームフォーマットエラーが検出されます。このエラーが発生すると、SPIx_SR レジスタの FRE フラグがセットされます。エラー発生時には SPI は無効にされず、この NSS パルスは無視されます。SPI は次の NSS パルスを待ってから新規の転送を開始します。このエラーの検出により 2 バイトのデータが失われるため、データは破壊される可能性があります。

SPIx_SR レジスタを読み出すと、FRE フラグがクリアされます。ERRIE ビットがセットされている場合、NSS エラー検出時に割り込みが生成されます。この場合、データの一貫性が保証されなくなるため、SPI を無効にする必要があり、またスレーブ SPI が再び有効化された場合は、マスタによって通信を再起動する必要があります。

31.4 SPI の特殊機能

31.4.1 TI モード

マスタモードでの TI プロトコル

SPI インタフェースは TI プロトコルと互換性があります。SPIx_CR2 レジスタの FRF ビットを使って、SPI をこのプロトコルに準拠させるように設定することができます。

SPIx_CR1 レジスタにセットされる値によらず、クロックの極性と位相は TI プロトコル要件に必ず適合します。NSS 管理も TI プロトコルに固有なものになります。これにより、この場合の SPIx_CR1 レジスタと SPIx_CR2 レジスタによる NSS 管理の設定 (SSM、SSI、SSOE) ができなくなります。

スレーブモードでは、SPI ポーレートプリスケアラを使用して、現在のトランザクションが終了した時点で、MISO ピンの状態がハイインピーダンスに変化するタイミングを制御します (図 278 を参照)。任意のポーレートが使用できるため、このタイミングを非常に柔軟に決定することができます。ただし、ポーレートは外部マスタクロックポーレートに設定されるのが一般的です。MISO 信号がハイインピーダンス (t_{release}) になるまでの遅延は、内部再同期と SPIx_CR1 レジスタの BR[2:0] ビットで設定されたポーレート値によって変わります。この値は次式で求められます：

$$\frac{t_{\text{baud_rate}}}{2} + 4 \times t_{\text{pclk}} < t_{\text{release}} < \frac{t_{\text{baud_rate}}}{2} + 6 \times t_{\text{pclk}}$$

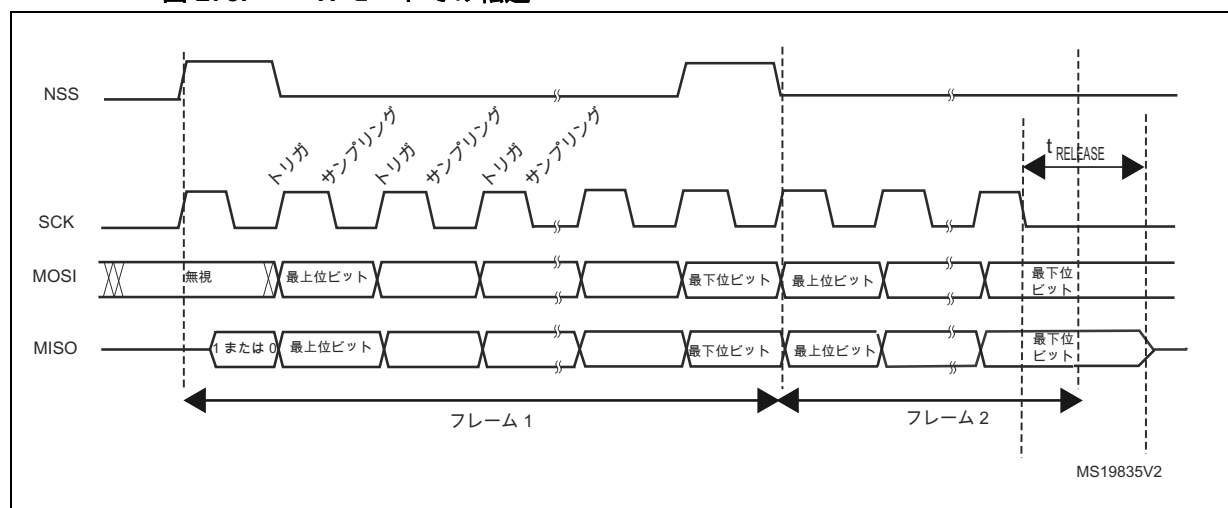
スレーブがデータフレームトランザクション中に NSS パルスの誤配置を検出すると、TIFRE フラグがセットされます。

この機能はモトローラの SPI 通信には使用できません (FRF ビットを 0 に設定)。

注： エラー割り込み (ERRIE = 1) を使ってスレーブ送信専用モードで TI フレームエラーを検出するには、SPI_CR1 レジスタで BIDIMODE と BIDIOE を 1 に設定して SPI を 2 線単方向モードに設定する必要があります。BIDIMODE を 0 に設定すると、OVR が 1 にセットされます。これは、データレジスタは決して読み出されることがなく、エラー割り込みが常に生成されるためです。これに対して BIDIMODE を 1 に設定した場合は、データが受信されず、OVR がセットされることはありません。

図 278 : TI モードでの転送 TI モードが選択されているときの SPI の通信波形を示します。

図 278. TI モードでの転送



31.4.2 CRC 計算

2 つの CRC 計算機（送信および受信データフロー用）が、送信データおよび受信データの信頼性をチェックするために実装されています。SPI は、DFF ビットを介して選択されたデータフォーマットに応じて、CRC8 または CRC16 を計算します。CRC は、SPI_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

CRC の原理

CRC 計算は、SPI が有効（SPE = 1）になる前に、SPIx_CR1 レジスタの CRCEN ビットをセットすることによって有効になります。CRC 値は、各ビットに対して奇数のプログラム可能な多項式の値を使用して計算されます。計算は、SPIx_CR1 レジスタの CPHA ビットと CPOL ビットによって定義されるサンプリングクロックエッジで行われます。計算された CRC 値は、データブロックの最後のみならず、CPU または DMA によって管理される転送に関しても、自動的にチェックされます。受信データをもとに内部で計算された CRC 値とトランスミッタが送信した CRC 値の間に不一致が検出された場合、データ破壊エラーを示すために CRCERR フラグがセットされます。CRC 計算を処理する正しい手順は、SPI の設定および選択された転送管理によって変わります。

注： 多項式の値は必ず奇数でなければなりません。偶数の値はサポートされていません。

CPU によって管理される CRC 転送

通信が開始され、最後のデータフレームが SPIx_DR レジスタで送信または受信されるまで正常に続きます。次に、CRC フレームトランザクションが現在処理中のデータフレームトランザクションの後に行われることを示すために、SPIx_CR1 レジスタの CRCNEXT ビットをセットする必要があります。CRCNEXT ビットは、最後のデータフレームトランザクションの終了前にセットしてください。CRC のトランザクション中は CRC 計算は行われません。

受信した CRC 値は、他のデータフレームと同じように Rx バッファに格納されます。

CRC フォーマットトランザクションは、データシーケンスの最後に通信を行うためにデータフレームをもう一つ受け取ります。

最後の CRC データを受信すると、受信した値と SPIx_RXCRC レジスタの値を比較する自動チェックが行われます。ソフトウェアは、SPIx_SR レジスタの CRCERR フラグをチェックして、データ転送の内容が破壊されているか否かを判断する必要があります。ソフトウェアは、CRCERR フラグに "0" を書き込んでクリアします。

CRC 受信後、CRC 値は Rx バッファに格納され、RXNE フラグをクリアするために SPIx_DR レジスタを読み出す必要があります。

DMA によって管理される CRC 転送

SPI 通信が CRC 通信と DMA モードで有効化される場合、CRC の送受信は通信の最後に自動で行われます（ただし、受信専用モードで CRC データを読み出す場合を除く）。CRCNEXT ビットはソフトウェアで処理する必要はありません。SPI 送信用 DMA チャンネルのカウンタは、転送するデータフレーム数から CRC フレームを除いてセットする必要があります。レシーバ側では、受信した CRC 値はトランザクションの終了時に DMA によって自動的に処理されますが、SPI_DR から受け取る CRC フレームは常にレシーバにロードされますので、ユーザは注意してそれを一掃してください。

転送中に内容の破壊が生じた場合、データと CRC の転送の最後に SPIx_SR レジスタの CRCERR フラグがセットされます。

SPIx_TXCRC および SPIx_RXCRC の値のリセット

SPIx_TXCRC および SPIx_RXCRC の値は、CRC 計算が有効になった時点で自動的にクリアされます。

CRC 機能を有効にした状態で SPI がスレーブモードに設定されると、NSS ピンにハイレベルが入力された場合でも、CRC 計算が行われます。これは、たとえば、通信マスタが複数のスレーブに交互に対処するマルチスレーブ環境の場合に起こることがあります。

スレーブの選択解除 (NSS のハイレベル) と新しいスレーブの選択 (NSS のローレベル) の間では、マスタとスレーブのそれぞれの CRC 計算を再同期するために、マスタ側とスレーブ側の両方で CRC 値をクリアしてください。

CRC をクリアするには、次の手順に従います。

1. SPI を無効にします。
2. CRCEN ビットをクリアします。
3. CRCEN ビットを有効にします。
4. SPI を有効にします。

注 : SPI がスレーブモードのとき、SPE ビットの値に関係なく、CRCEN がセットされるとすぐに、CRC 計算機は SCK スレーブ入力クロックの影響を受けます。CRC 計算の間違いを回避するために、ソフトウェアは、クロックが安定している (定常状態にある) ときのみ、CRC 計算を有効にする必要があります。SPI インタフェースがスレーブとして設定されている場合、データフェーズと CRC フェーズの間で NSS 内部信号をローレベルに維持する必要があります。

31.5 SPI 割り込み

SPI 通信中、割り込みは次のイベントによって生成できます。

- ロードする準備ができている Tx バッファの送信
- Rx バッファのデータ受信
- マスタモードフォールト
- オーバーランエラー
- TI フレームフォーマットエラー

割り込みは個別に有効または無効にできます。

表 128. SPI 割り込みリクエスト

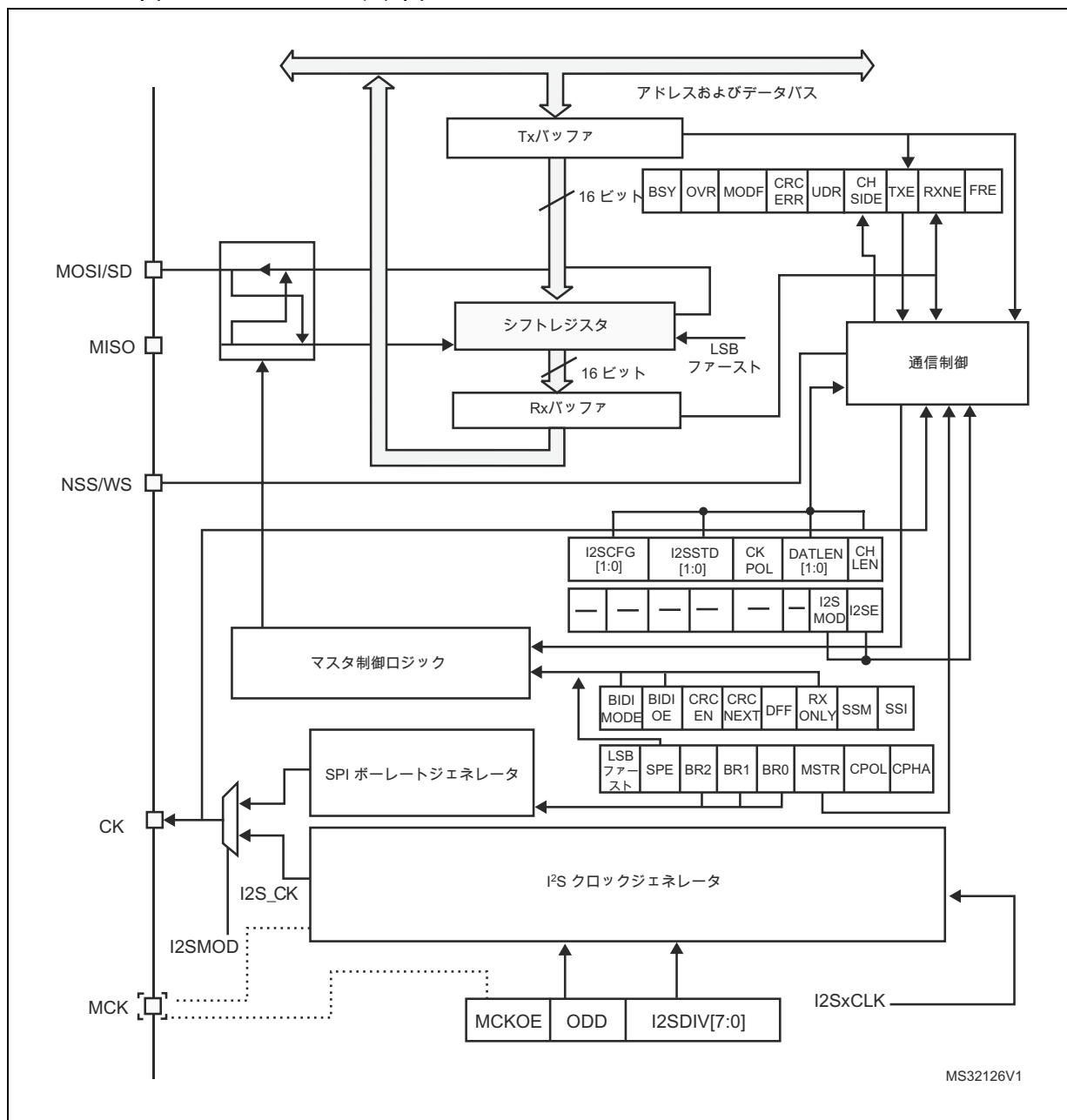
| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|--------------------------|---------|------------|
| ロードする準備ができている Tx バッファの送信 | TXE | TXEIE |
| Rx バッファのデータ受信 | RXNE | RXNEIE |
| マスタモードフォールトイベント | MODF | ERRIE |
| オーバーランエラー | OVR | |
| CRC エラー | CRCERR | |
| TI フレームフォーマットエラー | FRE | |

31.6 I²S の機能説明

31.6.1 I²S の概要

I²S のブロック図を [図 279](#) に示します。

図 279. I²S ブロック図



1. MCK は MISO ピンに配置されます。

SPI はオーディオ I²S インタフェースとして機能することができます。それには、SPIx_I2SCFGR レジスタの I2SMOD ビットをセットして I²S 機能を有効にしてください。このインタフェースは、主に SPI と同じピン、フラグ、および割り込みを使用します。

I²S と SPI は、以下の3 つのピンを共用します。

- SD (シリアルデータ) : MOSI ピンに配置され、2 つの時間多重化データチャネルを送受信します (半二重モードのみ)。
- WS (ワードセレクト) : NSS ピンに配置され、マスタモードではデータ制御信号の出力、スレーブモードでは入力です。
- CK (シリアルクロック) : SCK ピンに配置され、マスタモードではシリアルクロック出力、スレーブモードではシリアルクロック入力です。

外部オーディオデバイスにマスタクロック出力が必要な場合、追加のピンを使用できます。

- MCK (マスタクロック) : 別々に配置され、I²S がマスタモードに設定されている (かつ、SPIx_I2SPR レジスタの MCKOE ビットがセットされている) とき、 $256 \times f_s$ と等しい (f_s はオーディオサンプリング周波数) 設定済みの周波数で生成されたこの追加クロックを出力するために使用されます。

I²S は、マスタモードに設定されているとき、専用のクロックジェネレータを使用して通信クロックを生成します。このクロックジェネレータは、マスタクロック出力のソースでもあります。I²S モードでは、2 つの追加レジスタを使用できます。1 つはクロックジェネレータ設定 SPIx_I2SPR にリンクされ、もう 1 つは汎用 I²S 設定レジスタ SPIx_I2SCFGR (オーディオ規格、スレーブ/マスタモード、データフォーマット、パケットフレーム、クロック極性など) です。

SPIx_CR1 レジスタとすべての CRC レジスタは、I²S モードでは使用されません。同様に、SPIx_CR2 レジスタの SSOE ビットと SPIx_SR レジスタの MODF および CRCERR ビットも使用されません。

I²S は、16 ビット幅モードでのデータ転送に同じ SPI レジスタ (SPIx_DR) を使用します。

31.6.2 サポートされるオーディオプロトコル

3 線バスでは、一般に 2 つのチャネル (右チャネルと左チャネル) で時間多重化されたオーディオデータのみを処理する必要があります。しかしながら、送受信用には 1 つの 16 ビットレジスタしかありません。したがって、各チャネルサイドに対応する適切な値をデータレジスタに書き込んだり、データレジスタからデータを読み出して SPIx_SR レジスタの CHSIDE ビットをチェックして対応するチャネルを識別したりすることは、ソフトウェアの責任です。左チャネルは常に最初に送信され、その後で右チャネルが送信されます (CHSIDE は PCM プロトコルには無関係です)。

4 つのデータおよびパケットフレームを使用できます。データは次のフォーマットで送信されます。

- 16 ビットフレームにパックされた 16 ビットデータ
- 32 ビットフレームにパックされた 16 ビットデータ
- 32 ビットフレームにパックされた 24 ビットデータ
- 32 ビットフレームにパックされた 32 ビットデータ

32 ビットパケットに拡張された 16 ビットデータを使用するとき、最初の 16 ビット (MSB) は上位ビットであり、LSB の 16 ビットは、ソフトウェア動作や DMA リクエストを必要とせずに (1 回の読み込み/書き込み動作のみで) 強制的に 0 にされます。

24 ビットと 32 ビットのデータフレームは、SPIx_DR レジスタとの間で 2 回の CPU 読み出しまたは書き込み動作 (あるいはアプリケーションにとって DMA が望ましい場合は 2 回の DMA 動作) を必要とします。特に 24 ビットのデータフレームの場合、8 つの下位ビットはハードウェアによって 0 のビットで 32 ビットに拡張されます。

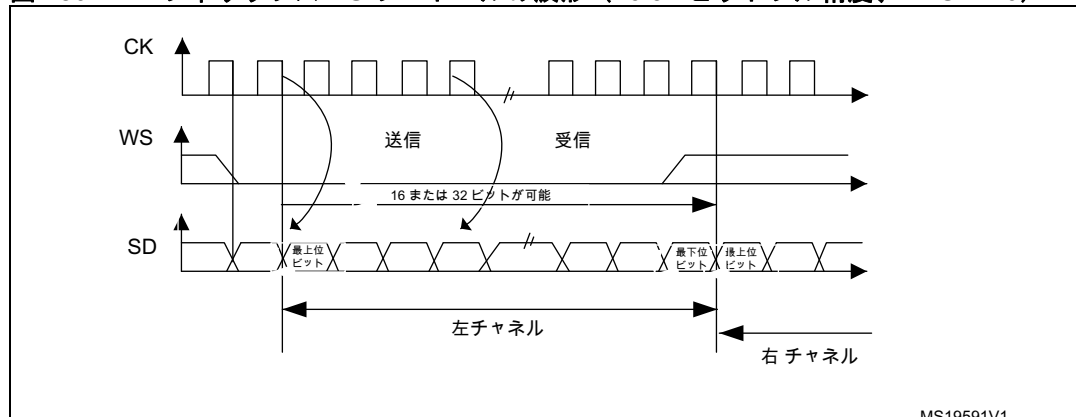
すべてのデータフォーマットと通信規格に対して、最上位ビットは常に最初に送信されます (MSB ファースト)。

I²S インタフェースは、SPIx_I2SCFGR レジスタの I2SSSTD[1:0] と PCMSYNC ビットを使用して設定可能な 4 つのオーディオ規格をサポートします。

フィリップス I²S 規格

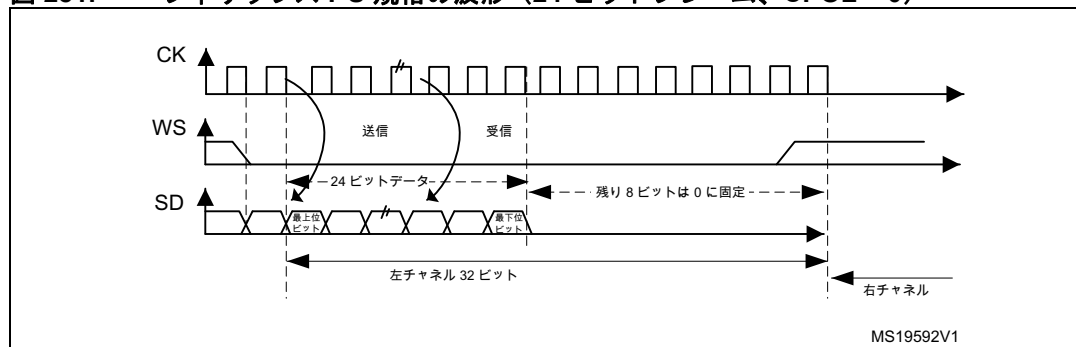
この規格では、どのチャンネルが送信されているかを示すために WS 信号を使用します。この信号が有効になってから 1 CK クロックサイクル後に最初のビット (MSB) が使用可能になります。

図 280. フィリップス I²S プロトコルの波形 (16/32 ビットフル精度、CPOL = 0)



データは、CK の立ち下がりエッジでラッチされ (トランスミッタの場合)、立ち上がりエッジで読み出されます (レシーバの場合)。WS 信号も CK の立ち下がりエッジでラッチされます。

図 281. フィリップス I²S 規格の波形 (24 ビットフレーム、CPOL = 0)

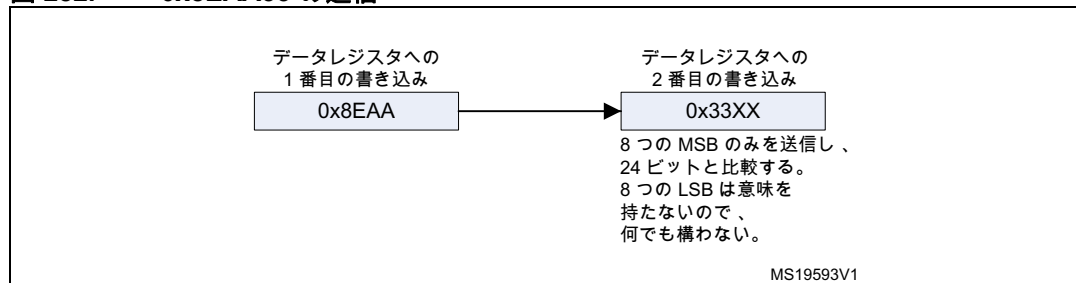


このモードでは、SPIx_DR レジスタに対して 2 回の書き込みまたは読み出し動作が必要です。

● 送信モード

0x8EAA33 を送信する必要がある場合 (24 ビット) :

図 282. 0x8EAA33 の送信



- 受信モード
データ 0x8EAA33 が受信される場合 :

図 283. 0x8EAA33 の受信

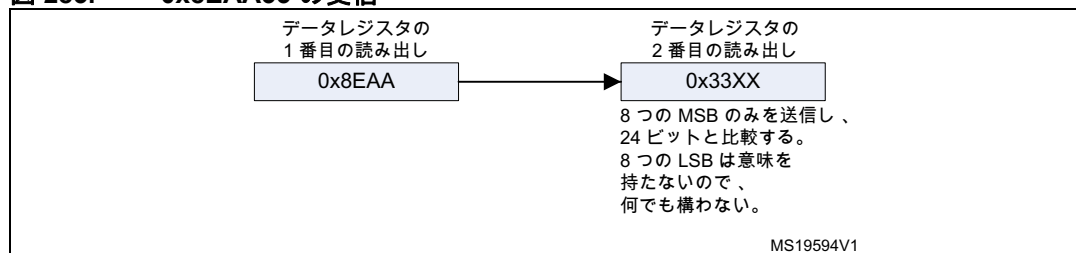
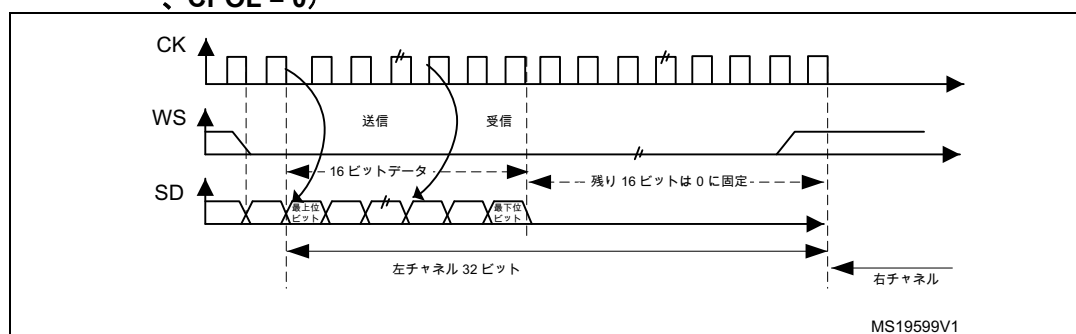


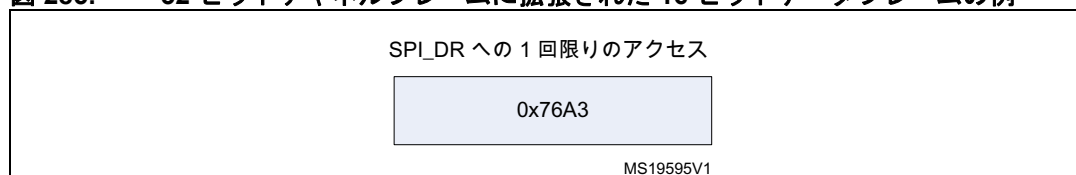
図 284. フィリップス I²S 規格 (32 ビットパケットフレームに拡張された 16 ビット、CPOL = 0)



I²S 設定フェーズで、32 ビットチャンネルフレームに拡張された 16 ビットデータフレームが選択されたとき、SPIx_DR レジスタへのアクセスは 1 回のみ必要です。残りの 16 ビットは、データを 32 ビットフォーマットに拡張するために、ハードウェアによって強制的に 0x0000 にされます。

送信するデータまたは受信したデータが 0x76A3 (32 ビットに拡張された 0x76A30000) であるとき、[図 285](#) に示す動作が要求されます。

図 285. 32 ビットチャンネルフレームに拡張された 16 ビットデータフレームの例



送信では、SPIx_DR に MSB が書き込まれるたびに、TXE フラグがセットされ、可能ならば、SPIx_DR レジスタに新しい送信値をロードするために割り込みが生成されます。これはハードウェアによって行われるため、0x0000 がまだ送信されていない場合でも行われます。

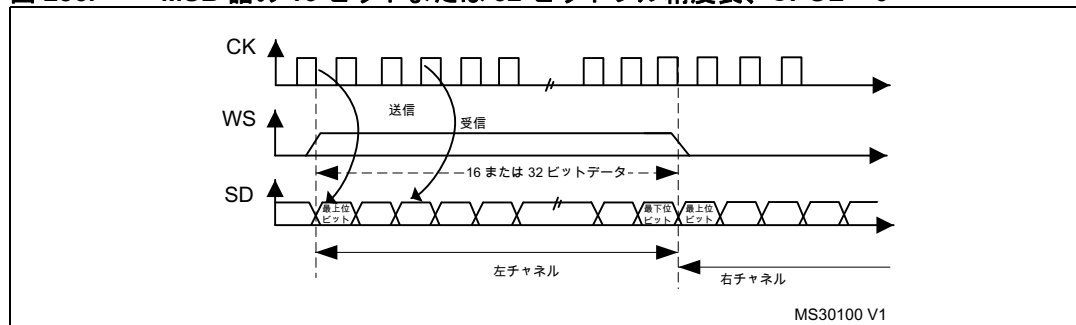
受信では、RXNE フラグがセットされ、可能ならば、最初の 16 MSB ハーフワードの受信時にその割り込みが生成されます。

このように、2 回の書き込みまたは読み出し動作の間にはより多くの時間が設けられるため、アンダーランまたはオーバーラン状態（データ転送の方向に依存）を避けられます。

MSB 詰め規格

この規格では、WS 信号は最初のデータビット（最上位ビット）と同時に生成されます。

図 286. MSB 詰め 16 ビットまたは 32 ビットフル精度長、CPOL = 0



データは、CK の立ち下がりエッジでラッチされ（トランスミッタの場合）、立ち上がりエッジで読み出されます（レシーバの場合）。

図 287. MSB 詰め 24 ビットフレーム長、CPOL = 0

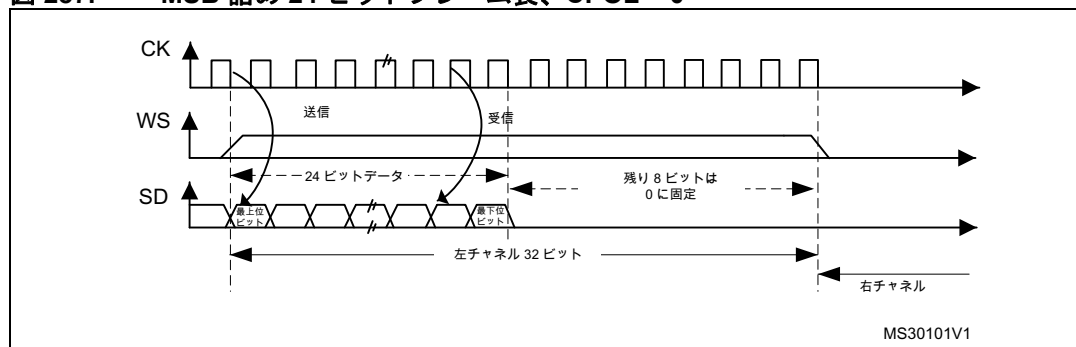
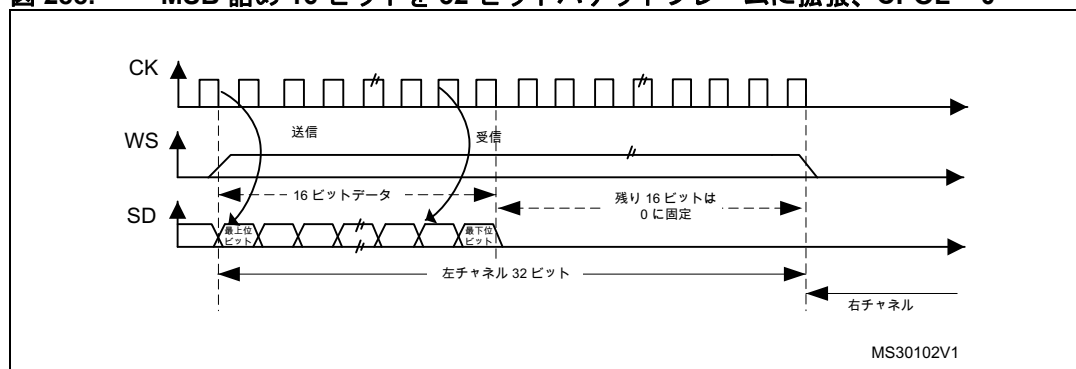


図 288. MSB 詰め 16 ビットを 32 ビットパッケージフレームに拡張、CPOL = 0



LSB 詰め規格

この規格は、MSB 詰め規格と似ています（16 ビットと 32 ビットのフル精度フレームフォーマットに違いはありません）。

図 289. LSB 詰め 16 ビットまたは 32 ビットフル精度、CPOL = 0

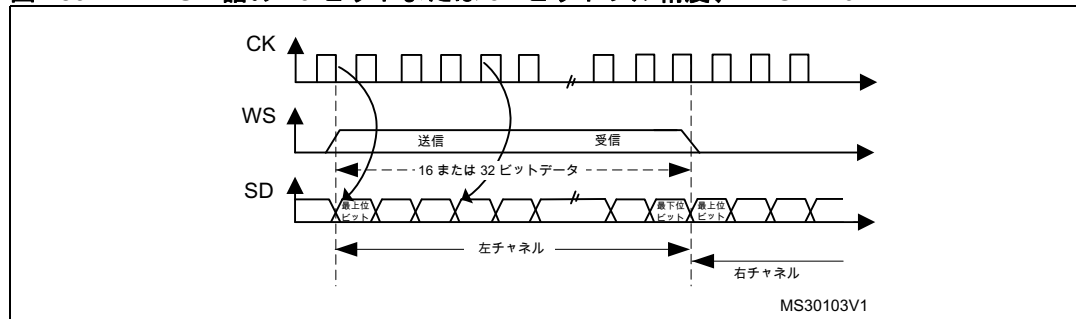
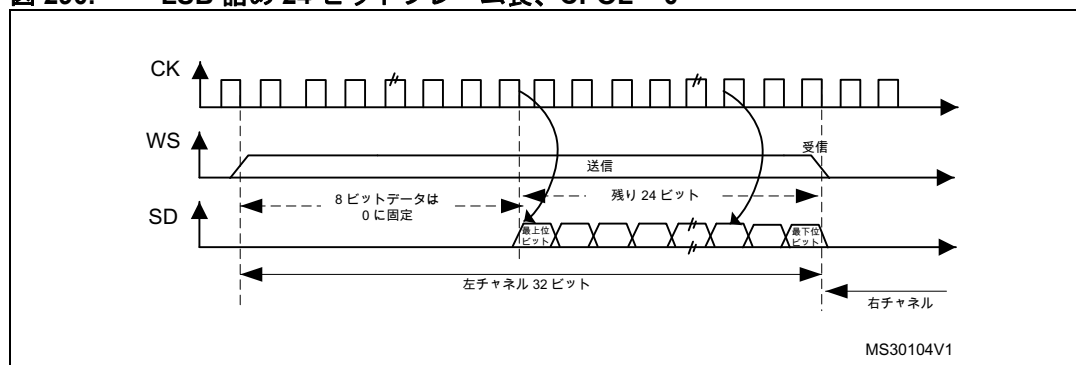


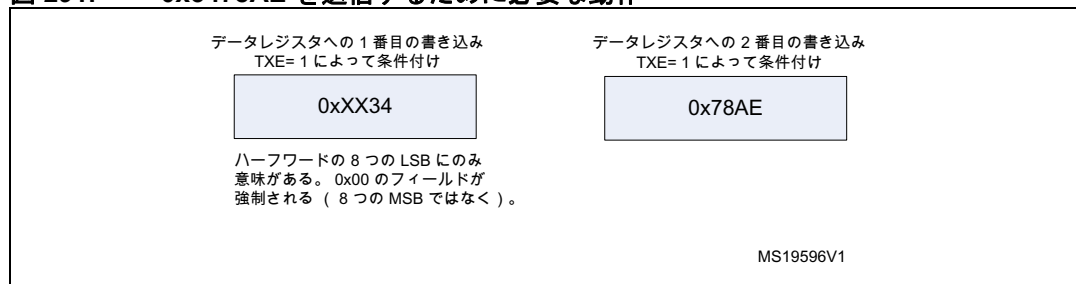
図 290. LSB 詰め 24 ビットフレーム長、CPOL = 0



● 送信モードの場合

データ 0x3478AE を送信する必要がある場合、ソフトウェアまたは DMA によって SPIx_DR レジスタへの 2 回の書き込み動作が必要です。この動作を次に示します。

図 291. 0x3478AE を送信するために必要な動作



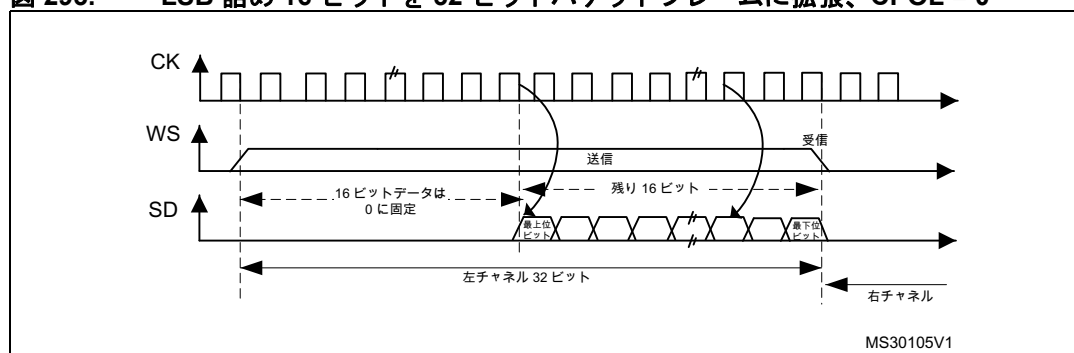
- 受信モードの場合

データ 0x3478AE が受信される場合、RXNE イベントごとに、SPIx_DR レジスタから連続する 2 回の読み出し動作が必要です。

図 292. 0x3478AE の受信に必要な動作



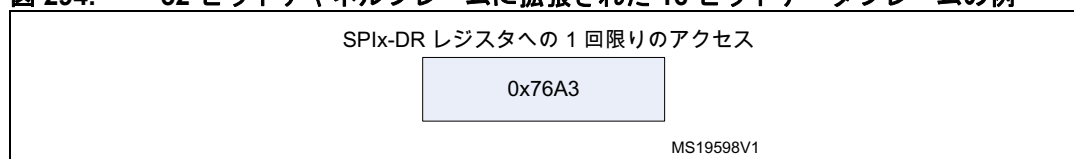
図 293. LSB 詰め 16 ビットを 32 ビットパケットフレームに拡張、CPOL = 0



I²S 設定フェーズで、32 ビットチャネルフレームに拡張された 16 ビットデータフレームが選択されたとき、SPIx_DR レジスタへのアクセスは 1 回のみ必要です。残りの 16 ビットは、データを 32 ビットフォーマットに拡張するために、ハードウェアによって強制的に 0x0000 にされます。この場合、それはハーフワード MSB に相当します。

送信するデータまたは受信したデータが 0x76A3 (32 ビットに拡張された 0x0000 76A3) であるとき、図 294 に示す動作が要求されます。

図 294. 32 ビットチャネルフレームに拡張された 16 ビットデータフレームの例



送信モードでは、TXE イベントが発生すると、アプリケーションは送信するデータ (この場合は 0x76A3) を書き込む必要があります。0x000 フィールドが最初に送信されます (32 ビット拡張部)。有効なデータ (0x76A3) が SD に送信されると、すぐに TXE フラグが再びセットされます。

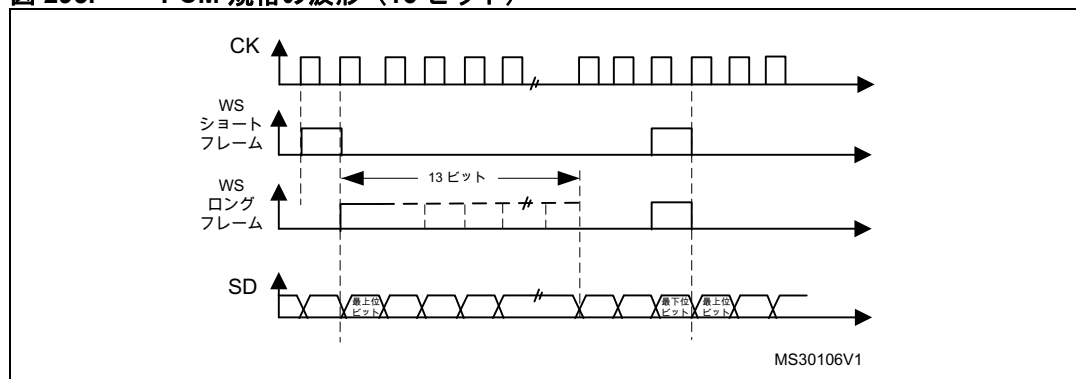
受信モードでは、0x0000 フィールドではない有効なハーフワードが受信されると、すぐに RXNE がアサートされます。

このように、アンダーランやオーバーランの状態を防ぐために、2 回の書き込みまたは読み出し動作の間には、より多くの時間が設けられています。

PCM 規格

PCM 規格では、チャンネルサイド情報を使用する必要はありません。SPIx_I2SCFGR レジスタの PCMSYNC ビットを使用して、2 つの PCM モード（ショートおよびロングフレーム）の使用および設定が可能です。

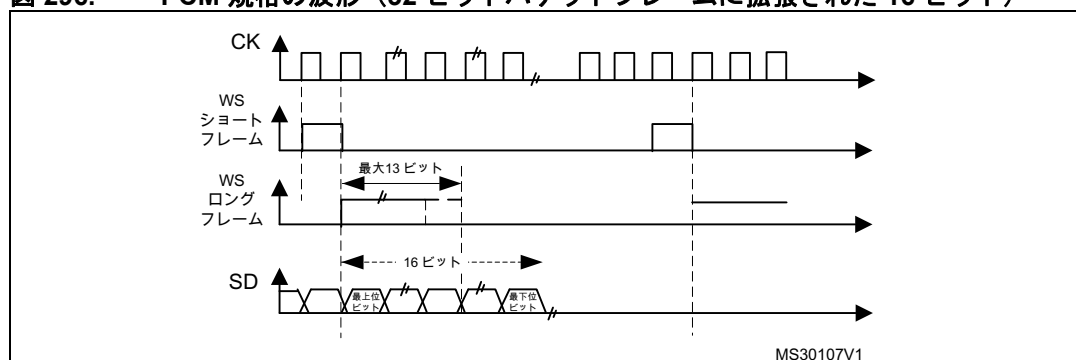
図 295. PCM 規格の波形（16 ビット）



ロングフレーム同期では、WS 信号のアサーション時間はマスタモードで 13 ビットに固定されています。

ショートフレーム同期では、WS 同期信号の長さは、わずか 1 サイクルです。

図 296. PCM 規格の波形（32 ビットパケットフレームに拡張された 16 ビット）



注： 2 つのモード（マスタとスレーブ）と 2 つの同期（ショートとロング）に関しては、スレーブモードでも、連続した 2 つのデータ（したがって 2 つの同期信号）間のビット数を（SPIx_I2SCFGR レジスタの DATLEN および CHLEN ビットで）指定する必要があります。

31.6.3 クロックジェネレータ

The I²S ビットレートにより、I²S データライン上のデータフローと I²S クロック信号周波数が決まります。

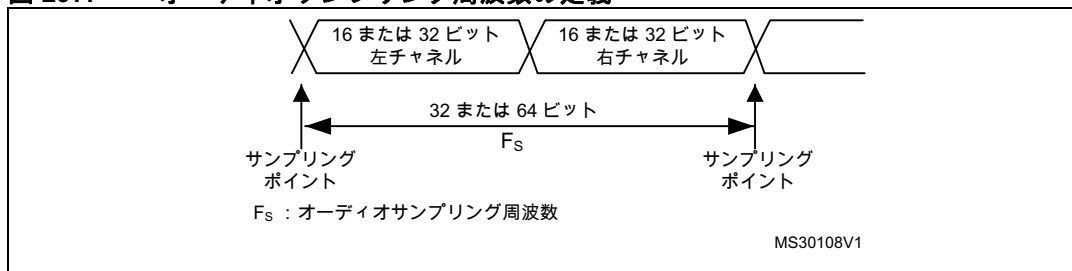
I²S ビットレート = チャンネルあたりのビット数 × チャンネル数 × オーディオサンプリング周波数

16 ビットオーディオ、左チャンネルおよび右チャンネルの場合、I²S ビットレートは次のように算出されます。

$$I^2S \text{ ビットレート} = 16 \times 2 \times f_s$$

パケット長が 32 ビットの場合、I²S ビットレート = 32 x 2 x f_s となります。

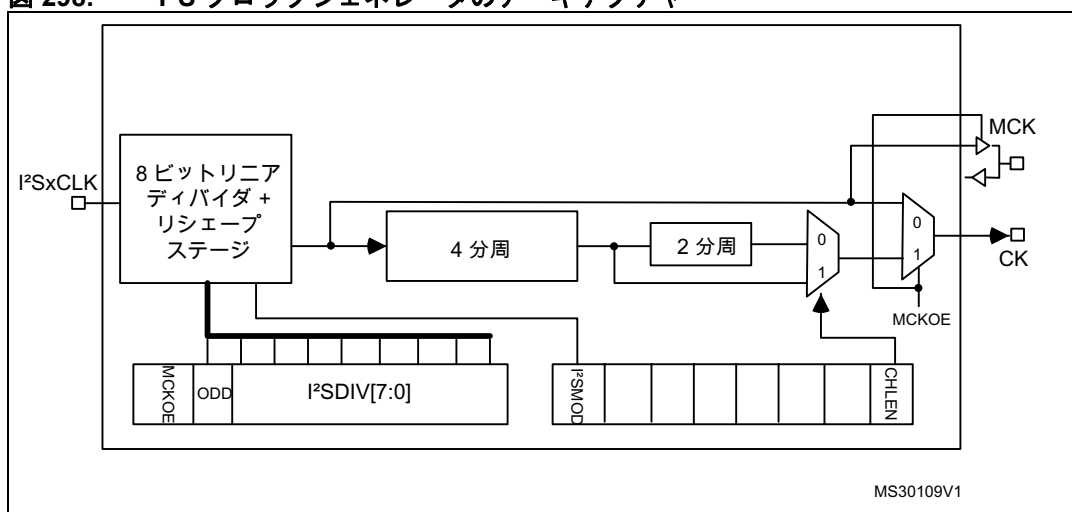
図 297. オーディオサンプリング周波数の定義



マスタモードが設定された場合、希望するオーディオ周波数で通信するために、特定の措置を講じてリニアディバイダを適切にプログラムする必要があります。

図 298 に、通信クロックのアーキテクチャを示します。I²Sx クロックは常にシステムクロックです。

図 298. I²S クロックジェネレータのアーキテクチャ



1. x = 2

オーディオサンプリング周波数の取り得る値は、192 KHz、96 kHz、48 kHz、44.1 kHz、32 kHz、22.05 kHz、16 kHz、11.025 kHz、8 kHz（またはその他のこの範囲内にある値）。希望する周波数を達成するには、次の式に基づいてリニアディバイダをプログラムする必要があります。

マスタクロックが生成される（SPIx_I2SPR レジスタの MCKOE ビットをセットされる）場合

$$f_s = I2SxCLK / [(16 \times 2) \times ((2 \times I2SDIV) + ODD) \times 8] \dots \text{チャンネルフレームが 16 ビット幅の場合}$$

$$f_s = I2SxCLK / [(32 \times 2) \times ((2 \times I2SDIV) + ODD) \times 4] \dots \text{チャンネルフレームが 32 ビット幅の場合}$$

マスタクロックが無効にされた場合（MCKOE ビットをクリア）

$$f_s = I2SxCLK / [(16 \times 2) \times ((2 \times I2SDIV) + ODD)] \dots \text{チャンネルフレームが 16 ビット幅の場合}$$

$$f_s = I2SxCLK / [(32 \times 2) \times ((2 \times I2SDIV) + ODD)] \dots \text{チャンネルフレームが 32 ビット幅の場合}$$

表 129 に、さまざまなクロック設定に対する精度値の例を示します。

注： 最適クロック精度を可能にするその他の設定も可能です。

表 129. 標準 8 MHz HSE を使用した場合のオーディオ周波数精度⁽¹⁾

| SYSCLK (MHz) | データ長 | I2SDIV | I2SODD | MCLK | ターゲット f _S (Hz) | 実際の f _S (Hz) | 誤差 |
|-----------------|------|--------|--------|------|------------------------------|-------------------------|---------|
| 48 | 16 | 8 | 0 | なし | 96000 | 93750 | 2.3438% |
| 48 | 32 | 4 | 0 | なし | 96000 | 93750 | 2.3438% |
| 48 | 16 | 15 | 1 | なし | 48000 | 48387.0968 | 0.8065% |
| 48 | 32 | 8 | 0 | なし | 48000 | 46875 | 2.3438% |
| 48 | 16 | 17 | 0 | なし | 44100 | 44117.647 | 0.0400% |
| 48 | 32 | 8 | 1 | なし | 44100 | 44117.647 | 0.0400% |
| 48 | 16 | 23 | 1 | なし | 32000 | 31914.8936 | 0.2660% |
| 48 | 32 | 11 | 1 | なし | 32000 | 32608.696 | 1.9022% |
| 48 | 16 | 34 | 0 | なし | 22050 | 22058.8235 | 0.0400% |
| 48 | 32 | 17 | 0 | なし | 22050 | 22058.8235 | 0.0400% |
| 48 | 16 | 47 | 0 | なし | 16000 | 15957.4468 | 0.2660% |
| 48 | 32 | 23 | 1 | なし | 16000 | 15957.447 | 0.2660% |
| 48 | 16 | 68 | 0 | なし | 11025 | 11029.4118 | 0.0400% |
| 48 | 32 | 34 | 0 | なし | 11025 | 11029.412 | 0.0400% |
| 48 | 16 | 94 | 0 | なし | 8000 | 7978.7234 | 0.2660% |
| 48 | 32 | 47 | 0 | なし | 8000 | 7978.7234 | 0.2660% |
| 48 | 16 | 2 | 0 | あり | 48000 | 46875 | 2.3430% |
| 48 | 32 | 2 | 0 | あり | 48000 | 46875 | 2.3430% |
| 48 | 16 | 2 | 0 | あり | 44100 | 46875 | 6.2925% |
| 48 | 32 | 2 | 0 | あり | 44100 | 46875 | 6.2925% |
| 48 | 16 | 3 | 0 | あり | 32000 | 31250 | 2.3438% |
| 48 | 32 | 3 | 0 | あり | 32000 | 31250 | 2.3438% |
| 48 | 16 | 4 | 1 | あり | 22050 | 20833.333 | 5.5178% |
| 48 | 32 | 4 | 1 | あり | 22050 | 20833.333 | 5.5178% |
| 48 | 16 | 6 | 0 | あり | 16000 | 15625 | 2.3438% |
| 48 | 32 | 6 | 0 | あり | 16000 | 15625 | 2.3438% |
| 48 | 16 | 8 | 1 | あり | 11025 | 11029.4118 | 0.0400% |
| 48 | 32 | 8 | 1 | あり | 11025 | 11029.4118 | 0.0400% |
| 48 | 16 | 11 | 1 | あり | 8000 | 8152.17391 | 1.9022% |
| 48 | 32 | 11 | 1 | あり | 8000 | 8152.17391 | 1.9022% |

1. この表の値は、さまざまなクロック設定に対するほんの一例です。最適クロック精度を可能にするその他の設定も可能です。



31.6.4 I²S マスタモード

I²S はマスタモードで設定できます。つまり、シリアルクロックは、ワードセレクト信号 WS だけでなく、CK ピン上でも生成されます。マスタクロック (MCK) は、SPIx_I2SPR レジスタの MCKOE ビットで、出力するかしないかを制御できます。

手順

1. 適切なオーディオサンプリング周波数に到達するシリアルクロックポーレートを定義するため、SPIx_I2SPR レジスタの I2SDIV[7:0] ビットを選択します。SPIx_I2SPR レジスタの ODD ビットも定義する必要があります。
2. 通信クロックの一定したレベルを定義するために、CKPOL ビットを選択します。外部の DAC/ADC オーディオコンポーネントにマスタクロック MCK を供給する必要がある場合、SPIx_I2SPR レジスタの MCKOE ビットをセットします (I2SDIV と ODD の値は、MCK 出力の状態に応じて計算する必要があります。詳細については、[セクション 31.6.3: クロックジェネレータ](#)を参照)。
3. SPIx_I2SCFGR レジスタの I2SMOD ビットをセットして I²S 機能を有効にし、I2SSTD[1:0] と PCMSYNC ビットにより I²S 規格を、DATLEN[1:0] ビットによりデータ長を、CHLEN ビットを設定してチャンネルあたりのビット数を、それぞれ選択します。SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットにより I²S マスタモードと方向 (トランスミッタまたはレシーバ) も選択します。
4. 必要の場合は、SPIx_CR2 レジスタに書き込むことによって、可能性のあるすべての割り込みソースと DMA 機能を選択します。
5. SPIx_I2SCFGR レジスタの I2SE ビットをセットする必要があります。

WS と CK は出力モードに設定されます。SPIx_I2SPR の MCKOE ビットがセットされている場合、MCK も出力です。

送信シーケンス

送信シーケンスは、Txバッファにハーフワードが書き込まれたときに開始されます。

Txバッファに書き込まれる最初のデータは、左チャンネルのデータに対応すると想定します。データが Txバッファからシフトレジスタに転送されると、TXE がセットされ、右チャンネルに対応するデータを Txバッファに書き込む必要があります。CHSIDE フラグは、どのチャンネルが送信されるかを示します。CHSIDE フラグは TXE がハイレベルになったときに更新されるため、このフラグは TXE フラグがセットされた場合に意味を持ちます。

完全なフレームとは、左チャンネルのデータ送信と、それに続く右チャンネルのデータ送信であるとみなす必要があります。左チャンネルのみが送信される部分的フレームは実現できません。

データハーフワードは、最初のビット送信時に 16 ビットシフトレジスタに同時にロードされてから、MOSI/SD ピンに MSB ファーストで連続的にシフトアウトされます。TXE フラグは、Txバッファからシフトレジスタへの毎回の転送後にセットされ、SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は割り込みが生成されます。

選択された I²S 規格モードに応じた書き込み動作の詳細については、[セクション 31.6.2: サポートされるオーディオプロトコル](#)を参照してください。

連続したオーディオデータ送信を行うには、現在の送信が終了する前に、次の送信データを SPIx_DR レジスタに書き込む必要があります。

I2SE をクリアすることによって I²S をスイッチオフするには、TXE = 1 および BSY = 0 になるまで待つ必要があります。

受信シーケンス

動作モードは、ポイント 3 を除いて、送信モードの場合と同じです（[セクション 31.6.4: I²S マスタモード](#)に示す手順を参照）。ポイント 3 では、I2SCFG[1:0] ビットを通じてマスタ受信モードを設定する必要があります。

データやチャネルの長さに関係なく、オーディオデータは 16 ビットの packets によって受信されます。つまり、Rxバッファが満杯になるたびに RXNE フラグがセットされ、SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は割り込みが生成されます。データ長とチャネル長の設定にもよりますが、Rxバッファへの 1 回または 2 回の受信によって、オーディオ値が右チャネルまたは左チャネルに受信されることがあります。

RXNE ビットは、SPIx_DR レジスタの読み出しによってクリアされます。

CHSIDE は毎回の受信後に更新されます。CHSIDE は、I²S セルによって生成される WS 信号に反応します。

選択された I²S 規格モードに応じた読み出し動作の詳細については、[セクション 31.6.2: サポートされるオーディオプロトコル](#)を参照してください。

前の受信データがまだ読み出されていない間にデータが受信された場合、オーバーランが生成され、OVR フラグがセットされます。SPIx_CR2 レジスタの ERRIE ビットがセットされている場合、割り込みが生成されてエラーを知らせます。

I²S をスイッチオフするには、I²S が新しいデータ転送を開始することなく転送サイクルを適切に完了できるように、特定の動作が要求されます。そのシーケンスは、データ長とチャネル長の設定、および選択したオーディオプロトコルモードに依存します。ケース別の説明

- 32 ビットチャネル長に拡張された 16 ビットデータ長(DATLEN = 00、CHLEN = 1)、LSB 詰めモードを使用 (I2SSTD = 10)
 - a) 最後から 2 番目の RXNE = 1 (n - 1) を待ちます。
 - b) 次に I²S の 17 クロックサイクルを待ちます (ソフトウェアループを使用)。
 - c) I²S を無効にします (I2SE = 0)。
- 32 ビットチャネル長に拡張された 16 ビットデータ長(DATLEN = 00 および CHLEN = 1)、MSB 詰め、I²S または PCM モード (それぞれ、I2SSTD = 00、I2SSTD = 01、または I2SSTD = 11)
 - a) 最後の RXNE を待ちます。
 - b) 次に I²S の 1 クロックサイクルを待ちます (ソフトウェアループを使用)。
 - c) I²S を無効にします (I2SE = 0)。
- DATLEN と CHLEN のその他すべての組み合わせについては、I2SSTD ビットを通じて選択したオーディオモードが何であれ、次のシーケンスを実行して I²S をスイッチオフします。
 - a) 最後から 2 番目の RXNE = 1 (n - 1) を待ちます。
 - b) 次に I²S の 1 クロックサイクルを待ちます (ソフトウェアループを使用)。
 - c) I²S を無効にします (I2SE = 0)。

注： 転送時、BSY フラグはローレベルに保持されます。

31.6.5 I²S スレーブモード

スレーブ設定の場合、I²S は送信または受信モードに設定できます。動作モードは、主として I²S マスタ設定に関して述べたものと同じ規則に従います。スレーブモードでは、I²S インタフェースによって生成されるクロックはありません。クロックと WS 信号は、I²S インタフェースに接続された外部マスタから入力されます。したがって、ユーザがクロックを設定する必要はありません。

設定ステップを次に示します。

1. SPIx_I2SCFGR レジスタの I2SMOD ビットをセットして I²S モードを選択し、I2SSTD[1:0] ビットにより I²S 規格を、DATLEN[1:0] ビットによりデータ長を、CHLEN ビットを設定してフレームのチャンネルあたりのビット数を、それぞれ選択します。SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットにより、スレーブのモード（送信または受信）も選択します。
2. 必要の場合は、SPIx_CR2 レジスタに書き込むことによって、可能性のあるすべての割り込みソースと DMA 機能を選択します。
3. SPIx_I2SCFGR レジスタの I2SE ビットをセットする必要があります。

送信シーケンス

送信シーケンスは、外部マスタデバイスがクロックを送信したとき、NSS_WS 信号がデータの転送を要求したときに開始されます。スレーブを有効にしなければ、外部マスタは通信を開始しません。I²S データレジスタは、マスタが通信を開始する前に、ロードする必要があります。

I²S、MSB 詰め、および LSB 詰めモードの場合、データレジスタに書き込まれる最初のデータは、左チャンネルのデータに対応します。通信が開始されると、データはTxバッファからシフトレジスタに転送されます。次に、右チャンネルのデータを I²S データレジスタに書き込むように要求するために、TXE フラグがセットされます。

CHSIDE フラグは、どのチャンネルが送信されるかを示します。スレーブモードでは、マスタ送信モードに比べて、CHSIDE は外部マスタからの WS 信号に反応します。つまり、スレーブが最初のデータの送信準備をしなければ、マスタはクロックを生成できません。WS アサーションは、最初に送信される左チャンネルに対応します。

注： I2SE は、マスタの最初のクロックが CK ラインに到達するよりも、少なくとも 2 PCLK サイクル前に書き込まれる必要があります。

データハーフワードは、最初のビット送信時に内部バスから 16 ビットシフトレジスタに同時にロードされてから、MOSI/SD ピンに MSB ファーストで連続的にシフトアウトされます。TXE フラグは、Txバッファからシフトレジスタへの毎回の転送後にセットされ、SPIx_CR2 レジスタの TXEIE ビットがセットされている場合は割り込みが生成されます。

なお、Txバッファへの書き込みの前に、TXE フラグが 1 であることを確認する必要があります。

選択された I²S 規格モードに応じた書き込み動作の詳細については、[セクション 31.6.2: サポートされるオーディオプロトコル](#)を参照してください。

連続したオーディオデータ送信を行うには、現在の送信が終了する前に、次の送信データを SPIx_DR レジスタに書きこむ必要があります。次のデータ通信の最初のクロックエッジよりも前にデータが SPIx_DR レジスタに書き込まれない場合、アンダーランフラグがセットされ、割り込みが生成されることがあります。これによって、転送データに誤りがあることがソフトウェアに知らされます。SPIx_CR2 レジスタの ERRIE ビットがセットされた場合、SPIx_SR レジスタの UDR フラグがハイレベルになると割り込みが生成されます。この場合、I²S をスイッチオフし、左チャンネルからデータ転送をリスタートする必要があります。

I2SE をクリアすることによって I²S をスイッチオフするには、TXE = 1 および BSY = 0 になるまで待つ必要があります。

受信シーケンス

動作モードは、ポイント 1 を除いて、送信モードの場合と同じです（[セクション 31.6.5: I²S スレーブモード](#)に示す手順を参照）。ポイント 1 では、SPIx_I2SCFGR レジスタの I2SCFG[1:0] ビットを通じてマスタ受信モードを設定する必要があります。

データやチャンネルの長さに関係なく、オーディオデータは 16 ビットの packets によって受信されます。つまり、Rx バッファがフルになるたびに SPIx_SR レジスタの RXNE フラグがセットされ、SPIx_CR2 レジスタの RXNEIE ビットがセットされている場合は割り込みが生成されます。データ長とチャンネル長の設定にもよりますが、Rx バッファへの 1 回または 2 回の受信によって、オーディオ値が右チャンネルまたは左チャンネルに受信されることがあります。

CHSIDE フラグは、SPIx_DR レジスタから読み出されるデータが受信されるたびに更新されます。このフラグは、外部マスタコンポーネントによって管理される外部 WS ラインに反応します。

RXNE ビットは、SPIx_DR レジスタの読み出しによってクリアされます。

選択された I²S 規格モードに応じた読み出し動作の詳細については、[セクション 31.6.2: サポートされるオーディオプロトコル](#)を参照してください。

前の受信データがまだ読み出されていない間にデータが受信された場合、オーバーランが生成され、OVR フラグがセットされます。SPIx_CR2 レジスタの ERRIE ビットがセットされた場合、エラーを示すために割り込みが生成されます。

受信モードで I²S をスイッチオフするには、最後の RXNE = 1 を受信した直後に I2SE をクリアする必要があります。

注： 外部マスタコンポーネントには、オーディオチャンネルを介して 16 ビットまたは 32 ビットの packets でデータを送受信する機能が必要です。

31.6.6 I²S ステータスフラグ

アプリケーションが I²S バスの状態を完全に監視できるように、3 つのステータスフラグが用意されています。

ビジーフラグ (BSY)

BSY フラグは、ハードウェアによってセット/クリアされます（このフラグへの書き込みは無効）。このフラグは I²S の通信層の状態を示します。

BSY がセットされると、I²S が通信中でビジーであることを示します。マスタ受信モード (I2SCFG=11) には 1 つの例外があり、BSY フラグは受信時にローレベルに保持されます。

ソフトウェアが I²S を無効にする必要がある場合、BSY フラグは転送の終わりを検出するために役立ちます。これによって、最後の転送データの破壊を回避します。このため、下記の手順を厳守する必要があります。

BSY フラグは転送が開始されるとセットされます。ただし、I²S がマスタ受信モードにある場合を除きます。

BSY フラグをクリアするタイミング

- 転送が完了したとき（ただし、通信が連続的だと思われるマスタ送信モードの場合を除く）
- I²S が無効化されたとき

通信が連続的な場合

- マスタ送信モードでは、BSY フラグはすべての転送期間を通じてハイレベルに保持されます。
- スレーブモードでは、BSY フラグは、各転送間で I²S の 1 クロックサイクルの間ローレベルになります。



注： 各データの送受信の処理には BSY フラグを使用しないでください。代わりに、TXE フラグと RXNE フラグを使用することをお勧めします。

Txバッファエンプティフラグ (TXE)

このフラグがセットされると、Txバッファはエンプティ（空）であり、次に送信するデータをバッファにロードできることを示します。送信されるデータがすでにTxバッファに格納されているとき、TXE フラグはリセットされます。TXE フラグは、I²S が無効にされている（I2SE ビットがリセット）ときにもリセットされます。

Rxバッファノットエンプティ (RXNE)

このフラグがセットされると、Rxバッファに有効な受信データがあることを示します。このフラグは、SPIx_DR レジスタが読み出されるとリセットされます。

チャネルサイドフラグ (CHSIDE)

送信モードでは、このフラグは TXE がハイレベルになるとリフレッシュされます。このフラグは、SD 上の転送データが属するチャネルサイドを示します。スレーブ送信モードでアンダーランエラーイベントが発生した場合、このフラグは信頼できないため、通信を再開する前に、I²S をスイッチオフし、さらにスイッチオンする必要があります。

受信モードでは、このフラグは SPIx_DR にデータが受信されるとリフレッシュされます。このフラグは、どちらのチャネルサイドからデータが受信されたかを示します。なお、エラー（OVR など）が発生した場合、このフラグは無意味になるため、I²S を無効にし、さらに有効にすることによってリセットする必要があります（変更が必要な場合は設定する）。

このフラグは、PCM 規格では意味を持ちません（ショートとロングフレームの両モード）。

SPIx_SR の OVR または UDR フラグがセットされ、SPIx_CR2 の ERRIE ビットもセットされると、割り込みが生成されます。この割り込みをクリアするには、割り込みソースをいったんクリアした後で、SPIx_SR ステータスレジスタを読み出します。

31.6.7 I²S エラーフラグ

I²S セルには 3つのエラーフラグがあります。

アンダーランフラグ (UDR)

スレーブ送信モードでは、ソフトウェアが SPIx_DR にまだ値をロードしていない間に、データ送信用の最初のクロックが現われると、このフラグがセットされます。このフラグは、SPIx_I2SCFGR レジスタの I2SMOD ビットがセットされると使用できます。SPIx_CR2 レジスタの ERRIE ビットがセットされている場合は、割り込みを生成できます。

UDR ビットは、SPIx_SR レジスタの読み出し動作によってクリアされます。

オーバーランフラグ (OVR)

このフラグがセットされるのは、データが受信され、前のデータが SPIx_DR レジスタからまだ読み出されていないときです。結果として、受信データは失われます。SPIx_CR2 の ERRIE ビットがセットされている場合、割り込みが生成されることがあります。

この場合、受信バッファの内容は、送信側のデバイスからの新しい受信データによって更新されません。SPIx_DR レジスタへの読み出し動作によって、前に正しく受信されたデータが返されます。それ以降に送信されたすべてのハーフワードは失われます。

OVR ビットをクリアするには、SPIx_DR レジスタを読み出し、続けて SPIx_SR レジスタに読み出しアクセスを行います。

フレームエラーフラグ (FRE)

このフラグは、I²S がスレーブモードに設定された場合にのみハードウェアによってセットすることができます。このフラグは、スレーブが WS ラインの変化を想定していないときに外部マスタが WS ラインを変化させようとしている場合にセットされます。同期が失われた場合、この状態から回復し、外部マスタデバイスと I²S スレーブデバイスを再同期させるためには、次のステップに従います。

- 1. I²S を無効にします。
- 2. 正しいレベルが WS ラインで検出されたとき (WS ラインは I²S モードではハイレベル、MSB 詰めまたは LSB 詰めあるいは PCM モードではローレベル)、再度有効にします。

マスタデバイスとスレーブデバイスの間の同期外れは、SCK 通信クロック上または WS フレーム同期ライン上のノイズの多い環境に起因する可能性があります。ERRIE ビットをセットすれば、エラー割り込みを生成させることができます。同期外れフラグ (FRE) は、ステータスレジスタを読み出すときに、ソフトウェアでクリアされます。

31.6.8 I²S 割り込み

表 130 に I²S 割り込みのリストを示します。

表 130. I²S 割り込みリクエスト

| 割り込みイベント | イベントフラグ | イネーブル制御ビット |
|-------------------|---------|------------|
| 送信バッファエンプティフラグ | TXE | TXEIE |
| 受信バッファノットエンプティフラグ | RXNE | RXNEIE |
| オーバーランエラー | OVR | ERRIE |
| アンダーランエラー | UDR | |
| フレームエラーフラグ | FRE | |

31.6.9 DMA の機能

I²S モードでは、DMA は SPI モードとまったく同じように機能します。異なる点は、I²S モードではデータ転送保護システムがないため、CRC 機能を使用できない点のみです。

31.7 SPI および I²S レジスタ

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。さらに、SPI_DR へは8 ビット単位でアクセスできます。

レジスタの説明で使用されている略語のリストについては、[セクション 1.1](#) を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

31.7.1 SPI 制御レジスタ 1 (SPI_CR1) (I²S モードでは使用しません)

アドレスオフセット：0x00

リセット値：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------|------------|-----------|-------------|-----|------------|-----|-----|--------------|-----|----------|----|----|------|------|------|
| BIDI MODE | BIDI OE | CRC EN | CRC NEXT | DFF | RX ONLY | SSM | SSI | LSB FIRST | SPE | BR [2:0] | | | MSTR | CPOL | CPHA |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15 **BIDIMODE** : 双方向データモードイネーブル

このビットは、共通の双方向データラインを 1 本使用して、半二重通信を有効にします。双方向モードがアクティブのときは、RXONLY ビットをクリアされたままにします。

0 : 2 線単方向データモードを選択します。

1 : 1 線双方向データモードを選択します。

注： このビットは I²S モードでは使用しません。

ビット 14 **BIDIOE** : 双方向モードでの出力イネーブル

双方向モードでの転送方向は、このビットと BIDIMODE ビットを組み合わせで選択します。

0 : 出力は無効です（受信専用モード）。

1 : 出力は有効です（送信専用モード）。

注： このビットは I²S モードでは使用しません。

MISO ピンがスレーブモードで使用されている間、マスタモードでは、MOSI ピンが使用されます。

ビット 13 **CRCEN** : ハードウェア CRC 計算イネーブル

0 : CRC 計算は無効です。

1 : CRC 計算は有効です。

注： 正しい動作のためには、このビットへの書き込みは、SPI が無効 (SPE = 0) のときにのみ行います。

このフラグは I²S モードでは使用しません。

ビット 12 **CRCNEXT** : CRC 転送指示

0 : データフェーズです（CRC フェーズなし）。

1 : 次の転送は CRC です（CRC フェーズ）。

注： SPI が全二重モードまたは送信専用モードに設定されている場合、最後のデータが SPI_DR に書き込まれた直後に、CRCNEXT ビットを書き込む必要があります。

SPI が受信専用モードに設定されている場合、最後から 2 番目のデータが受信された後に CRCNEXT ビットをセットする必要があります。

転送が DMA で管理される場合、このビットはクリアされた状態に保持する必要があります。

このフラグは I²S モードでは使用しません。

ビット 11 DFF : データフレームフォーマット

- 0 : 送受信に 8 ビットデータフレームフォーマットが選択されます。
- 1 : 送受信に 16 ビットデータフレームフォーマットが選択されます。

注 : 正しい動作のためには、このビットへの書き込みは、SPI が無効 (SPE = 0) のときにのみ行います。
このフラグは I²S モードでは使用しません。

ビット 10 RXONLY : 受信専用モードイネーブル

このビットは、データ受信専用の単方向ラインを 1 本使用して、単方向通信を有効にします。受信専用モードがアクティブのときは、BIDIMODE ビットをクリアされたままにします。

このビットはマルチスレーブシステムでも役立ちます。そのシステムでは、この特定のスレーブはアクセスされず、アクセスされたスレーブからの出力は破壊されません。

- 0 : 全二重 (送受信)
- 1 : 出力は無効です (受信専用モード)。

注 : このビットは I²S モードでは使用しません。

ビット 9 SSM : ソフトウェアスレーブ管理

SSM ビットがセットされているとき、NSS ピンの入力 SSI ビットからの値に置き換えられます。

- 0 : ソフトウェアスレーブ管理は無効です。
- 1 : ソフトウェアスレーブ管理は有効です。

注 : このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 8 SSI : 内部スレーブ選択

このビットは、SSM ビットがセットされているときにのみ有効です。このビットの値は強制的に NSS ピンに設定され、NSS ピンの IO 値は無視されます。

注 : このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 7 LSBFIRST : フレームフォーマット

- 0 : MSB が最初に送信されます。
- 1 : LSB が最初に送信されます。

注 : このビットは、通信中には変更しないでください。
このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 6 SPE : SPI イネーブル

- 0 : ペリフェラルは無効です。
- 1 : ペリフェラルは有効です。

注 : このビットは I²S モードでは使用しません。

SPI を無効にするときは、[セクション 31.3.9 : SPI を無効にする手順](#)に記載されている手順に従ってください。

ビット 5:3 BR[2:0] : ボーレート制御

- 000 : f_{HCLK} /2
- 001 : f_{HCLK} /4
- 010 : f_{HCLK} /8
- 011 : f_{HCLK} /16
- 100 : f_{HCLK} /32
- 101 : f_{HCLK} /64
- 110 : f_{HCLK} /128
- 111 : f_{HCLK} /256

注 : これらのビットは、通信中には変更しないでください。
これらのビットは I²S モードでは使用されません。

- ビット 2 **MSTR** : マスタ選択
- 0 : スレーブ設定
 - 1 : マスタ設定
- 注 :** このビットは、通信中には変更しないでください。
このフラグは I²S モードでは使用しません。
- ビット 1 **CPOL** : クロック極性
- 0 : アイドル時に CK は 0 となります。
 - 1 : アイドル時に CK は 1 となります。
- 注 :** このビットは、通信中には変更しないでください。
このビットは I²S モードおよび SPI TI モードでは使用しません。
- ビット 0 **CPHA** : クロック位相
- 0 : 最初のクロック遷移が最初のデータキャプチャエッジです。
 - 1 : 2 番目のクロック遷移が最初のデータキャプチャエッジです。
- 注 :** このビットは、通信中には変更しないでください。
このビットは I²S モードおよび SPI TI モードでは使用しません。

31.7.2 SPI 制御レジスタ 2 (SPI_CR2)

アドレスオフセット : 0x04

リセット値 : 0x0000 :

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|-------|--------|-------|-----|------|------|---------|---------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXEIE | RXNEIE | ERRIE | FRF | Res. | SSOE | TXDMAEN | RXDMAEN |
| | | | | | | | | r/w | r/w | r/w | r/w | | r/w | r/w | r/w |

- ビット 15:8 予約済みであり、リセット値に保持する必要があります。
- ビット 7 **TXEIE** : Txバッファエンプティ割り込みイネーブル
- 0 : TXE 割り込みはマスクされます。
 - 1 : TXE 割り込みはマスクされません。TXE フラグがセットされたとき、割り込みリクエストの生成に使用されます。
- ビット 6 **RXNEIE** : Rxバッファノットエンプティ割り込みイネーブル
- 0 : RXNE 割り込みはマスクされます。
 - 1 : RXNE 割り込みはマスクされません。RXNE フラグがセットされたとき、割り込みリクエストの生成に使用されます。
- ビット 5 **ERRIE** : エラー割り込みイネーブル
- このビットは、エラー条件が発生したとき (SPI モードでは OVR、CRCERR、MODF、FRE ; I²S モードでは UDR、OVR、FRE)、割り込みの生成を制御します。
 - 0 : エラー割り込みはマスクされます。
 - 1 : エラー割り込みは有効です。
- ビット 4 **FRF** : フレームフォーマット
- 0 : SPI モトローラモード
 - 1 : SPI TI モード
- 注 :** このビットは I²S モードでは使用しません。
- ビット 3 予約済みハードウェアによって 0 に固定されています。



この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

ビット 2 **SSOE** : SS 出力イネーブル

- 0 : マスタモードで SS 出力は無効にされ、セルはマルチマスタ設定で機能できます。
1 : セルが有効であるとき、マスタモードで SS 出力は有効です。セルはマルチマスタ環境では機能できません。

注 : このビットは I²S モードおよび SPI TI モードでは使用しません。

ビット 1 **TXDMAEN** : Txバッファ DMA イネーブル

- このビットがセットされると、TXE フラグがセットされるたびに DMA リクエストが行われます。
0 : Txバッファ DMA は無効です。
1 : Txバッファ DMA は有効です。

ビット 0 **RXDMAEN** : Rxバッファ DMA イネーブル

- このビットがセットされると、RXNE フラグがセットされるたびに DMA リクエストが行われます。
0 : Rxバッファ DMA は無効です。
1 : Rxバッファ DMA は有効です。

31.7.3 SPI ステータスレジスタ (SPI_SR)

アドレスオフセット : 0x08

リセット値 : 0x0002 :

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|-----|-----|-----|------|------------|-----|--------|-----|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | FRE | BSY | OVR | MODF | CRC ERR | UDR | CHSIDE | TXE | RXNE |
| | | | | | | | r | r | r | r | rc_w0 | r | r | r | r |

ビット 15:9 予約済みハードウェアによって 0 に固定されています。

ビット 8 **FRE** : フレームエラー

- 0 : フレームエラーはありません。
1 : フレームエラーが発生しました。
このビットは、ハードウェアによってセットされ、SPIx_SR レジスタを読み出すとソフトウェアでクリアされます。
このビットは、選択したオーディオプロトコルに関係なく、SPI TI モードまたは I²S モードで使用されます。また、スレーブモードにおいて予想されないタイミングで起こるNSS または WS ライン上の変化を検出し、外部マスタデバイスとスレーブの間の同期外れに関する情報を提供します。

ビット 7 **BSY** : ビジーフラグ

- 0 : SPI (または I²S) はビジー状態ではありません。
1 : SPI (または I²S) が通信ビジー状態であるか、またはTxバッファが空ではありません。
このフラグはハードウェアによってセット/クリアされます。

注 : **BSY フラグを使用する際は注意が必要です。セクション 31.3.11 : SPI ステータスフラグ およびセクション 31.3.9 : SPI を無効にする手順を参照してください。**

ビット 6 **OVR** : オーバーランフラグ

- 0 : オーバーランは発生していません。
1 : オーバーランが発生しました。
このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、[セクション 31.3.12 : SPI エラーフラグ](#)を参照してください。

ビット 5 **MODF** : モードフォールト

- 0 : モードフォールトは発生していません。
1 : モードフォールトが発生しました。
このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、[セクション 31.4 \(811 ページ\)](#)を参照してください。

注 : このビットは I²S モードでは使用しません。



- ビット 4 **CRCERR** : CRC エラーフラグ

0 : 受信した CRC 値が SPI_RXCRCR 値と一致します。
1 : 受信した CRC 値が SPI_RXCRCR 値と一致しません。
このフラグは、ハードウェアによってセットされ、ソフトウェアによって 0 を書き込むことでクリアされます。
注： このビットは I²S モードでは使用しません。
- ビット 3 **UDR** : アンダーランフラグ

0 : アンダーランは発生していません。
1 : アンダーランが発生しました。
このフラグは、ハードウェアによってセットされ、ソフトウェアシーケンスによってリセットされます。ソフトウェアシーケンスについては、[セクション 31.6.7 : I²S エラーフラグ](#)を参照してください。
注： このビットは SPI モードでは使用しません。
- ビット 2 **CHSIDE** : チャネルサイド

0 : 左チャネルを送信する必要があるか、または受信が行われました。
1 : 右チャネルを送信する必要があるか、または受信が行われました。
注： このビットは、SPI モードに対しては使用しません。また、PCM モードでは意味を持ちません。
- ビット 1 **TXE** : 送信バッファエンプティ

0 : Txバッファは空ではありません。
1 : Txバッファは空です。
- ビット 0 **RXNE** : 受信バッファノットエンプティ

0 : Rxバッファは空です。
1 : Rxバッファは空ではありません。

31.7.4 SPI データレジスタ (SPI_DR)

アドレスオフセット : 0x0C
リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|----------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DR[15:0] | | | | | | | | | | | | | | | |
| RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW | RW |

- ビット 15:0 **DR[15:0]** : データレジスタ

受信したデータまたは送信されるデータ
データレジスタは 2 つのバッファに分割されます。一方は書き込み用 (送信バッファ) であり、他方は読み出し用 (受信バッファ) です。データレジスタへの書き込みは、Txバッファへの書き込みになります。データレジスタからの読み出しでは、Rxバッファに保持されている値が返されます。
注： 以下の注記は SPI モードに適用されます。
送受信されるデータは、データフレームフォーマットの選択ビット (SPI_CR1 レジスタの DFF ビット) に応じて、8 ビットまたは 16 ビットになります。正しい動作のためには、この選択を行ってから、SPI を有効にする必要があります。
8 ビットデータフレームの場合、バッファは 8 ビットであり、レジスタの LSB (SPI_DR[7:0]) のみが送受信に使用されます。受信モードのとき、レジスタの MSB (SPI_DR[15:8]) は強制的に 0 にされます。
16 ビットデータフレームの場合、バッファは 16 ビットであり、レジスタ全体 (SPI_DR[15:0]) が送受信に使用されます。

31.7.5 SPI CRC 多項式レジスタ (SPI_CRCPR) (I²S モードでは使用しません)

アドレスオフセット : 0x10

リセット値 : 0x0007

| | | | | | | | | | | | | | | | |
|---------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CRCPOLY[15:0] | | | | | | | | | | | | | | | |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 15:0 **CRCPOLY[15:0]** : CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を格納します。

CRC 多項式 (0007h) は、このレジスタのリセット値です。必要に応じて、別の多項式を設定することができます。

注 : これらのビットは I²S モードでは使用しません。

31.7.6 SPI RX CRC レジスタ (SPI_RXCRCR) (I²S モードでは使用しません)

アドレスオフセット : 0x14

リセット値 : 0x0000

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

ビット 15:0 **RXCRC[15:0]** : Rx CRC レジスタ

CRC 計算が有効なとき、RxCRC[15:0] ビットには、その後に受信したバイトから算出された CRC 値が格納されています。このレジスタは、SPI_CR1 レジスタの CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPI_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。

データフレームフォーマットが 8 ビットデータに設定された場合 (SPI_CR1 の DFF ビットがクリアされている)、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビットデータフレームフォーマットが選択された場合 (SPI_CR1 レジスタの DFF ビットがセットされている)、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

注 : BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。これらのビットは I²S モードでは使用しません。

31.7.7 SPI TX CRC レジスタ (SPI_TXCRCR) (I²S では使用しません)

アドレスオフセット : 0x18

リセット値 : 0x0000

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

ビット 15:0 **TXCRC[15:0]** : Tx CRC レジスタ

CRC 計算が有効なとき、TxCRC[7:0] ビットには、その後に送信されたバイトから算出された CRC 値が格納されます。このレジスタは、SPI_CR1 の CRCEN ビットに 1 が書き込まれたときにリセットされます。CRC は、SPI_CRCPR レジスタにプログラムされた多項式を使用して連続的に計算されます。データフレームフォーマットが 8 ビットデータに設定された場合 (SPI_CR1 の DFF ビットがクリアされている)、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 規格に基づいて行われます。

16 ビットデータフレームフォーマットが選択された場合 (SPI_CR1 レジスタの DFF ビットがセットされている)、このレジスタの 16 ビット全体が考慮されます。CRC 計算は、任意の CRC16 規格に基づいて行われます。

注: *BSY フラグがセットされているときにこのレジスタを読み出すと、誤った値が返されることがあります。
これらのビットは I²S モードでは使用しません。*

31.7.8 SPI_I2S 設定レジスタ (SPI_I2SCFGR)

アドレスオフセット : 0x1C

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|--------|------|--------|----|-------------|------|--------|----|-------|--------|----|-------|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | I2SMOD | I2SE | I2SCFG | | PCMSYN C | Res. | I2SSTD | | CKPOL | DATLEN | | CHLEN |
| | | | | rW | rW | rW | rW | rW | | rW | rW | rW | rW | rW | rW |

ビット 15:12 予約済みであり、リセット値に保持する必要があります。

ビット 11 **I2SMOD** : I²S モード選択

0 : SPI モードが選択されます。

1 : I²S モードが選択されます。

注: *このビットは、SPI または I²S が無効なときに設定してください。*

ビット 10 **I2SE** : I²S イネーブル

0 : I²S ペリフェラルは無効です。

1 : I²S ペリフェラルは有効です。

注: *このビットは SPI モードでは使用しません。*

ビット 9:8 **I2SCFG** : I²S 設定モード

00 : スレーブ - 送信

01 : スレーブ - 受信

10 : マスタ - 送信

11 : マスタ - 受信

注: *このビットは、I²S が無効なときに設定してください。*

SPI モードでは使用しません。

ビット 7 **PCMSYNC** : PCM フレーム同期

0 : ショートフレーム同期

1 : ロングフレーム同期

注 : このビットは、I²SSTD = 11 (PCM 規格使用) の場合にのみ意味を持ちます。

SPI モードでは使用しません。

ビット 6 予約済みであり、ハードウェアによって 0 に固定されています。

ビット 5:4 **I²SSTD** : I²S 規格選択

00 : フィリップス I²S 規格

01 : MSB 詰め規格 (左詰め)

10 : LSB 詰め規格 (右詰め)

11 : PCM 規格

I²S 規格に関する詳細は、[セクション 31.6.2 \(815 ページ\)](#) を参照してください。SPI モードでは使用しません。

注 : 正しい動作のためには、これらのビットは、I²S が無効のときに設定してください。

ビット 3 **CKPOL** : 定常状態クロック極性

0 : I²S クロックの定常状態はローレベルです。

1 : I²S クロックの定常状態はハイレベルです。

注 : 正しい動作のためには、このビットは、I²S が無効のときに設定してください。

このビットは SPI モードでは使用しません。

ビット 2:1 **DATLEN** : 転送されるデータ長

00 : 16 ビットデータ長

01 : 24 ビットデータ長

10 : 32 ビットデータ長

11 : 設定禁止

注 : 正しい動作のためには、これらのビットは、I²S が無効のときに設定してください。

このビットは SPI モードでは使用しません。

ビット 0 **CHLEN** : チャネル長 (オーディオチャネルごとのビット数)

0 : 16 ビット幅

1 : 32 ビット幅

ビット書き込み動作は、DATLEN = 00 のときにのみ意味を持ちます。そうでない場合、書き込まれた値とは無関係に、チャネル長はハードウェアによって 32 ビットに固定されます。SPI mode では使用しません。

注 : 正しい動作のためには、このビットは、I²S が無効のときに設定してください。

31.7.9 SPI_I2S プリスケーラレジスタ (SPI_I2SPR)

アドレスオフセット : 0x20
リセット値 : 0000 0010 (0x0002)

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|-------|-----|--------|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | MCKOE | ODD | I2SDIV | | | | | | | |
| | | | | | | rw | rw | rw | | | | | | | |

- ビット 15:10 予約済みであり、リセット値に保持する必要があります。
- ビット 9 **MCKOE** : マスタクロック出カインエーブル
- 0 : マスタクロック出力は無効です。
 - 1 : マスタクロック出力は有効です。
- 注 :** このビットは、I²S が無効なときに設定してください。このビットは、I²S がマスタモードのときにのみ使用します。
- このビットは SPI モードでは使用しません。
- ビット 8 **ODD** : プリスケーラの奇数分周比
- 0 : 真のディバイダ値 = I2SDIV *2
 - 1 : 真のディバイダ値 = (I2SDIV * 2)+1
- [セクション 31.6.3 \(821 ページ\)](#) を参照してください。SPI モードでは使用しません。
- 注 :** このビットは、I²S が無効なときに設定してください。このビットは、I²S がマスタモードのときにのみ使用します。
- ビット 7:0 **I2SDIV** : I²S リニアプリスケーラ
- I2SDIV [7:0] = 0 または I2SDIV [7:0] = 1 は禁止されている値です。
- [セクション 31.6.3 \(821 ページ\)](#) を参照してください。SPI モードでは使用しません。
- 注 :** これらのビットは、I²S が無効なときに設定してください。これらのビットは、I²S がマスタモードのときにのみ使用します。

31.7.10 SPI レジスタマップ

次の表に、SPI レジスタマップとリセット値を示します。

表 131. SPI レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | | | | |
|-----------|-------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|---------------|-------|-------|---------|------|--------|---------|-------|----------|-------|----------|--------|------|---------|---------|------|---|---|--|--|--|
| 0x00 | SPI_CR1 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BIDMODE | BIDOE | CRCEN | CRCNEXT | DFE | RXONLY | SSM | SSI | LSBFIRST | SPE | BR [2:0] | | MSTR | CPOL | CPHA | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x04 | SPI_CR2 | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TXEIE | RXNEIE | ERRIE | FRF | Res. | SSE | TXDMAEN | RXDMAEN | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x08 | SPI_SR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | FRE | BSY | OVR | MODF | CRCERR | UDR | CHSIDE | TXE | RXNE | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | | | | |
| 0x0C | SPI_DR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DR[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x10 | SPI_CRCPR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CRCPOLY[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | | | | |
| 0x14 | SPI_RXCRCR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RxCRC[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x18 | SPI_TXCRCR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | TxCRC[15:0] | | | | | | | | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x1C | SPI_I2SCFGR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | I2SMOD | I2SE | I2SCFG | PCMSYNC | Res. | I2SSTD | CKPOL | DATLEN | CHLEN | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| 0x20 | SPI_I2SPR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | MCKOE | ODD | I2SDIV | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | | | |

レジスタ境界アドレスについては、表 2 (43 ページ) を参照してください。

32 ユニバーサルシリアルバスフルスピードデバイスインタフェース (USB)

32.1 概要

USB ペリフェラルは、フルスピードの USB 2.0 バスと APB バスの間のインタフェースを実装します。

USB サスペンド／レジュームがサポートされるため、節電のためにデバイスのクロックを停止できます。

32.2 USB の主な機能

- USB 仕様バージョン 2.0 フルスピード準拠
- 1 から 8 まで設定可能なエンドポイント数
- 最大 1024 バイトの専用パケットバッファメモリ SRAM
- 巡回冗長検査 (CRC) の生成／チェック、NRZI (Non-return-to-zero Inverted) 方式のエンコード／デコード、およびビットスタッフィング
- アイソクロナス転送サポート
- ダブルバッファバブルク／アイソクロナスエンドポイントサポート
- USB サスペンド／レジューム操作
- フレームロッククロックパルス生成
- USB 2.0 リンク電源管理サポート
- バッテリ充電仕様 Revision 1.2 サポート
- USB 接続／切断機能 (USB DP ライン上の制御可能な組み込みプルアップレジスタ)

32.3 USB の実装

表 132 に、STM32L0x3 デバイスでの USB 実装を示します。

表 132. STM32L0x3 USB の実装

| USB の機能 ⁽¹⁾ | |
|-----------------------------|-------------------------|
| | USB |
| エンドポイント数 | 8 |
| 専用パケットバッファメモリ SRAM のサイズ | 1024 バイト ⁽²⁾ |
| 専用パケットバッファメモリ SRAM のアクセス方式 | 2 x 16 ビット/ワード |
| USB 2.0 リンク電源管理 (LPM) のサポート | X |
| バッテリ充電検出 (BCD) のサポート | X |
| USB_DP ライン上の組み込みプルアップレジスタ | X |

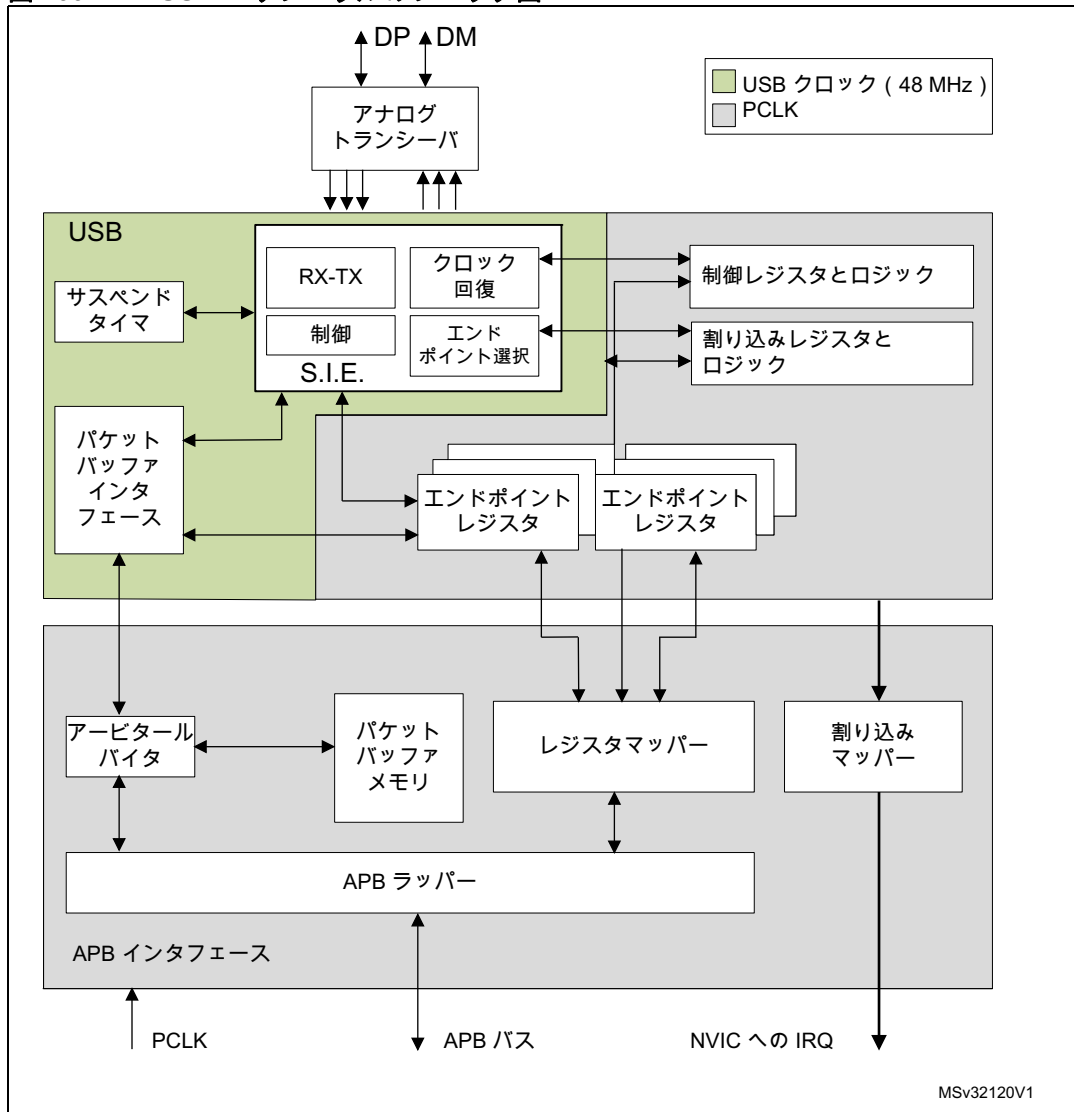
1. X = サポート。
2. RCC_APB1ENR レジスタで CAN ペリフェラルクロックが有効なときには、最初の 768 のみが USB で使用可能であり、残りの 256 バイトは CAN によって使用されます。



32.4 USART の機能詳細

図 299 に、USB ペリフェラルのブロック図を示します。

図 299. USB ペリフェラルのブロック図



USB ペリフェラルは、ホスト PC とマイクロコントローラによって実装される機能の間に USB 準拠接続を提供します。ホスト PC とシステムメモリ間のデータ転送は、USB ペリフェラルによって直接アクセスされる専用バッファメモリを通じて行われます。この専用メモリのサイズは最大 1024 バイトであり、最大 16 の単方向または 8 つの双方向エンドポイントを使用できます。USB ペリフェラルは USB ホストとインタフェースして、トークン packets を検出し、データの送受信を処理し、USB 標準の要求に従ってハンドシェイク packets を処理します。トランザクションのフォーマット化は、CRC の生成とチェックも含めて、ハードウェアによって行われます。

各エンドポイントは、エンドポイント関連のメモリ領域の位置、その大きさ、または送信バイト数を示すバッファ記述ブロックに関連付けられます。有効な機能／エンドポイントのペアのトークンが USB ペリフェラルによって検出されると、関連するデータ転送が行われます（必要であり、エンドポイントが設定されている場合）。USB ペリフェラルによってバッファされたデータは、内部 16 ビットレジスタに配置され、専用バッファへのメモリアクセスが行われます。すべてのデータが転送され

ると、必要な場合には、転送方向に従って、適切なハンドシェイクパケットが USB 経由で生成され、または転送の方向によって期待されます。

トランザクションの終了時には、エンドポイント固有の割り込みが生成され、ステータスレジスタを読み出すか、別の割り込み応答ルーチンを使用します。マイクロコントローラは、以下を決めることができます。

- 処理するエンドポイント
- エラーが発生した場合に行うトランザクションのタイプ（ビットスタッフィング、フォーマット、CRC、プロトコル、ACK の欠如、オーバーラン／アンダーランなど）

アイソクロナス転送と高スループットバルク転送に対する特別なサポートが提供され、ダブルバッファ私用が実装されることにより、USB ペリフェラルが使用できるバッファが常にあり、マイクロコントローラ用のバッファが別にあります。

必要なときには、制御レジスタに書き込むことによって、ユニットを低電力モード（SUSPEND モード）にできます。このとき、すべての静的消費電流が回避され、USB クロックをスローダウンまたは停止できます。低電力モード中に USB 入力でアクティビティが検出されると、デバイスは非同期でウェイクアップします。特殊な割り込みソースをウェイクアップラインに直接接続することにより、システムは通常のクロック生成をただちに再開し、直接クロックの開始／停止をサポートできます。

32.4.1 USB ブロックの説明

USB ペリフェラルは USB インタフェースに関するすべての機能を実装し、これには以下のブロックが含まれます。

- USB 物理インタフェース (USB PHY) : このブロックは、外部 USB ホストへの電気インタフェースを維持します。ディファレンシャルアナログトランシーバ本体、制御可能な組み込みプルアップレジスタ (USB_DP ラインに接続)、およびバッテリー充電検出 (BCD) のサポート、同じ USB_DP および USB_DM ラインでの多重化を含みます。アナログトランシーバの出力有効制御信号 (アクティブロー) は、USB_NOE により外部から供給されます。これを使用して、アクティビティ LED を駆動したり、実際の通信方向に関する情報を他の回路に提供することができます。
- シリアルインタフェースエンジン (SIE) : このブロックの機能には、同期パターン認識、ビットスタッフィング、CRC 生成およびチェック、PID 確認／生成、およびハンドシェイク評価が含まれます。USB トランシーバとインタフェースし、ローカルデータストレージ用のパケットバッファインタフェースによって提供される仮想バッファを使用する必要があります。このユニットは、フレーム開始 (SOF)、USB_Reset、データエラーなどの USB ペリフェラルイベントと、送信終了、パケットの正しい受信などのエンドポイント関連イベントに従って信号を生成し、これらの信号は割り込みの生成に使用されます。
- タイマ : このブロックは、フレーム開始ロッククロックパルスを生成し、トラフィックが 3 ms の間、受信されなかったときにはグローバルサスペンドを (ホストから) 検出します。
- パケットバッファインタフェース : このブロックは、ローカルメモリを管理して、送信と受信の両方のためのバッファのセットを柔軟に実装します。SIE から着信したリクエストに従って適切なバッファを選び、エンドポイントレジスタによって示されるメモリアドレスに配置します。パケットの終わりまで、各バイトの交換後にアドレスをインクリメントして、交換バイト数を追跡し、バッファが最大容量までオーバーランするのを防ぎます。
- エンドポイント関連レジスタ : 各エンドポイントには、エンドポイントのタイプと現在のステータスを含んだレジスタが関連付けられます。単方向／シングルバッファエンドポイントの場合、1 つのレジスタで 2 つのエンドポイントを実装できます。レジスタの数は 8 つであり、最大 16 の単方向／シングルバッファまたは最大 7 つのダブルバッファエンドポイントを任意の組み合わせで使用できます。たとえば、USB ペリフェラルをプログラムして、4 つのダブルバッファエンドポイントと 8 つのシングルバッファ／単方向エンドポイントを使用できます。

- 制御レジスタ：これらは USB ペリフェラル全体のステータスに関する情報を含むレジスタであり、レジュームやパワーダウンなどの USB イベントを強制するために使用されます。
- 割り込みレジスタ：これらは割り込みマスクとイベントのレコードを含みます。割り込み理由、割り込みステータスの問い合わせや、保留中の割り込みのステータスをクリアするために使用できます。

注： * エンドポイント 0 は、常にシングルバッファモードでの転送の制御に使用されます。

USB ペリフェラルは APB バスに APB インタフェースを通じて接続され、以下のブロックを含みます。

- パケットメモリ：これは、パケットバッファを物理的に含むローカルメモリです。パケットバッファインタフェースによって使用して、データ構造を作成し、アプリケーションソフトウェアから直接アクセスできます。パケットメモリのサイズは、最大 1024 バイトであり、512 個のハーフワードと 16 ビットで構成されます。
- アービター：このブロックは、APB バスおよび USB インタフェースから着信するリクエストを受け入れます。APB アクセスに優先順位を付けることによってコンフリクトを解決するとともに、常にメモリ帯域幅の半分をすべての USB 転送を完了するために確保しています。このタイムデュプレックス方式によって仮想デュアルポート SRAM が実装され、USB トランザクション中のメモリアccessを可能にします。この方式により、任意の長さのマルチワード APB 転送も可能です。
- レジスタマッパー：このブロックは、USB ペリフェラルのさまざまなバイト長およびビット長のレジスタを APB によってアドレス指定された構造化された 16 ビット長のハーフワードセットに集めます。
- APB ラッパー：これは、メモリとレジスタに APB へのインタフェースを提供します。また、USB ペリフェラル全体を APB アドレス空間にマップします。
- 割り込みマッパー：このブロックは、USB イベントによって割り込みを生成する方法を選択し、それらを NVIC にマップするために使用されます。

32.5 プログラミングに関する考慮事項

以下のセクションでは、アプリケーションソフトウェア開発を容易にするために、USB ペリフェラルとアプリケーションプログラムの間の予期される相互作用について説明します。

32.5.1 汎用 USB デバイスのプログラミング

このパートでは、USB 準拠動作を得るためにアプリケーションソフトウェアに必要な主なタスクについて説明します。ほとんどの汎用 USB イベントに関連するアクションが考慮に入れられており、ダブルバッファエンドポイントとアイソクロナス転送という特殊ケースについて、それぞれのパラグラフで説明します。システムリセットを別にする、アクションは常に USB ペリフェラルによって開始され、以下に述べる USB イベントの 1 つによって駆動されます。

32.5.2 システムリセットとパワーオンリセット

システムリセットおよびパワーオンリセットの際、アプリケーションソフトウェアが最初に行うべき操作は、必要なクロック信号のすべてを USB ペリフェラルに与えることであり、その後、リセット信号をネゲートして、レジスタにアクセスできるようにしなければなりません。初期化シーケンス全体は、以下のとおりです。

最初のステップとして、アプリケーションソフトウェアは、デバースクロック管理ロジックによって提供される関連の制御ビットを使用して、レジスタマクロセルクロックを有効にし、マクロセル固有のリセット信号をネゲートする必要があります。



その後、CNTR レジスタの PDWN ビットを使用して、USB トランシーバに関連するデバイスのアナログ部分をオンにする必要があります、これには特別な処理が必要です。このビットは、ポートトランシーバに供給される内部電圧基準をオンにすることを目的としています。この回路にはスタートアップ時間が定義されており（データシートで指定されている $t_{STARTUP}$ ）、この間の USB トランシーバの動作は定義されていません。したがって、CNTR レジスタの PDWN ビットをセットした後、この時間だけ待つ必要があります、その後、USB 部分のリセット条件を解除します（CNTR レジスタの FRES ビットをクリアすることによって）。ISTR レジスタをクリアすると、他のマクロセル動作が有効になる前に、誤って保留中の割り込みが削除されます。

システムリセット時には、マイクロコントローラは必要なすべてのレジスタとパケットバッファ記述テーブルを初期化して、USB ペリフェラルが割り込みとデータ転送を正しく生成できるようにする必要があります。エンドポイント固有ではないレジスタはすべて、アプリケーションソフトウェアの必要に応じて初期化する必要があります（有効にする割り込みの選択、パケットバッファのアドレスの選択など）。USB リセットの場合、その後、プロセスが続行します（以下のパラグラフを参照）。

USB リセット (RESET 割り込み)

このイベントが発生すると、USB ペリフェラルは、前のパラグラフで説明した初期化後のシステムリセットと同じ条件になります。すなわち、すべてのエンドポイントレジスタで通信が無効になります（USB ペリフェラルはパケットに応答しません）。USB リセットイベントに対して、USB 機能を有効にして、USB アドレスを 0 にし、デフォルトの制御エンドポイントのみを実装します（エンドポイントアドレスも 0）。これは、USB_DADDR レジスタの機能有効 (EF) ビットをセットし、EP0R レジスタに関連するパケットバッファを初期化することによって行われます。USB エンুমレーションプロセス時、ホストは一意なアドレスをこのデバイスに割り当て、これが USB_DADDR レジスタの ADD[6:0] ビットに書き込まれなければならない、他に必要なエンドポイントを設定する必要があります。

RESET 割り込みが受信されると、アプリケーションソフトウェアは、割り込みをトリガしたリセットシーケンスの終了から 10 ms 以内に、USB 機能 0 のデフォルトのエンドポイントを再び有効にする必要があります。

パケットバッファの構造と用途

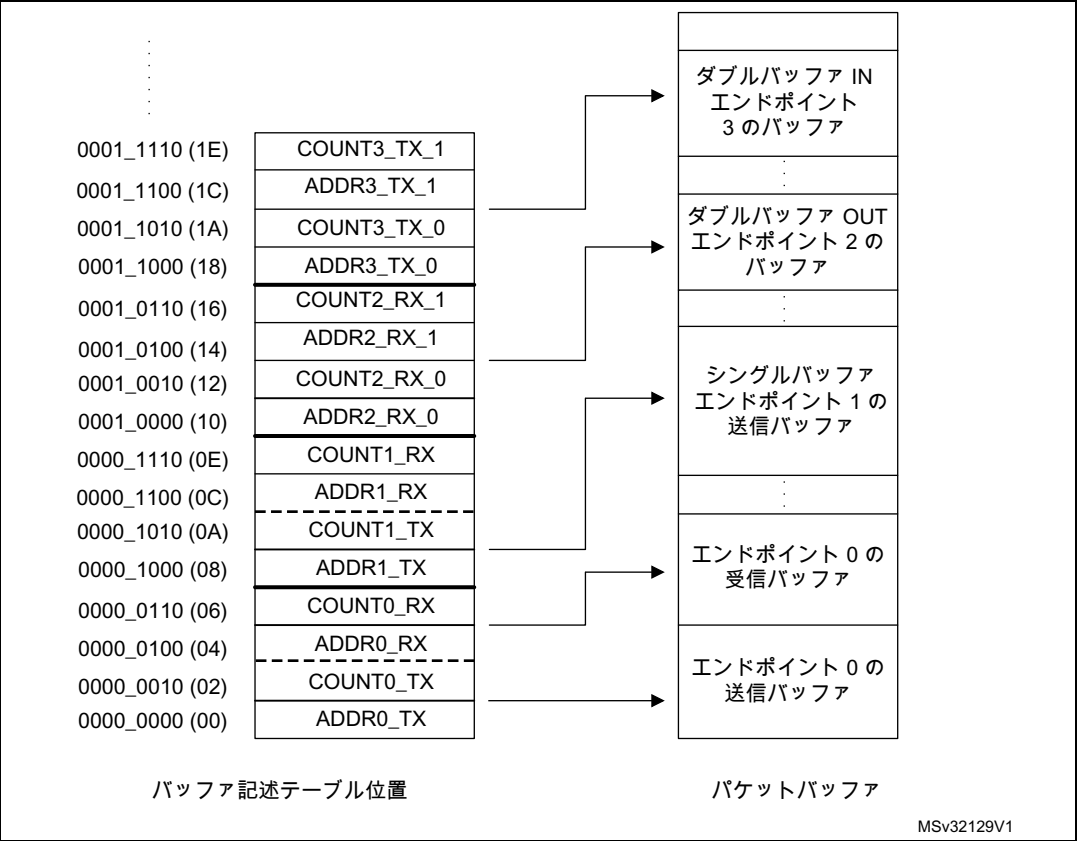
各双方向エンドポイントは、ホストとの間でデータを送受信できます。受信データは、そのエンドポイント用に予約された専用のメモリバッファに格納され、別のメモリバッファにはエンドポイントによって送信されるデータが格納されます。このメモリへのアクセスは、パケットバッファインタフェースブロックによって行われ、これはメモリアクセスリクエストを配信し、確認応答を待ちます。マイクロコントローラもパケットバッファメモリにアクセスする必要があるため、アクセスコンフリクトを避けるためにアービトレーションロジックが使用され、APB サイクルの半分がマイクロコントローラのアクセスに使用され、残りの半分が USB ペリフェラルのアクセスに使用されます。このように、パケットメモリがデュアルポート SRAM であるかのように両方のエージェントが動作でき、マイクロコントローラが連続アクセスを実行中でも、コンフリクトを気にする必要がありません。USB ペリフェラルロジックは、専用クロックを使用します。この専用クロックの周波数は、USB 標準の要件により 48 MHz に固定されており、これは APB バスへのインタフェースに使用されるクロックと異なる場合があります。APB クロック周波数が USB ペリフェラルのクロック周波数より高い、または低い場合、さまざまなクロック設定が可能です。

注： **USB データレートとパケットメモリインタフェース要件により、APB クロックは、データのオーバーラン/アンダーラン問題を回避するために、少なくとも 10 MHz 以上の周波数でなければなりません。**

各エンドポイントには 2 つのパケットバッファが関連付けられます（通常は、送信用に 1 つ、受信用に 1 つ）。バッファの位置とサイズはバッファ記述テーブルで指定されるので、バッファはパケットメモリ内の任意の場所に配置でき、このテーブルも USB_BTABLE レジスタによって示されるパケットメモリ内のアドレスに配置されます。各テーブルエントリがエンドポイントレジスタに関連付けら

れ、4 つの 16 ビットハードワードで構成されるため、テーブル開始アドレスは常に 8 バイトの境界に揃えられます (USB_BTABLE レジスタの下位 3 ビットは常に 000)。バッファディスクリプタテーブルのエントリについては、[セクション 32.6.3 : バッファディスクリプタテーブル](#)で説明します。エンドポイントが単方向であり、アイソクロナスでもダブルバッファバルクでもない場合、必要なパケットバッファは 1 つだけです (サポートされる転送方向に応じた 1 つのバッファ)。サポートされない転送方向または未使用のエンドポイントに関連するテーブル位置は、ユーザが使用できます。アイソクロナスおよびダブルバッファバルクエンドポイントには、パケットバッファの特別な取り扱いがあります (それぞれ、[セクション 32.5.4 : アイソクロナス転送](#)および [セクション 32.5.3 : ダブルバッファエンドポイント](#)を参照)。バッファ記述テーブルのエントリとパケットバッファ領域の関係を [図 300](#)に示します。

図 300. パケットバッファ領域とバッファ記述テーブルの位置の例



各パケットバッファは、始めから終わりまで、受信または送信のいずれかに使用されます。USB ペリフェラルは、割り当てられたメモリバッファに隣接するメモリ位置の内容を変えることはありません。割り当てられたバッファ長を超えるパケットが受信された場合 (バッファオーバーラン条件)、使用可能な最期の位置までしか、データはメモリにコピーされません。

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

エンドポイントの初期化

エンドポイントを初期化するための最初のステップは、ADDRn_TX/ADDRn_RX レジスタに適切な値を書き込んで、送信データがすでに使用可能になったことと、受信データをバッファできることを USB ペリフェラルに知らせることです。USB_EPnR レジスタの EP_TYPE ビットをエンドポイントのタイプに応じてセットし、最終的には EP_KIND ビットを使用して、特に必要な機能を有効にする必要があります。送信側では、USB_EPnR レジスタの STAT_TX ビットを使用してエンドポイントを有効にする必要があります。受信の場合、STAT_RX ビットをセットして受信を有効にする必要があります。BL_SIZE および NUM_BLOCK フィールドを使用して、割り当てられたバッファサイズを COUNTn_RX に書き込む必要があります。単方向エンドポイントは、アイソクロナスおよびダブルバッファバルクエンドポイントを除き、サポートされる方向に関するビットとレジスタのみを初期化する必要があります。送信または受信が有効になると、レジスタ USB_EPnR と位置 ADDRn_TX/ADDRn_RX、COUNTn_TX/COUNTn_RX（それぞれ）をアプリケーションソフトウェアで変更しないでください。これらの値はハードウェアが直接変更できます。データ転送操作が完了し、CTR 割り込みイベントによって通知されると、再びアクセスできるようになり、新しい操作を再び有効にできます。

IN パケット（データ送信）

IN トークンパケットを受信するとき、受信したアドレスが設定済みの有効なエンドポイントと一致した場合、USB ペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内の ADDRn_TX および COUNTn_TX 位置にアクセスします。これらの位置の内容は、内部 16 ビットレジスタ ADDR および COUNT に格納されます（ソフトウェアからはアクセスできません）。パケットメモリが再びアクセスされ、最初の送信バイトを読み出し（[パケットバッファの構造と用途 \(844 ページ\)](#)）を参照）、USB_EPnR のビット DTOG_TX に従って、DATA0 または DATA1 PID の送信を開始します。PID が完了すると、バッファメモリから読み出された最初のバイトが出力シフトレジスタにロードされ、USB バスで送信されます。最後のデータバイトが送信された後、計算された CRC が送信されます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_EPnR レジスタの STAT_TX ビットに従って、NAK または STALL ハンドシェイクパケットがデータパケットの代わりに送信されます。

ADDR 内部レジスタは、現在のバッファメモリ位置を指すポインタとして使用され、COUNT は、残りの送信バイト数をカウントするために使用されます。パケットバッファメモリから読み出された各ハーフワードは、最下位バイトから順に USB バスで送信されます。送信バッファメモリは、COUNTn_TX/2 ハーフワードの場合、ADDRn_TX によって示されるアドレスから順に読み出されます。送信パケットが奇数個のバイトで構成される場合、最後にアクセスされたハーフワードの下位半分のみが使用されます。

ホストが ACK を受信した場合、USB_EPnR レジスタは次のように更新されます。DTOG_TX ビットがトグルされ、STAT_TX=10 (NAK) をセットすることによってエンドポイントが無効化され、ビット CTR_TX がセットされます。アプリケーションソフトウェアは、まず、USB_ISTR レジスタの EP_ID および DIR ビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。CTR_TX イベントの処理は、割り込みビットをクリアすることから始まります。次に、アプリケーションソフトウェアは、送信データでいっぱい別のバッファを準備し、COUNTn_TX テーブル位置を次の転送時に送信されるバイト数で更新し、最後に、STAT_TX を 11 (VALID) にセットして、送信を再び有効にします。STAT_TX ビットが 10 (NAK) に等しいときには、そのエンドポイントにアドレス指定された IN リクエストは NAK され、フロー制御条件を示します。USB ホストは、成功するまで、トランザクションを再試行します。CTR 割り込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された 2 番目の IN トランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。

OUT および SETUP パケット (データ受信)

この 2 つのトークンは、USB ペリフェラルによっておおむね同じように扱われます。SETUP パケットの取り扱いにおける違いについては、制御転送に関する以下のパラグラフで詳しく述べます。OUT/SETUP PID を受信するとき、アドレスが有効なエンドポイントと一致した場合、USB ペリフェラルは、アドレス指定されたエンドポイントに関連するバッファディスクリプタテーブルエントリ内の ADDRn_RX および COUNTn_RX 位置の内容にアクセスします。ADDRn_RX の内容は、内部レジスタ ADDR に直接格納されます。COUNT がリセットされ、BL_SIZE および NUM_BLOCK ビットフィールドの値が COUNTn_RX 内に読み出されると、それを使用して BUF_COUNT が初期化され、これは、バッファオーバーラン条件のチェックに使用される内部 16 ビットカウンタです (これらの内部レジスタはすべて、ソフトウェアからはアクセスできません)。その後、USB ペリフェラルによって受信されるデータバイトは、ハーフワード単位でパケット化され (最初の受信バイトが最下位バイトとして格納されます)、内部 ADDR レジスタに含まれるアドレスから順にパケットバッファに転送され、各バイト転送により、BUF_COUNT はデクリメントされ、COUNT はインクリメントされます。DATA パケットの終わりが検出されると、受信された CRC の正しさがテストされ、受信時にエラーが発生しなかった場合のみ、ACK ハンドシェイクパケットが送信元ホストに送り返されます。

CRC が正しくなかった場合、または他の種類のエラー (ビットスタフ違反、フレームエラーなど) が発生した場合、データバイトは、少なくともエラー検出ポイントまではパケットメモリにコピーされますが、ACK パケットは送信されず、USB_ISTR レジスタの ERR ビットがセットされます。ただし、この場合、通常はソフトウェアのアクションは不要です。USB ペリフェラルは受信エラーから回復し、次のトランザクションを処理する準備ができた状態を保ちます。アドレス指定されたエンドポイントが有効でなかった場合は、USB_EPnR レジスタの STAT_RX ビットに従って、NAK または STALL ハンドシェイクパケットがデータパケットの代わりに送信され、データは受信メモリバッファに書き込まれません。

受信メモリバッファの位置は、ADDRn_RX に含まれるアドレスから、CRC を含めた受信データパケット長に対応するバイト数 (データペイロード長 + 2) だけ、または、BL_SIZE および NUM_BLOCK によって定義された割り当て済みの最後のメモリ位置のいずれか先に来る方まで書き込まれます。このように、USB ペリフェラルは、割り当てられた受信メモリバッファ領域の終わりを越えて書き込むことはありません。データパケットペイロードの長さ (アプリケーションによって使用された実際のバイト数) が割り当てられたバッファより大きい場合、USB ペリフェラルはバッファオーバーラン条件を検出します。この場合、通常の ACK の代わりに STALL ハンドシェイクが送信されて、問題をホストに通知し、割り込みは生成されず、トランザクションは失敗したとみなされます。

ACK ハンドシェイクパケットを送信することにより、トランザクションが正常に完了すると、内部 COUNT レジスタがバッファ記述テーブルエントリ内の COUNTn_RX 位置にコピーされ、影響を受けなかった BL_SIZE および NUM_BLOCK フィールドはそのままに保たれ (通常は再書き込みの必要はありません)、USB_EPnR レジスタが次のように更新されます。DTOG_RX ビットがトグルされ、STAT_RX=10 (NAK) をセットすることによってエンドポイントが無効化され、ビット CTR_RX がセットされます。エラーまたはバッファオーバーランによってトランザクションが失敗した場合、上記のアクションはいずれも実行されません。アプリケーションソフトウェアは、まず、USB_ISTR レジスタの EP_ID および DIR ビットを調べることによって、マイクロコントローラの処理を要求しているエンドポイントを識別する必要があります。CTR_RX イベントの処理は、まず、トランザクションのタイプを決定することによって行われます (USB_EPnR レジスタの SETUP ビット)。アプリケーションソフトウェアは、割り込みフラグビットをクリアして、処理されるエンドポイントに関連するバッファ記述テーブルエントリ内の COUNTn_RX 位置を読み出して、受信バイト数を取得する必要があります。受信データが処理された後、アプリケーションは USB_EPnR の STAT_RX ビットを 11 (有効) にセットして、次のトランザクションを有効にする必要があります。STAT_RX ビットが 10 (NAK) に等しいときには、そのエンドポイントにアドレス指定された OUT リクエストは NAK され、フロー制御条件を示します。USB ホストは、成功するまで、トランザクションを再試行します。CTR 割り込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された 2 番目の OUT トランザクションの通知が失われるのを避けるためには、一連の操作を上記の順序で実行する必要があります。



制御転送

制御転送は、SETUP トランザクション、すべて同じ方向のゼロ個以上のデータステージ、およびステータスステージ（反対方向のゼロバイトの転送）で構成されます。SETUP トランザクションは制御エンドポイントのみによって処理され、OUT トランザクション（データ受信）と非常に似ていますが、制御転送を初期化するために、アドレス指定されたエンドポイントレジスタの DTOG_TX および DTOG_RX ビットの値がそれぞれ 1 と 0 にセットされる点と、ソフトウェアが後続のトランザクションが IN か OUT かを SETUP の内容に応じて決定できるように、STAT_TX と STAT_RX の両方が 10 (NAK) にセットされる点が異なります。制御エンドポイントは CTR_RX イベントごとに USB_EPnR レジスタの SETUP ビットをチェックして、通常の OUT トランザクションと SETIP トランザクションを区別する必要があります。USB デバイスは、SETUP ステージで転送されたデータを解釈することによって、データステージの数と方向を判断でき、エラーの場合にはトランザクションを STALL する必要があります。そのためには、最後以外のすべてのデータステージで、未使用の方向を STALL にセットして、ホストが転送方向を反転するのが速過ぎた場合にはステータスステージとして STALL されるようにする必要があります。

最後のデータステージを有効にするときには、反対方向を NAK にセットして、ホストが（ステータスステージを実行するために）転送方向をすぐに反転した場合、制御操作の完了を待つようにする必要があります。制御操作が正常に完了した場合、ソフトウェアは NAK を VALID に変更し、そうでない場合は STALL に変更します。同時に、ステータスステージが OUT の場合、STATUS_OUT (USB_EPnR レジスタの EP_KIND) ビットをセットして、ステータストランザクションが非ゼロのデータで実行された場合はエラーが生成されるようにする必要があります。ステータストランザクションが処理されると、アプリケーションは STATUS_OUT ビットをクリアし、STAT_RX を VALID にセットし（新しいコマンドを受け入れるため）、STAT_TX を NAK にセットします（次のセットアップの直後のステータスステージを遅延するため）。

USB 仕様では、SETUP パケットには ACK 以外のハンドシェイクでは応答できず、結果として、以前に発行されたコマンドをアボートして新しいコマンドを開始すると規定されているため、USB ロジックでは、制御エンドポイントはホストから受信した SETUP トークンに対して NAK または STALL パケットで応答することはできません。

STAT_RX ビットが 01 (STALL) または 10 (NAK) にセットされ、SETUP トークンが受信されると、USB はデータを受け入れ、要求されたデータ転送を実行し、ACK ハンドシェイクを返送します。そのエンドポイントに以前に発行された CTR_RX リクエストがあり、アプリケーションによってまだ確認応答されていない場合（すなわち、CTR_RX ビットが以前に完了した受信からセットされたままになっている場合）、USB は SETUP トランザクションを破棄し、状態にかかわらずハンドシェイクパケットで応答せず、受信エラーをシミュレートして、ホストが SETIP トークンを再び送信しなければならないようにします。これは、CTR_RX 割り込みをトリガしたトランザクションの直後に、同じエンドポイントにアドレス指定された SETUP トランザクションの通知が失われるのを避けるために行われます。

32.5.3 ダブルバッファエンドポイント

USB 標準によって定義されているすべてのエンドポイントタイプは、さまざまなトラフィックモデルを表し、各種のデータ転送操作の典型的な要件を記述しています。データの大部分をホスト PC と USB 機能の間で転送する必要があるときには、バルクエンドポイントタイプが最適のモデルです。ホストはバルクトランザクションをスケジュールして、フレーム内の使用可能な帯域幅のすべてをいっぱいにして、アドレス指定された USB 機能がバルクトランザクションを処理できる限り、実際の転送速度を最大化します。次のトランザクションが着信したときに、USB 機能が前のトランザクションでビジーであった場合は、NAK ハンドシェイクで応答し、ホスト PC は、USB 機能が処理できるようになるまで、同じトランザクションを再び発行するので、再送信によって帯域幅が占有されるため、実際の転送速度は低下します。このため、バルクエンドポイントでは、「ダブルバッファリング」と呼ばれる専用機能を使用できます。

「ダブルバッファリング」が有効なときには、データトグルシーケンシングが使用されて、USB ペリフェラルが要求されたデータ転送を実行するために使用するバッファが選択され、送信と受信の両方のパケットメモリ領域を使用して、トランザクションの成功ごとのバッファスワッピングを管理するため、アプリケーションは常に完全なバッファを使用でき、USB ペリフェラルは他方のバッファをいっぱいにします。たとえば、「受信」ダブルバッファバルクエンドポイント宛ての OUT トランザクション時、1 つのバッファが USB ホストから着信する新しいデータで満たされている間、もう 1 つのバッファはマイクロコントローラソフトウェアが使用できます（「送信」ダブルバッファバルクエンドポイントと IN トランザクションでも同様です）。

スワップバッファ管理には、アドレスポイントと割り当て済みメモリバッファの長さをホストする 4 つすべてのバッファ記述テーブル位置を使用する必要があるため、ダブルバッファバルクエンドポイントの実装に使用される USB_EPnR レジスタは、強制的に単方向として使用されます。したがって、1 つの STAT ビットのみを 00（無効）以外の値にセットする必要があります。ダブルバッファバルクエンドポイントが受信について有効な場合は STAT_RX、ダブルバッファバルクエンドポイントが送信について有効な場合は STAT_TX。ダブルバッファバルクエンドポイントを受信と送信の両方について有効にする必要がある場合は、2 つの USB_EPnR レジスタを使用する必要があります。

ダブルバッファリング機能を利用して、可能な限り高い転送速度を達成するには、前の章で説明したエンドポイントフロー制御構造を変更して、トランザクションの正常官僚の旅ではなく、USB ペリフェラルとアプリケーションの間でバッファコンフリクトが発生したときのみ、エンドポイントのステータスを NAK に切り替える必要があります。USB ペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関連する DTOG ビットによって定義されます。すなわち、「受信」ダブルバッファバルクエンドポイントの場合は DTOG_RX（USB_EPnR レジスタのビット 14）、または「送信」ダブルバッファバルクエンドポイントの場合は DTOG_TX（USB_EPnR レジスタのビット 6）です。新しいフロー制御方式を実装するには、USB ペリフェラルはアプリケーションソフトウェアによって使用されているパケットバッファを知り、コンフリクトに注意する必要があります。USB_EPnR レジスタには 2 つの DTOG ビットがありますが、データおよびバッファシーケンシングのために USB ペリフェラルによって使用されるのは 1 つだけであり（ダブルバッファリング機能に必要な単方向成約のため）、もう 1 つはアプリケーションソフトウェアが、使用中のバッファを示すために使用できます。この新しいバッファフラグは、SW_BUF と呼ばれます。次の表で、「送信」および「受信」ダブルバッファバルクエンドポイントの場合の USB_EPnR レジスタのビットと DTOG/SW_BUF 定義の対応を説明します。

表 133. ダブルバッファリングバッファフラグの定義

| バッファフラグ | 「送信」エンドポイント | 「受信」エンドポイント |
|---------|-------------------------|--------------------------|
| DTOG | DTOG_TX（USB_EPnR ビット 6） | DTOG_RX（USB_EPnR ビット 14） |
| SW_BUF | USB_EPnR ビット 14 | USB_EPnR ビット 6 |

USB ペリフェラルによって現在使用されているメモリバッファは、DTOG バッファフラグによって定義され、アプリケーションソフトウェアによって現在使用されているバッファは、SW_BUF バッファフラグによって示されます。バッファフラグの値と使用されているパケットバッファの関係は、どちらの場合も同じであり、次の表に示されています。

表 134. バルクダブルバッファリングメモリバッファの用途

| エンド ポイント タイプ | DTOG | SW_BUF | USB ペリフェラルによって 使用されるパケットバッファ | アプリケーションソフトウェアに よって使用されるパケットバッファ |
|--------------------|------|--------|---|---|
| IN | 0 | 1 | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 | ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置。 |
| | 1 | 0 | ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置 | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 |
| | 0 | 0 | なし ⁽¹⁾ | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 |
| | 1 | 1 | なし ⁽¹⁾ | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 |
| OUT | 0 | 1 | ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。 | ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。 |
| | 1 | 0 | ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。 | ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。 |
| | 0 | 0 | なし ⁽¹⁾ | ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。 |
| | 1 | 1 | なし ⁽¹⁾ | ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。 |

1. NAK ステータスのエンドポイント。

バルクエンドポイントのダブルバッファリング機能は、次のようにして有効化されます。

- USB_EPnR レジスタの EP_TYPE ビットフィールドに 00 を書き込み、エンドポイントをバルクとして定義します。
- 同じレジスタの EP_KIND ビットを 1 (DBL_BUF) にセットします。

アプリケーションソフトウェアは、最初に使用されるバッファに応じて、DTOG および SW_BUF ビットを初期化する責任があります。これは、この 2 つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。DBL_BUF のセット後に発生した初めてのトランザクションの終了により、ダブルバッファバルクエンドポイントの特殊なフロー制御がトリガされ、DBL_BUF がセットされたままである限り、このエンドポイントにアドレス指定された他のすべてのトランザクションについて、これが使用されます。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイント USB_EPnR レジスタの CTR_RX または CTR_TX ビットがセットされます。同時に、USB_EPnR レジスタの影響を受けた DTOG ビットがハードウェアによってトグルされ、USB ペリフェラルのバッファスワッピングを完全にソフトウェアから独立させます。通常のトランザクション、および DBL_BUF のセット後の初めてのトランザクションと違って、STAT ビットペアはトランザクション終了の影響を受けず、値は 11 (有効) のままです。ただし、新しいトランザクションのトークンパケットが受信されると、実際のエンドポイントステータスは、USB ペリフェラルとアプリケーションソフトウェアの間のバッファコンフリクトが検出されたときには 10 (NAK) としてマスクされます (この条件は DTOG と SW_BUF が同じ値を持つことによって示されます。表 134 (850 ページ) を参照)。アプリケーションソフトウェアは、CTR イベント通知に応答して、割り込みフラグをクリアし、完了したトランザクションの要求された処理を開始します。アプリケーションパケットバッファの仕様が終了すると、ソフトウェアは SW_BUF ビットに 1 を書き込んでトグルし、そのバッファが使用可能になったことを USB ペリフェラルに通知します。このように、NAK されるトランザクションの数は、トランザクションデータのアプリケーション得ラボレーション時間によってのみ制限されます。得ラボレーション時間が USB バスでのトランザクションの完了に必要な時間

より短かった場合、フロー制御による再送信は行われず、実際の転送速度はホスト PC によってのみ制限されます。

アプリケーションソフトウェアは、11 (有効) とは異なる明示的なステータスを、関連する USB_EPnR レジスタの STAT ビットペアに書き込むことによって、ダブルバッファバルクエンドポイントのために実装される特殊なフロー制御をオーバーライドできます。この場合、USB ペリフェラルは、バッファ使用条件に関係なく、プログラムされたエンドポイントステータスを常に使用します。

32.5.4 アイソクロナス転送

USB 標準は、固定された正確なデータ生成/消費周波数を必要とするフルスピードペリフェラルをサポートし、この種類のトラフィックを「アイソクロナス」として定義しています。このデータの典型的な例は、オーディオサンプル、圧縮ビデオストリーム、および一般に、供給される周波数の正確さについて厳密な要件を持つ、あらゆる種類のサンプルデータです。エニュメレーションフェーズでエンドポイントが「アイソクロナス」として定義されると、ホストは必要な帯域幅をフレームに割り当て、エンドポイントの方向に応じて、各フレームに正確に 1 つの IN または OUT パケットを供給します。帯域幅要件を制限するために、アイソクロナストラフィックについては、失敗したトランザクションの再送信はできません。このため、アイソクロナストランザクションにはハンドシェイクフェーズはなく、ACK パケットは予期されず、データパケット後に送信されません。同じ理由で、アイソクロナス転送はデータグルーピングをサポートせず、常に DATA0 PID を使用してデータパケットを開始します。

エンドポイントのアイソクロナス動作は、USB_EPnR レジスタの EP_TYPE ビットを 10 にセットすることによって選択されます。ハンドシェイクフェーズがないため、STAT_RX/STAT_TX ビットの唯一の正当な値は 00 (無効) と 11 (有効) であり、他の値の場合は、USB 標準に準拠しない結果になります。アイソクロナスエンドポイントはダブルバッファリングを実装して、アプリケーションソフトウェア開発を容易にし、「送信」と「受信」の両方のパケットメモリ領域を使用して、トランザクションの成功のたびにバッファスワッピングを管理し、アプリケーションは常に完全なバッファを使用でき、USB ペリフェラルはもう 1 つのバッファを満たします。

USB ペリフェラルによって現在使用されているメモリバッファは、エンドポイント方向に関する DTOG ビットによって、表 135 に従って定義されます (「受信」アイソクロナスエンドポイントの場合は DTOG_RX、「送信」アイソクロナスエンドポイントの場合は DTOG_TX であり、両方とも、関連する USB_EPnR レジスタ)。

表 135. アイソクロナスメモリバッファの使用

| エンドポイント タイプ | DTOG ビットの値 | USB ペリフェラルによって 使用されるパケットバッファ | アプリケーションソフトウェアによって 使用されるパケットバッファ |
|----------------|---------------|---|---|
| IN | 0 | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 | ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置。 |
| | 1 | ADDRn_TX_1 / COUNTn_TX_1 バッファ記述テーブル位置。 | ADDRn_TX_0 / COUNTn_TX_0 バッファ記述テーブル位置。 |
| OUT | 0 | ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。 | ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。 |
| | 1 | ADDRn_RX_1 / COUNTn_RX_1 バッファ記述テーブル位置。 | ADDRn_RX_0 / COUNTn_RX_0 バッファ記述テーブル位置。 |

ダブルバッファアイソクロナスエンドポイントで発生すると、アイソクロナスエンドポイントの実装に使用される USB_EPnR レジスタは強制的に単方向として使用されます。アイソクロナスエンドポイントを受信と送信の両方について有効にする必要がある場合は、2 つの USB_EPnR レジスタを使用する必要があります。

アプリケーションソフトウェアは、最初に使用されるバッファに応じて、DTOG ビットを初期化する責任があります。これは、この 2 つのビットが持つ特殊なトグル専用プロパティを考慮して行う必要があります。各トランザクションの終了時、有効化された方向に応じて、アドレス指定されたエンドポイント USB_EPnR レジスタの CTR_RX または CTR_TX ビットがセットされます。同時に、USB_EPnR レジスタの影響を受けた DTOG ビットがハードウェアによってトグルされ、バッファスワッピングを完全にソフトウェアから独立させます。STAT ビットペアはトランザクションの完了に影響を受けません。アイソクロナス転送ではハンドシェイクフェーズがないため、フロー制御は不可能であり、エンドポイントは常に 11 (有効) のままです。CRC エラーまたはバッファオーバーラン条件がアイソクロナス OUT 転送時に発生した場合、常に正しいトランザクションとみなされ、常に CTR_RX イベントをトリガします。ただし、CRC エラーが発生すると、常に USB_ISTR レジスタの ERR ビットがセットされて、データ破損の可能性をソフトウェアに通知します。

32.5.5 サスペンド／レジュームイベント

USB 標準では、SUSPEND と呼ばれる特殊なペリフェラル状態が定義され、USB バスからの平均電流は 2.5 mA より大きくなってはなりません。この要件はバスパワードデバイスにとって基本的な重要性を持ちますが、セルフパワードデバイスは、この厳格な消費電力成約を順守する必要はありません。サスペンドモードでは、ホスト PC は 3 ms を超えて USB バスにトラフィックを送信しないことによって通知を送信します。SOF パケットは、通常動作では 1 ms ごとに送信されなければならないので、USB ペリフェラルは 3 回連続の SOF パケットの欠落をホスト PC からのサスペンドリクエストとして検出し、USB_ISTR レジスタの SUSP ビットを 1 にセットし、有効な場合は割り込みが生成されます。デバイスがサスペンドされると、いわゆる RESUME シーケンスによって通常操作が復元され、これはホスト PC から、またはペリフェラルそのものから直接開始できますが、常にホスト PC によって終了されます。サスペンドされた USB ペリフェラルは、常に RESET シーケンスを検出して、通常の USB リセットイベントとして、このイベントに対応できなければなりません。

USB ペリフェラルをサスペンドするために使用される実際の手順は、デバイスの構成によって合計消費量を削減するために必要なアクションが異なるため、デバイスに依存します。

USB ペリフェラルの SUSP 通知にตอบสนองするアプリケーションソフトウェアルーチンの USB 関連の局面を中心にして、典型的なサスペンド手順の簡単な説明を以下に示します。

1. USB_CNTR レジスタの FSUSP ビットを 1 にセットします。このアクションによって、USB ペリフェラル内のサスペンドモードが有効になります。サスペンドモードが有効になると、USB がサスペンドされている間に SUSP 割り込みが発行されるのを回避するために、SOF 受信時のチェックが無効になります。
2. USB ペリフェラルとは異なるブロックの静的消費電力を除去または削減します。
3. USB_CNTR レジスタの LP_MODE ビットを 1 にセットして、アナログ USB トランシーバの静的消費電力を除去しつつ、レジュームアクティビティを検出できるままにしておきます。
4. オプションで、外部オシレータとデバイス PLL をオフにして、デバイス内部のアクティビティを停止します。

デバイスが SUSPEND モード中に USB イベントが発生したときには、RESUME 手順が呼び出されて、通常のチェックと復元し、通常の USB 動作を復元する必要があります。ウェイクイベントが USB リセットシーケンスである時には、このプロセスが 10 ms 以内で終了するように特に注意が必要です (詳細については、「ユニバーサルシリアルバス仕様」を参照してください)。USB ペリフェラルのサスペンド中にレジュームまたはリセットシーケンスが開始されると、USB_CNTR レジスタの LP_MODE ビットが非同期でクリアされます。このイベントによって WKUP 割り込みがトリガされた場合でも、システムクロックの再開による長い遅延のため、割り込み応答ルーチンの使用は慎重に評価する必要があります。公称クロックの再有効化の前の遅延を短くするには、サスペンド手順の終了直後にレジューム手順を行って、システムクロックが再開したらすぐにコードが実行されるようにします。システムをウェイクアップする際の ESD 放電またはその他の種類のノイズを防止するために (サスペンドモードの終了は、非同期イベントです)、サスペンド中にデータラインステータスの適切なアナログフィルタを有効にします。フィルタ幅は約 70 ns です。

以下は、レジューム手順で行うべきアクションのリストです。

1. オプションで、外部オシレータおよびデバイス PLL をオンにします。
2. USB_CNTR レジスタの FSUSP ビットをクリアします。
3. レジュームをトリガしたイベントを識別する必要がある場合は、USB_FNR レジスタのビット RXDP および RXDM を [表 136](#) に従って使用でき、この表には、あらゆる場合のソフトウェアのアクションもリストされています。必要な場合には、上記のビットのステータスを監視することによって、レジュームまたはリセットシーケンスの終了を検出でき、これらが「10」設定 (アイドルバス状態を表します) に達したことをチェックします。さらに、リセットシーケンスの終了時には、USB_ISTR レジスタの RESET ビットが 1 にセットされ、有効な場合は割り込みが発行され、通常通りに処理されます。

表 136. レジュームイベント検出

| [RXDP,RXDM] ステータス | ウェイクアップイベント | 必要なレジュームソフトウェアアクション |
|-------------------|--------------|---------------------|
| "00" | ルートのリセット | なし |
| "10" | なし (バス上のノイズ) | サスペンドモードに戻る |
| "01" | ルートのレジューム | なし |
| "11" | 不可 (バス上のノイズ) | サスペンドモードに戻る |

デバイスは、USB プロトコルに直接関係しない特定のイベントに対する応答として、サスペンドモード化を終了しなければならない場合があります (たとえば、マウスの移動はシステム全体をウェイクアップします)。この場合、USB_CNTR レジスタの RESUME ビットを 1 にセットし、1 ms から 15 ms までのインターバル後に 0 にリセットすることによって、レジュームシーケンスを開始できます (このインターバルは、システムクロックが公称周波数で動作しているときには 1 ms 周期で発生する ESOF 割り込みによって計測できます)。RESUME ビットがクリアされると、レジュームシーケンスはホスト PC によって完了され、その終了は、USB_FNR レジスタの RXDP および RXDM ビットを使用して監視できます。

注: RESUME ビットは、USB ペリフェラルがサスペンドモードにされた (USB_CNTR レジスタの FSUSP ビットを 1 にセットした) 後でのみ使用する必要があります。



32.6 USBレジスタ

USB ペリフェラルレジスタは、以下のグループに分けることができます。

- 共通レジスタ：割り込みおよび制御レジスタ。
- エンドポイントレジスタ：エンドポイント設定およびステータス
- バッファディスクリプタテーブル：データバッファの配置に使用されたパケットメモリの位置

すべてのレジスタアドレスは、USB ペリフェラルレジスタのベースアドレス 0x4000 5C00 に対するオフセットとして表されますが、バッファディスクリプタテーブル位置は USB_BTABLE レジスタによって指定されたアドレスから始まります。パケットバッファメモリ位置は、0x4000 6000 から順に配置されます。

レジスタの説明で使用されている略語のリストについては、[セクション 1.1 \(38 ページ\)](#) を参照してください。

ペリフェラルレジスタには、ハーフワード（16 ビット）またはワード（32 ビット）単位でアクセスする必要があります。

32.6.1 共通レジスタ

これらのレジスタは USB ペリフェラルの全般的動作に影響し、動作モード、割り込み処理、デバイスアドレス、およびホスト PC によって更新される現在のフレーム番号へのアクセス付与を定義します。

USB 制御レジスタ (USB_CNTR)

アドレスオフセット：0x40

リセット値：0x0003

ビット 15 **CTRM**：正しい転送割り込みマスク

- 0：正しい転送 (CTR) 割り込みは無効です。
- 1：CTR 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 14 **PMAOVRM**：パケットメモリ領域のオーバー/アンダーラン割り込みマスク

- 0：PMAOVR 割り込みは無効です。
- 1：PMAOVR 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 13 **ERRM**：エラー割り込みマスク

- 0：ERR 割り込みは無効です。
- 1：ERR 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 12 **WKUPM**：ウェイクアップ割り込みマスク

- 0：WKUP 割り込みは無効です。
- 1：WKUP 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 11 **SUSPM**：サスペンドモード割り込みマスク

- 0：サスペンドモードリクエスト (SUSP) 割り込みは無効です。
- 1：SUSP 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 10 **RESETM**：USB リセット割り込みマスク

- 0：RESET 割り込みは無効です。
- 1：RESET 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 9 **SOFM** : フレーム開始割り込みマスク

0 : SOF 割り込みは無効です。

1 : SOF 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 8 **ESOFM** : 予期されるフレーム開始割り込みマスク

0 : 予期されるフレーム開始 (ESOF) 割り込みは無効です。

1 : ESOF 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

ビット 7 **L1REQM** : LPM L1 状態リクエスト割り込みマスク

0 : LPM L1 状態リクエスト (L1REQ) 割り込みマスク無効

1 : L1REQ 割り込みは有効であり、USB_ISTR レジスタの対応するビットがセットされたときには割り込みリクエストが生成されます。

注 :

ビット 6 予約済み

ビット 5 **L1RESUME** : LPM L1 レジュームリクエスト

マイクロコントローラはこのビットをセットして、ホストに LPM L1 レジューム信号を送信できます。

注 :

ビット 4 **RESUME** : レジュームリクエスト

マイクロコントローラはこのビットをセットして、ホストにレジューム信号を送信できます。USB 仕様に従って、ホスト PC がレジュームシーケンスを終了まで駆動する準備ができてから 1 ms 以上 15 ms 以下で有効にする必要があります。

ビット 3 **FSUSP** : 強制サスペンド

ソフトウェアは、SUSP 割り込みが受信されたときにこのビットをセットする必要があるため、USB ペリフェラルが 3 ms 間、トラフィックを受信しなかったときに発行されます。

0 : 影響なし。

1 : サスペンドモードに入ります。アナログトランシーバのクロックと静的消費電流は影響を受けません。サスペンド消費電力が要件の場合 (バスパスワードデバイス)、アプリケーションソフトウェアは FSUSP の後、以下の説明に従って LP_MODE ビットをセットする必要があります。

ビット 2 **LP_MODE** : 低電力モード

このモードは、サスペンドモードの電力制約から、外部プルアップレジスタに供給される電力を除き、すべての静的消費電流を回避されるときに使用されます。USB サスペンド条件の消費電力要件を満たすために、アプリケーションがすべてのシステムクロックを停止するか、周波数を削減する準備ができたときに、この条件に入るべきです。サスペンドモード中に USB アクティビティ (WKUP イベント) が発生すると、このビットは非同期でリセットされます (ソフトウェアによってリセットすることもできます)。

0 : 低電力モードではありません。

1 : 低電力モードに入ります。

ビット 1 **PDWN** : パワーダウン

このビットは、何らかの理由で USB ペリフェラルを完全に無効にする必要がある場合、USB 関連のすべてのアナログ部品を完全にオフにするために使用されます。このビットがセットされると、USB ペリフェラルはトランシーバから切断され、使用できなくなります。

0 : パワーダウンを終了します。

1 : パワーダウンモードに入ります。

ビット 0 **FRES** : 強制的 USB リセット

0 : USB リセットをクリアします。

1 : USB に RESET 信号が送られたときと同様に、USB ペリフェラルを強制的にリセットします。ソフトウェアがこのビットをクリアするまで、USB ペリフェラルは RESET 状態に保たれます。有効な場合は、「USB-RESET」割り込みが生成されます。

USB 割り込みステータスレジスタ (USB_ISTR)

アドレスオフセット : 0x44

リセット値 : 0x0000 0000

このレジスタはすべての割り込みソースのステータスを含むため、アプリケーションソフトウェアは割り込みリクエストの原因になったイベントを判断できます。

このレジスタの上位部分はシングルビットを含み、そのそれぞれが特定のイベントを表します。これらのビットは、関連するイベントが発生したときにハードウェアによってセットされます。USB_CNTR レジスタの対応するビットがセットされている場合、汎用割り込みリクエストが生成されます。各ビットを調べる割り込みルーチンは、必要なすべてのアクションを実行し、最後に処理済みビットをクリアします。そのいずれかがクリアされていない場合、割り込みはまだ保留中であるとみなされ、割り込みラインは再びハイに保たれます。複数のビットが同時にセットされた場合、1 つの割り込みだけが生成されます。

エンドポイントトランザクションの完了は、割り込み応答遅延を軽減するために、さまざまな方法で処理できます。CTR ビットは、エンドポイントがトランザクションを正常に完了するとすぐにハードウェアによってセットされ、USB_CNTR の対応するビットがセットされていた場合は汎用割り込みリクエストが生成されます。エンドポイント専用割り込み条件は、USB_CNTR レジスタの CTRM ビットとは無関係に有効化されます。どちらの割り込み条件も、ソフトウェアが対応する USB_EPnR レジスタの保留中のビットをクリアするまでアクティブなままです (CTR ビットは実際には読み出し専用ビットです)。エンドポイント関連の割り込みの場合、ソフトウェアは、トランザクション方向 (DIR) および EP_ID 読み出し専用ビットを使用して、最後の割り込みリクエストを発行して、対応する割り込みサービスルーチンを呼び出したエンドポイントを特定できます。

ユーザは、ソフトウェアが割り込みサービスルーチンで USB_ISTR ビットをチェックする順序を指定することによって、同時に保留中の USB_ISTR ビットの相対的優先順位を選ぶことができます。イベントに関連し、処理済みのビットだけがクリアされます。サービスルーチンの終了時、残りの条件を処理するために、別の割り込みがリクエストされます。

ビットの誤ったクリアを避けるために、ロード命令付きでクリアすることが推奨され、その場合、変更してはならないすべてのビットに 1 が書き込まれ、クリアすべきすべてのビットに 0 が書き込まれます (これらのビットはソフトウェアによってのみクリアできます)。読み出し-変更-書き込みのサイクルは避けるべきです。読み出し操作と書き込み操作の間に、別のビットがハードウェアによってセットされる可能性があり、マイクロプロセッサにイベントを処理する時間ができる前に、次の書き込みによってクリアされる可能性があるためです。

以下に、各ビットについて詳しく説明します。

ビット 15 CTR : 正しい転送

このビットは、エンドポイントがトランザクションを正常に完了したことを示すために、ハードウェアによってセットされます。DIR および EP_ID ビットを使用して、ソフトウェアは割り込みをリクエストしたエンドポイントを判断できます。このビットは読み出し専用です。

ビット 14 PMAOVR : パケットメモリ領域のオーバー／アンダーラン

このビットは、マイクロコントローラに USB メモリリクエストに対応する時間がなくなった場合にセットされます。USB ペリフェラルは、このイベントを次のように処理します。受信時、ACK ハンドシェイクパケットは送信されず、送信時、送信ストリームのビットスタッフエラーが強制的に発行されます。いずれの場合も、ホストはトランザクションを再試行します。通常操作では、PMAOVR 割り込みは発生しません。失敗したトランザクションはホストによって再試行されるので、アプリケーションソフトウェアには、この割り込み処理中にデバイス操作をスピードアップして、次のトランザクションの差異施行に備える機会があります。ただし、アイソクロナス転送では（アイソクロナストランザクションは再試行されません）、データの喪失につながるため、これは行われません。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 13 ERR : 誤差

このフラグは、下記にリストされているエラーのいずれかが発生したときにセットされます。
NANS : 応答なし (No Answer)。ホスト応答がタイムアウトになりました。
CRC : 巡回冗長検査 (Cyclic Redundancy Check) エラー。トークンまたはデータで受信された CRC の 1 つが正しくありませんでした。
BST : ビットスタッフエラー。ビットスタッフエラーが、PID、データ、または CRC で検出されました。
FVIO : フレーミングフォーマット違反。非標準のフレームが受信されました（正しい位置にない EOP、正しくないトークンシーケンスなど）。
USB ペリフェラルと PC ホストは、エラーの場合、完全にトランスペアレントな方法で再送信を管理するため、USB ソフトウェアは、通常、エラーを無視できます。この割り込みは、ソフトウェア開発フェーズで、または USB バスでの送信品質の監視に役立ち、問題の可能性をユーザに知らせることができます（コネクタの緩み、ノイズが多い環境、USB ケーブルの導体の破損など）。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 12 WKUP : ウェイクアップ

このビットは、サスペンドモード中に、USB ペリフェラルをウェイクアップするアクティビティが検出されたときにハードウェアによって 1 にセットされます。このイベントは CTRL レジスタの LP_MODE ビットを非同期でクリアし、USB_WAKEUP ラインを有効にし、レジュームプロセスの開始に関して残りのデバイス（ウェイクアップユニットなど）に通知するために使用されます。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 11 SUSP : サスペンドモードリクエスト

このビットは、3 ms にわたってトラフィックが受信されず、USB バスからのサスペンドモードリクエストを示したときに、ハードウェアによってセットされます。サスペンド条件チェックは、USB リセットの直後に有効化され、サスペンドモードがアクティブなとき (FSUSP=1)、レジュームシーケンスの終了まで、ハードウェアによって無効にされます。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 10 RESET : USB リセットリクエスト

USB ペリフェラルが入力でアクティブな USB RESET 信号を検出したときにセットされます。USB ペリフェラルは、RESET に対応して、内部プロトコル状態マシンをリセットし、USB_CNTR レジスタの RESETM 有効ビットがセットされている場合、割り込みが生成されます。RESET ビットがクリアされるまで、受信と送信は無効です。すべての設定レジスタがリセットされるわけではありません。マイクロコントローラはこれらのレジスタを明示的にクリアする必要があります（これは、RESET 割り込みの安全な配信を確保し、RESET の直前のトランザクションを完了できるようにするためです）。機能アドレスおよびエンドポイントレジスタは、USB リセットイベントによってリセットされます。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 9 SOF : フレーム開始

このビットは、新しい USB フレームの開始を知らせ、SOF パケットが USB バス経由で着信したときにセットされます。割り込みサービスルーチンによって SOF イベントを監視して、USB ホストへの 1 ms の同期イベントを実行し、SOF パケット受信時に更新される USB_FNR レジスタを安全に読み出すことができます（これはアイソクロナスアプリケーションに便利です）。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 8 ESOF : 予期されたフレーム開始

このビットは、予期された SOF パケットが受信されなかったときにハードウェアによってセットされます。ホストは 1 ms ごとに SOF パケットを送信しますが、ハブが正しく受信しなかった場合、サスペンドタイマがこの割り込みを発行します。3 回連続して ESOF 割り込みが生成され（すなわち、3 つの SOF パケットが失われ）、その間にトラフィックが発生しなかった場合、SUSP 割り込みが生成されます。このビットは、サスペンドタイマがまだロックされていないときに SOF パケットの喪失が発生したときにセットされます。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

ビット 7 L1REQ : LPM L1 状態リクエスト

このビットは、L1 状態を入力する LPM コマンドが正常に受信され、確認応答された場合に、ハードウェアによってセットされます。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んでも効果はありません。

注 :

ビット 6:5 予約済み**ビット 4 DIR** : トランザクションの方向

このビットは、割り込みリクエストを生成した成功したトランザクションの方向に従って、ハードウェアによって書き込まれます。

DIR ビット = 0 の場合、割り込みをかけたエンドポイントに関連する USB_EPnR レジスタの CTR_TX ビットがセットされます。割り込みの原因となったトランザクションは、IN タイプです (USB ペリフェラルからホスト PC へのデータ送信)。

DIR ビット = 1 の場合、割り込みをかけたエンドポイントに関連する USB_EPnR レジスタの CTR_RX ビットまたは CTR_TX と CTR_RX の両方がセットされます。割り込みの原因となったトランザクションは OUT タイプであるか (ホスト PC からのデータを USB ペリフェラルが受信)、または 2 つの保留中のトランザクションが処理を待っています。

これは割り込みを保留中にした方向を表すので、アプリケーションソフトウェアはこの情報を使用して、トリガしたトランザクションに関連する USB_EPnR ビットにアクセスできます。このビットは読み出し専用です。

ビット 3:0 EP_ID[3:0] : エンドポイント識別子

これらのビットは、割り込みリクエストを生成したエンドポイント番号に従って、ハードウェアによって書き込まれます。複数のエンドポイントトランザクションが保留中の場合、ハードウェアは、次のように定義される最も高い優先順位を持つエンドポイントに関連するエンドポイント識別子を書き込みます。優先順位の順に、2 つのエンドポイントセットが定義されます。アイソクロナスおよびダブルバッファバルクエンドポイントが最初に考慮され、次に、その他のエンドポイントが調べられます。同じセットの複数のエンドポイントが割り込みをリクエストしている場合、最下位リクエストエンドポイントレジスタに従って、USB_ISTR レジスタの EP_ID ビットが割り当てられ、EP0R が最も高い優先順位を持ち、その後 EP1R が続き、以下同様です。アプリケーションソフトウェアは、この優先順位方式に従って、レジスタを各エンドポイントに割り当てて、同時発生したエンドポイントリクエストを適切に順序づけることができます。これらのビットは読み出し専用です。

USB フレーム番号レジスタ (USB_FNR)

アドレスオフセット : 0x48

リセット値 : 0x0XXX (X は未定義)

| | | | | | | | | | | | | | | | |
|------|------|-----|-----------|----|----------|---|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| RXDP | RXDM | LCK | LSOF[1:0] | | FN[10:0] | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

- ビット 15 **RXDP** : 受信データ + ラインステータス
- このビットを使用して、受信データプラスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。
- ビット 14 **RXDM** : 受信データ - ラインステータス
- このビットを使用して、受信データマイナスアップストリームポートデータラインのステータスを観測できます。サスペンド終了ルーチン時に使用して、ウェイクアップイベントの決定を容易にできます。
- ビット 13 **LCK** : ロック
- このビットは、USB リセット条件の終了後、または USB レジュームシーケンスの終了後に、少なくとも 2 つ連続した SOF パケットが受信されたときに、ハードウェアによってセットされます。ロックされると、USB リセットまたは USB サスペンドイベントが発生するまで、フレームタイマはこの状態に保たれます。
- ビット 12:11 **LSOF[1:0]** : SOF の喪失
- これらのビットは、ESOF 割り込みが生成されたときに、ハードウェアによって書き込まれ、失われた連続 SOF パケットの数をカウントします。SOF パケットの受信時、これらのビットはクリアされます。
- ビット 10:0 **FN[10:0]** : フレーム番号
- このビットフィールドは、最後に受信された SOF パケットに含まれる 11 ビットのフレーム番号を含みます。フレーム番号は、ホストによって送信されたフレームごとにインクリメントされ、アイソクロナス転送の場合に便利です。このビットフィールドは、SOF 割り込みの生成時に更新されます。

USB デバイスアドレス (USB_DADDR)

アドレスオフセット : 0x4C

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|----|------|------|------|------|------|------|------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EF | ADD6 | ADD5 | ADD4 | ADD3 | ADD2 | ADD1 | ADD0 |
| | | | | | | | | rw | rw | rw | rw | rw | rw | rw | rw |

ビット 15:8 予約済み

ビット 7 EF : 機能有効

このビットは、USB デバイスを有効にするために、ソフトウェアによってセットされます。このデバイスのアドレスは、次の ADD[6:0] ビットに含まれます。このビットが 0 の場合、USB_EPnR レジスタの設定にかかわらず、トランザクションは処理されません。

ビット 6:0 ADD[6:0] : デバイスアドレス

これらのビットは、エニュメレーションプロセス時にホスト PC によって割り当てられた USB 機能を含みます。必要なエンドポイントへのトランザクションを処理するためには、関連する USB_EPnR レジスタのこのフィールドとエンドポイントアドレス (EA) フィールドの両方とも、USB トークンに含まれる情報に一致する必要があります。

バッファテーブルアドレス (USB_BTABLE)

アドレスオフセット : 0x50

リセット値 : 0x0000

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

ビット 15:3 BTABLE[15:3] : バッファテーブル

これらのビットは、専用パケットメモリ内のバッファ割り当てテーブルの開始アドレスを含みます。このテーブルは、各エンドポイントバッファ位置とサイズを記述し、8 バイトの境界に揃えられなければなりません (3 つの最下位ビットが常に 0)。このデバイスにアドレス指定されたトランザクションの開始ごとに、USB ペリフェラルはアドレス指定されたエンドポイントに関連このテーブルの要素を読み出して、バッファ開始位置とバッファサイズを取得します ([パケットバッファの構造と用途 \(844 ページ\)](#) を参照)。

ビット 2:0 予約済み、ハードウェアによって 0 に固定されています。

LPM 制御およびステータスレジスタ (USB_LPMCSR)

アドレスオフセット : 0x54

リセット値 : 0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|-----------|---|---|---|----------|------|---------|--------|
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BESL[3:0] | | | | REM WAKE | Res. | LPM ACK | LPM EN |
| | | | | | | | | r | | | | r | | rw | rw |

ビット 15:8 予約済み

ビット 7:4 **BESL[3:0]** : BESL 値

これらのビットは、最後に ACK された LPM トークンとともに受信した BESL 値を含みます。

ビット 3 **REM_WAKE** : bRemoteWake 値

このビットは、最後に ACK された LPM トークンとともに受信した bRemoteWake 値を含みます。

ビット 2 予約済み

ビット 1 **LPMACK** : LPM トークン確認応答イネーブル

0 : 有効な LPM トークンは NYET です。

1 : 有効な LPM トークンは ACK です。

NYET/ACK は、LPM トランザクションが成功した場合にのみ返されます。

EXT トークンと LPM トークンの両方にエラーがない場合 (それ以外の場合は ERROR)

有効な bLinkState = 0001B (L1) を受信した場合 (それ以外の場合は STALL)

ビット 0 **LPMEN** : LPM サポートイネーブル

このビットは、USB デバイス内の LPM サポートを有効にするために、ソフトウェアによってセットされます。このビットが 0 の場合は、LPM トランザクションは処理されません。

バッテリー充電検出回路(USB_BCDR)

アドレスオフセット : 0x58
リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------------|------|------|-----------|------|------|-----------|-----------|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| DPPU | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PS2 DET | SDET | PDET | DC DET | SDEN | PDEN | DCD EN | BCD EN |
| rw | | | | | | | | r | r | r | r | rw | rw | rw | rw |

ビット 15 **DPPU** : DP プルアップ制御
このビットは、DP ライン上の埋め込まれたプルアップを有効にするために、ソフトウェアによってセットされます。0 にクリアすると、ユーザのソフトウェアが必要とするときに、ホストとの接続が切断されたことを通知します。

ビット 14:8 **予約済み**
ビット 7 **PS2DET** : DM プルアップ検出ステータス
このビットは、PD 中にのみアクティブになって DM 電圧レベルと V_{LGC} 閾値の比較の結果を示します。通常の状況では、DM レベルは閾値を下回る必要があります。閾値を上回る場合は、DM が外部的にハイレベルにプルアップされることを意味します。これは、PS2 ポート (DP および DM ラインの両方をプルアップする) への接続、または BCD 仕様に準拠しない一部の独自の充電器への接続によって発生します。
0 : 通常のポートが検出されました (SDP、ACA、CDP、または DCP に接続されたポート)。
1 : PS2 ポートまたは独自の充電器が検出されました。

ビット 6 **SDET** : 2 次検出 (SD) ステータス
このビットは、SD の結果を示します。
0 : CDP が検出されました。
1 : DCP が検出されました。

ビット 5 **PDET** : 1 次検出 (PD) ステータス
このビットは、PD の結果を示します。
0 : BCD サポートは検出されませんでした (SDP または独自デバイスに接続)。
1 : BCD のサポートが検出されました (ACA、CDP、または DCP に接続)。

ビット 4 **DCDET** : データ接触検出 (DCD) ステータス
このビットは、DCD の結果を示します。
0 : データラインの接触は検出されませんでした。
1 : データラインの接触が検出されました。

ビット 3 **SDEN** : 2 次検出 (SD) モードイネーブル
このビットは、BCD を SD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード (DCD、PD、SD、または OFF) を 1 つだけ選択する必要があります。

この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

- ビット 2 **PDEN** : 1 次検出 (PD) モードイネーブル
- このビットは、BCD を PD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード (DCD、PD、SD、または OFF) を 1 つだけ選択する必要があります。
- ビット 1 **DCDEN** : データ接触検出 (DCD) モードイネーブル
- このビットは、BCD を DCD モードにするために、ソフトウェアによってセットされます。正しく動作するには、検出モード (DCD、PD、SD、または OFF) を 1 つだけ選択する必要があります。
- ビット 0 **BCDEN** : バッテリ充電検出 (BCD) イネーブル
- このビットは、USB デバイス内の BCD サポートを有効にするために、ソフトウェアによってセットされます。有効にすると、USB PHY は BCD によって完全に制御され、通常の通信では使用できなくなります。BCD の検出完了後は、通常の USB 動作を可能にするために、このビットを 0 にクリアして BCD を OFF モードにします。

32.6.2 エンドポイント固有レジスタ

これらのレジスタの数は、USB ペリフェラルが処理する設計になっているエンドポイント数に応じて変化します。USB ペリフェラルは、最大 8 つの双方向エンドポイントをサポートします。各 USB デバイスは、制御エンドポイントをサポートしなければならず、そのアドレス (EA ビット) は 0 にセットされなければなりません。複数のエンドポイントが有効であり、同じエンドポイント番号値を持つ場合、USB ペリフェラルの動作は未定義です。各エンドポイントについて、USB_EPnR レジスタはエンドポイント固有の情報を格納できます。

USB エンドポイント n レジスタ (USB_EPnR)、n=[0..7]

アドレスオフセット : 0x00 から 0x1C

リセット値 : 0x0000

| | | | | | | | | | | | | | | | |
|--------|---------|--------------|----|-------|--------------|----|---------|--------|---------|--------------|---|---------|----|----|----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| CTR_RX | DTOG_RX | STAT_RX[1:0] | | SETUP | EP TYPE[1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX[1:0] | | EA[3:0] | | | |
| rc_w0 | t | t | t | r | rw | rw | rw | rc_w0 | t | t | t | rw | rw | rw | rw |

USB リセットが USB バスから受信されたとき、または CTLR レジスタのビット FRES を通じて強制されたときにもリセットされますが、USB リセットイベントの直前の正しいパケット通知の喪失を回避するために、CTR_RX および CTR_TX ビットは変更されません。各エンドポイントに USB_EPnR レジスタがあり、n はエンドポイント識別子です。

これらのレジスタでは読み出し-変更-書き込みのサイクルは避けるべきです。読み出し操作と書き込み操作の間に、いくつかのビットがハードウェアによってセットされる可能性があり、CPU に変更を検出する時間ができる前に、次の書き込みによって変更される可能性があるためです。この目的のために、この問題の影響を受けるすべてのビットが「不変」の値を持ち、変更が不要なときに使用される必要があります。これらのレジスタはロード命令付きで変更することが推奨され、この場合、ハードウェアによって変更できるすべてのビットには「不変」値が書き込まれます。

ビット 15 CTR_RX : 受信の場合の正しい転送

このビットは、このエンドポイントで OUT/SETUP トランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアのみがこのビットをクリアできます。USB_CNTR レジスタの CTRM ビットがセットされた場合、エンドポイント関連の割り込み条件（常に有効化されます）とともに汎用割り込み条件が生成されます。発生したトランザクションのタイプ（OUT または SETUP）は、下記の SETUP ビットから決めることができます。

NAK または STALL ハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。このビットは読み出し／書き込みですが、0 のみを書き込むことができ、1 を書き込んででも効果はありません。

ビット 14 DTOG_RX : 受信転送の場合のデータトグル

エンドポイントがアイソクロナスでない場合、このビットは次の受信データパケットのデータトグルビット（0=DATA0、1=DATA1）の予期される値を含みます。データ PID 値が一致するデータパケット受信後、ACK ハンドシェイクが USB ホストに送信されたときに、ハードウェアはこのビットをトグルします。エンドポイントが制御エンドポイントとして定義された場合、ハードウェアはこのエンドポイントにアドレス指定された SETUP PID の受信時に、このビットをクリアします。

エンドポイントがダブルバッファリングを使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます（[セクション 32.5.3 : ダブルバッファエンドポイント](#)を参照）。

エンドポイントがアイソクロナスの場合、この種類のエンドポイントではデータトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケットバッファスワッピングをサポートするためにのみ使用されます（[セクション 32.5.4 : アイソクロナス転送](#)を参照）。アイソクロナス転送ではハンドシェイクは使用されないため、ハードウェアはデータパケット受信の終了直後に、このビットをトグルします。

このビットは、値を初期化するため（エンドポイントが制御エンドポイントでないときは必須）、または特定のデータトグル／パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが 0 を書き込むと、DTOG_RX の値は変更されませんが、1 を書き込むと、ビット値がトグルされます。このビットは読み出し／書き込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 13:12 STAT_RX[1:0] : 受信転送の場合のステータスビット

これらのビットは、[表 137 : 受信ステータスエンコード \(866 ページ\)](#) にリストされているエンドポイントステータスに関する情報を含みます。これらのビットは、ソフトウェアによってトグルして、値を初期化することができます。アプリケーションソフトウェアが 0 を書き込むと、値は変更されませんが、1 を書き込むと、ビット値がトグルされます。ハードウェアは、このエンドポイントにアドレス指定された OUT または SETUP（制御のみ）トランザクションに対応する正しい転送が発生したとき（CTR_RX=1）、STAT_RX ビットを NAK にセットするので、ソフトウェアは新しいトランザクションに確認応答する前に受信データを処理する時間があります。

ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します（[セクション 32.5.3 : ダブルバッファエンドポイント](#)を参照）。

エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみなので、ハードウェアはトランザクションの成功後にエンドポイントのステータスを変更できません。ソフトウェアがアイソクロナスエンドポイントの STAT_RX ビットを STALL または NAK にセットした場合、USB ペリフェラルの動作は未定義です。これらのビットは読み出し／書き込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 11 SETUP : セットアップトランザクション完了

このビットは読み出し専用であり、最後に完了したトランザクションが SETUP のときにハードウェアによってセットされます。このビットは制御エンドポイントについてのみ値を変更します。成功した受信トランザクションの場合（CTR_RX イベント）、これを調べて、発生したトランザクションのタイプを判断する必要があります。割り込みサービスルーチンを次の着信トークンによる SETUP ビットの変更から保護するために、このビットは CTR_RX ビットが 1 の間は停止され、CTR_RX が 0 のときに状態が変更されます。このビットは読み出し専用です。

ビット 10:9 EP_TYPE[1:0] : エンドポイントタイプ

これらのビットは、[表 138 : エンドポイントタイプエンコード \(866 ページ\)](#)に記載されているように、このエンドポイントの動作を設定します。エンドポイント 0 は常に制御エンドポイントでなければならず、各 USB 機能にはアドレス 0 を持つ制御エンドポイントが少なくとも 1 つ必要ですが、必要な場合は、他の制御エンドポイントを使用することもできます。制御エンドポイントだけが SETUP トランザクションを処理し、他の種類のエンドポイントによって無視されます。SETUP トランザクションは NAK または STALL で応答することはできません。制御エンドポイントが NAK として定義された場合、SETUP トランザクションが受信されたときには、受信方向では USB ペリフェラルは応答せず、受信エラーをシミュレートします。制御エンドポイントが受信方向で STALL として定義された場合、SETUP パケットは受け入れられ、データを送信し、CTR 割り込みを発行します。エンドポイントが制御エンドポイントの場合でも、OUT トランザクションの受信は通常通りに処理されます。バルクおよび割り込みエンドポイントの動作はほぼ同様ですが、EP_KIND 設定ビットを使用して使用可能な特殊な機能のみが異なります。アイソクロナスエンドポイントの使用については、[セクション 32.5.4 : アイソクロナス転送](#)で説明します。

ビット 8 EP_KIND : エンドポイントの種類

このビットの意味は、EP_TYPE ビットによって設定されたエンドポイントのタイプに依存します。[表 139](#)はさまざまな意味を要約します。

DBL_BUF : このビットは、このバルクエンドポイントのダブルバッファリング機能を有効にするために、ソフトウェアによってセットされます。ダブルバッファバルクエンドポイントの使用については、[セクション 32.5.3 : ダブルバッファエンドポイント](#)で説明します。

STATUS_OUT : このビットは、ステータスアウトトランザクションが予期されることを示すためにソフトウェアによってセットされます。この場合、1 個以上のデータバイトを含んでいるすべての OUT トランザクションは、ACK ではなく STALL で応答されます。このビットを使用して、制御転送時のプロトコルエラーに対するアプリケーションの堅牢性を向上させることができ、その使用は制御エンドポイントのみを対象としています。STATUS_OUT がリセットされると、OUT トランザクションは必要に応じて任意の数のバイトを持つことができます。

ビット 7 CTR_TX : 送信の場合の正しい転送

このビットは、このエンドポイントで IN トランザクションが正常に完了したときに、ハードウェアによってセットされます。ソフトウェアのみがこのビットをクリアできます。USB_CNTR レジスタの CTRM ビットがセットされた場合、エンドポイント関連の割り込み条件（常に有効化されます）とともに汎用割り込み条件が生成されます。

NAK または STALL ハンドシェイクで終了したトランザクションでは、プロトコルエラーやデータトグル不一致の場合のように、実際にはデータは転送されていないため、このビットはセットされません。このビットは読み出し／書き込みですが、0 のみを書き込むことができます。

ビット 6 DTOG_TX : 送信転送の場合のデータトグル

エンドポイントがアイソクロナスでない場合、このビットは次の送信データパケットのデータトグルビット (0=DATA0、1=DATA1) の必要な値を含みます。データパケットの送信後、USB ホストから ACK ハンドシェイクが受信されると、ハードウェアはこのビットをトグルします。エンドポイントが制御エンドポイントとして定義された場合、このエンドポイントにアドレス指定された SETUP PID の受信時に、ハードウェアはこのビットを 1 にセットします。

エンドポイントがダブルバッファ機能を使用している場合、このビットはパケットバッファスワッピングをサポートするためにも使用されます ([セクション 32.5.3 : ダブルバッファエンドポイント](#)を参照)。

エンドポイントがアイソクロナスの場合、この種類のエンドポイントではデータトグルは使用されず、DATA0 パケットのみが送信されるため、このビットはパケットバッファスワッピングをサポートするためにのみ使用されます ([セクション 32.5.4 : アイソクロナス転送](#)を参照)。アイソクロナス転送ではハンドシェイクは使用されないため、ハードウェアはデータパケット送信の終了直後に、このビットをトグルします。

このビットは、値を初期化するため（エンドポイントが制御エンドポイントでないときは必須）、または特定のデータトグル／パケットバッファ使用のために、ソフトウェアによってトグルすることもできます。アプリケーションソフトウェアが 0 を書き込むと、DTOG_TX の値は変更されませんが、1 を書き込むと、ビット値がトグルされます。このビットは読み出し／書き込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 5:4 **STAT_TX[1:0]** : 送信転送の場合のステータスビット

これらのビットは、表 140 にリストされているエンドポイントのステータスに関する情報を含みます。これらのビットは、ソフトウェアによってトグルして、値を初期化できます。アプリケーションソフトウェアが 0 を書き込むと、値は変更されませんが、1 を書き込むと、ビット値がトグルされます。ハードウェアは、このエンドポイントにアドレス指定された IN または SETUP (制御のみ) トランザクションに対応する正しい転送が発生したとき (CTR_TX=1)、STAT_TX ビットを NAK にセットします。次に、ソフトウェアが次の送信データセットを準備するのを待ちます。ダブルバッファバルクエンドポイントは、バッファ可用性条件に基づいてステータスを制御する特殊なトランザクションフロー制御を実装します ([セクション 32.5.3 : ダブルバッファエンドポイント](#) を参照)。

エンドポイントがアイソクロナスとして定義された場合、そのステータスは「VALID」または「DISABLED」のみです。したがって、ハードウェアは、トランザクションの成功後にエンドポイントのステータスを変更することはできません。ソフトウェアがアイソクロナスエンドポイントの STAT_TX ビットを STALL または NAK にセットした場合、USB ペリフェラルの動作は未定義です。これらのビットは読み出し/書き込みですが、1 を書き込むことによってのみ、トグルが可能です。

ビット 3:0 **EA[3:0]** : エンドポイントアドレス

ソフトウェアは、このエンドポイント宛てのトランザクションを識別するために使用される 4 ビットのアドレスをこのフィールドに書き込む必要があります。値は、対応するエンドポイントを有効にする前に書き込まれる必要があります。

表 137. 受信ステータスエンコード

| STAT_RX[1:0] | 意味 |
|--------------|---|
| 00 | DISABLED : このエンドポイントにアドレス指定されたすべての受信リクエストは無視されます。 |
| 01 | STALL : エンドポイントは停止していて、すべての受信リクエストは STALL ハンドシェイクで終了します。 |
| 10 | NAK : エンドポイントは NAK され、すべての受信リクエストは NAK ハンドシェイクで終了します。 |
| 11 | VALID : このエンドポイントは受信に有効です。 |

表 138. エンドポイントタイプエンコード

| EP_TYPE[1:0] | 意味 |
|--------------|---------|
| 00 | BULK |
| 01 | CONTROL |
| 10 | ISO |
| 11 | インタラプト |

表 139. エンドポイントの種類の意味

| EP_TYPE[1:0] | | EP_KIND の意味 |
|--------------|---------|-------------|
| 00 | BULK | DBL_BUF |
| 01 | CONTROL | STATUS_OUT |
| 10 | ISO | 未使用 |
| 11 | インタラプト | 未使用 |

表 140. 送信ステータスエンコード

| STAT_TX[1:0] | 意味 |
|--------------|---|
| 00 | DISABLED : このエンドポイントにアドレス指定されたすべての送信リクエストは無視されます。 |
| 01 | STALL : エンドポイントは停止していて、すべての送信リクエストは STALL ハンドシェイクで終了します。 |
| 10 | NAK : エンドポイントは NAK され、すべての送信リクエストは NAK ハンドシェイクで終了します。 |
| 11 | VALID : このエンドポイントは送信に有効です。 |

32.6.3 バッファディスクリプタテーブル

バッファディスクリプタテーブルはパケットバッファメモリ内にありますが、そのエントリは、USB マクロセルとデバイス間でデータを交換するためのパケットバッファの位置とサイズを設定する追加レジスタとみなすことができます。

最初のパケットメモリ位置は、0x4000 6000 にあります。USB_EPnR レジスタに関連するバッファディスクリプタテーブルのエントリについて、以下に説明します。パケットメモリへのアクセスは、バイト（8 ビット）またはハーフワード（16 ビット）アクセスによってのみ行う必要があります。ワード（32 ビット）アクセスは許されません。

パケットバッファとバッファディスクリプタテーブルの使用に関する詳細については、[パケットバッファの構造と用途 \(844 ページ\)](#) を参照してください。

送信バッファアドレス n (USB_ADDRn_TX)

アドレスオフセット : [USB_BTABLE] + n*8

注 : IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_TX_0 と呼ばれます。
OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_RX_0 と呼ばれます。

| | | | | | | | | | | | | | | | |
|----------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ADDRn_TX[15:1] | | | | | | | | | | | | | | | - |
| rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | rW | - |

ビット 15:1 ADDRn_TX[15:1] : 送信バッファアドレス
これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の IN トークンで送信されるデータを含んでいるパケットバッファの開始アドレスを指します。

ビット 0 パケットメモリはハーフワード長であり、すべてのパケットバッファがハーフワードで揃えられている必要があるため、常に 0 が書き込まれる必要があります。

この資料は、STMicroelectronics NV 並びにその子会社（以下 ST）が英文で記述した資料（以下、「正規英語版資料」）を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス㈱が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス㈱は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

送信バイトカウント n (USB_COUNTn_TX)

アドレスオフセット : [USB_BTABLE] + n*8 + 2

注 : *IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_TX_0 と呼ばれます。*
OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_RX_0 と呼ばれます。

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | COUNTn_TX[9:0] | | | | | | | | | |
| | | | | | | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w |

ビット 15:10 USB 仕様によりパケットサイズは 1023 バイトに制限されるので、これらのビットは使用されません。これらの値は、USB ペリフェラルによって考慮されません。

ビット 9:0 **COUNTn_TX[9:0]** : 送信バイトカウント
これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の IN トークンで送信されるバイト数を含みます。

受信バッファアドレス n (USB_ADDRn_RX)

アドレスオフセット : [USB_BTABLE] + n*8 + 4

注 : *OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_RX_1 と呼ばれます。*
IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_ADDRn_TX_1 と呼ばれます。

| | | | | | | | | | | | | | | | |
|----------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ADDRn_RX[15:1] | | | | | | | | | | | | | | | - |
| r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | r/w | - |

ビット 15:1 **ADDRn_RX[15:1]** : 送信バッファアドレス
これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、次の OUT/SETUP トークンで受信されるデータを含んでいるパケットバッファの開始アドレスを指します。

ビット 0 パケットメモリはハーフワード長であり、すべてのパケットバッファがハーフワードで揃えられている必要があるため、常に 0 が書き込まれる必要があります。

受信バイトカウント n (USB_COUNTn_RX)

アドレスオフセット : [USB_BTABLE] + n*8 + 6

注 : *OUT 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_RX_1 と呼ばれます。*
IN 方向のダブルバッファまたはアイソクロナスエンドポイントの場合、このアドレス位置は USB_COUNTn_TX_1 と呼ばれます。

| | | | | | | | | | | | | | | | |
|--------|----------------|----|----|----|----|----------------|---|---|---|---|---|---|---|---|---|
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| BLSIZE | NUM_BLOCK[4:0] | | | | | COUNTn_RX[9:0] | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | r | r | r | r | r | r | r | r | r | r |

このテーブル位置は、両方ともパケット受信時に必要とされる 2 つの値を格納するために使用されます。最上位ビットは割り当てられるバッファサイズの定義を含み、バッファオーバーフロー検出を可能にし、この位置の最下位部分には、受信終了時に USB ペリフェラルによって実際の受信バイト数を書き込まれます。使用可能なビット数が制限されているため、バッファサイズは割り当てられたメモリブロック数を使用して表され、細かい粒度／小さいバッファと粗い粒度／大きなバッファの間のトレードオフにより、ブロックサイズを選択できます。割り当てられたバッファのサイズは、エンドポイントディスクリプタの一部であり、通常、エニュメレーションプロセス時に maxPacketSize パラメータ値にしたって定義されます (「ユニバーサルシリアルバス仕様」を参照)。

ビット 15 BL_SIZE : ブロックサイズ

このビットは、割り当てられたバッファ領域を定義するために使用されるメモリブロックのサイズを選択します。

- BL_SIZE=0 の場合、メモリブロックは 2 バイト長であり、これは、ハーフワード長のメモリで可能な最小ブロックです。このブロックサイズでは、割り当てられるバッファサイズは 2 ~ 62 バイトです。
- BL_SIZE=1 の場合、メモリブロックは 32 バイト以上であり、USB 仕様によって定義された最大パケット長に達することができます。このブロックサイズでは、割り当てられるバッファサイズは、理論上、32 ~ 1024 バイトであり、USB 標準仕様で可能な最長のパケットサイズです。ただし、適用可能なサイズは、使用可能なバッファメモリによって制限されます。

ビット 14:10 NUM_BLOCK[4:0] : ブロック数

これらのビットは、このパケットバッファに割り当てられるメモリブロック数を定義します。実際の割り当てメモリの量は、表 141 に示されているように、BL_SIZE の値に依存します。

ビット 9:0 COUNTn_RX[9:0] : 受信バイトカウント

これらのビットは、USB_EPnR レジスタに関連するエンドポイントによって、最後の OUT/SETUP トランザクションで受信されたバイト数を含みます。

表 141. 割り当てられるバッファメモリの定義

| NUM_BLOCK[4:0]の値 | BL_SIZE=0 のとき割り当てられるメモリ | BL_SIZE=1 のとき割り当てられるメモリ |
|------------------|-------------------------|-------------------------|
| 0 ('00000) | 不可 | 32 バイト |
| 1 ('00001) | 2 バイト | 64 バイト |
| 2 ('00010) | 4 バイト | 96 バイト |
| 3 ('00011) | 6 バイト | 128 バイト |
| ... | ... | ... |
| 14 ('01110) | 28 バイト | 480 バイト |
| 15 ('01111) | 30 バイト | |
| 16 ('10000) | 32 バイト | |
| ... | ... | ... |
| 29 ('11101) | 58 バイト | |
| 30 ('11110) | 60 バイト | |
| 31 ('11111) | 62 バイト | N/A |

32.6.4 USB レジスタマップ

次の表に、USB レジスタマップとリセット値を示します。

表 142. USB レジスタマップとリセット値

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------|----------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|--------|---------|------------------|-------|------------------|---------|--------|---------|------------------|---------|---|---|---|---|---|---|
| 0x00 | USB_EP0R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x04 | USB_EP1R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | USB_EP2R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x0C | USB_EP3R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x10 | USB_EP4R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x14 | USB_EP5R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x18 | USB_EP6R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | SETUP | EP_TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



表 142. USB レジスタマップとリセット値 (続き)

| オフ セット | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | | |
|---------------|------------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|------|--------------|---------|------------------|---------------|----------|--------|---------------------|-----------|----------|---------|------------------|---------|------------|--------|------|-------|-------|---|
| 0x1C | USB_EP7R | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR_RX | DTOG_RX | STAT_RX [1:0] | 0 | 0 | SETUP | EP TYPE [1:0] | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | EA[3:0] | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | |
| 0x20- 0x3F | 予約済み | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0x40 | USB_CNTR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTRM | PMAOVRM | ERRM | WKUPM | SUSPM | RESETM | SOFM | ESOFM | L1REQM | Res. | L1RESUME | RESUME | FSUSP | LPMODE | PDWN | FRES | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 1 | 1 | | |
| 0x44 | USB_ISTR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | CTR | PMAOVR | ERR | WKUP | SUSP | リセット | SOF | ESOF | L1REQ | Res. | Res. | DIR | EP_ID[3:0] | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | | |
| 0x48 | USB_FNR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | RXDP | RXDM | LOK | LSOF [1:0] | FN[10:0] | | | | | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | x | x | x | x | x | x | x | x | x | x | x | | |
| 0x4C | USB_DADDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | EF | ADD[6:0] | | | | | | | | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 0x50 | USB_BTABLE | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BTABLE[15:3] | | | | | | | | | | | | | | Res. | Res. | Res. | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | |
| 0x54 | USB_LPMCSR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | BESL[3:0] | | | | REMWAKE | Res. | Res. | Res. | Res. | | |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 |
| 0x58 | USB_BCDR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DPPU | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | PS2DET | SDET | PDET | DCDET | SDEN | PDEN | DCDEN | BODEN | |
| | リセット値 | | | | | | | | | | | | | | | | | 0 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

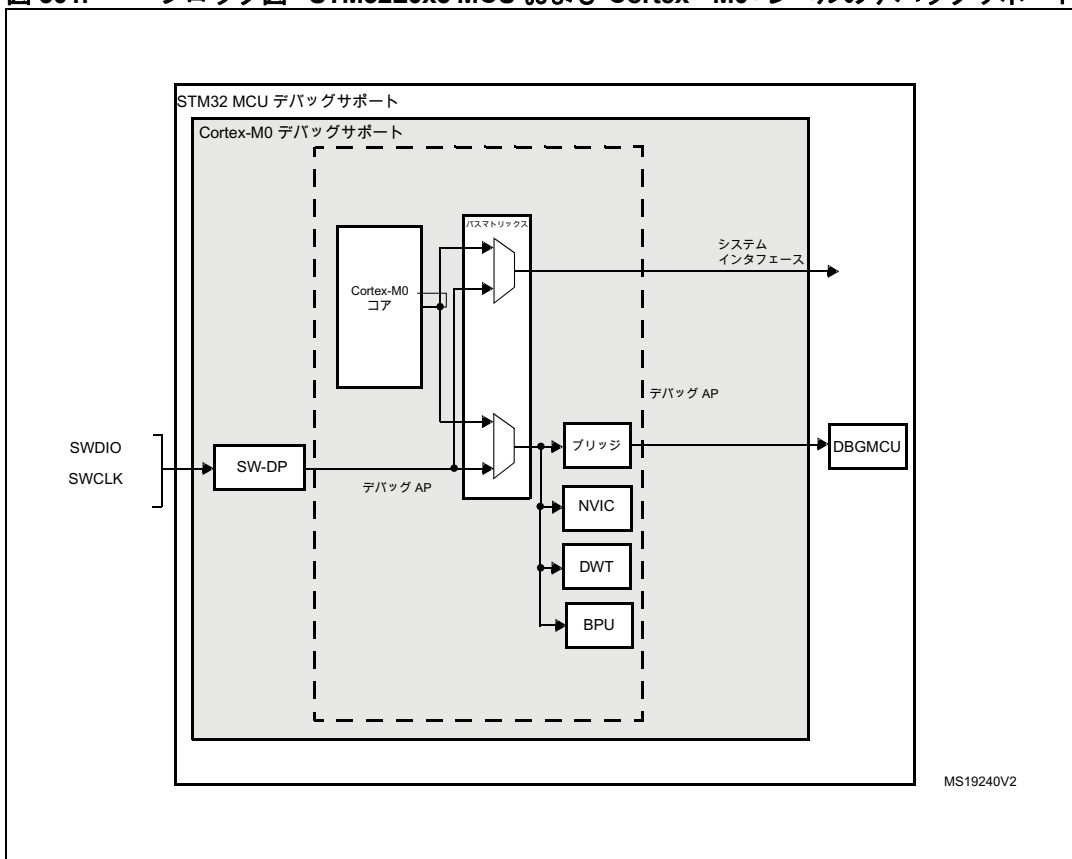
レジスタ境界アドレスについては、[セクション 2.2.2](#) を参照してください。



33.1 概要

● シリアルワイヤ

図 301. ブロック図 - STM32L0x3 MCU および Cortex[®]-M0+レベルのデバッグサポート



1. Cortex®-M0+ コアに内蔵されているデバッグ機能は、ARM CoreSight Design Kit のサブセットです。

ARM Cortex®-M0+ コアは、以下の要素で構成される統合的なオンチップデバッグサポート機能を提供します。

- SW-DP : シリアルワイヤ
- BPU : ブレークポイントユニット
- DWT : データウォッチポイントトリガ

また、STM32L0x3専用の以下のデバッグ機能も内蔵されています。

- 柔軟性の高いデバッグピンの割り当て
- MCU デバッグボックス（低電力モードのサポート、ペリフェラルクロックの制御など）

注： **ARM Cortex®-M0+ コアがサポートするデバッグ機能の詳細については、Cortex®-M0+ Technical Reference Manual を参照してください（[セクション 33.2 : ARM リファレンス資料](#)）。**

33.2 ARM リファレンス資料

- Cortex®-M0+ Technical Reference Manual (TRM)
www.infocenter.arm.com から入手可能です。
- ARM Debug Interface V5
- ARM CoreSight Design Kit revision r1p1 Technical Reference Manual

33.3 ピン名とデバッグポートピン

STM32L0x3 MCUは、使用できるピン数の異なるさまざまなパッケージに組み込まれています。

33.3.1 SWD ポートピン

汎用入出力のオルタネート機能として、2 個のピンが SW-DP 用の出力に使用されます。これらのピンはすべてのパッケージで使用できます。

表 143. SW デバッグポートピン

| SW-DP ピン名 | SW デバッグポート | | ピン割り当て |
|-----------|------------|---------------|--------|
| | タイプ | デバッグ割り当て | |
| SWDIO | 入出力 | シリアルワイヤデータ入出力 | PA13 |
| SWCLK | 入力 | シリアルワイヤクロック | PA14 |

33.3.2 SW-DP ピンの割り当て

リセット後 (SYSRESETn または PORESETn)、SW-DP 用に使用されるピンは、デバッグホストによってすぐに使用可能な専用ピンとして割り当てられます。

ただし、MCU は SWD ポートを無効にすることができますので、汎用 IO (GPIO) に使用する関連ピンを解放する可能性もあります。SW-DP ポートピンを無効にする方法の詳細については、[セクション 9.3.2 : I/O ピンオルタネート機能マルチプレクサと配置 \(208 ページ\)](#) を参照してください。

33.3.3 SWD ピンでの内部プルアップ／プルダウン

SW 入出力がユーザソフトウェアによって解放されると、GPIO コントローラがこれらのピンを制御します。GPIO 制御レジスタがリセット状態にあるとき、入出力も同じ状態になります。

- SWDIO : 入力プルアップ
- SWCLK : 入力プルダウン

プルアップおよびプルダウン抵抗を内蔵しているため、外部抵抗を追加する必要がありません。

33.4 ID コードとロック機構

MCU 内部には、いくつかの ID コードがあります。ST 社がツール設計者に強く推奨するのは、アドレス 0x40015800 に配置されている MCU デバイス ID コードを使用してデバッガをロックすることです。

デバッガ／プログラマツールでは、DEV_ID(15:0) のみを識別に使用してください（リビジョン ID を考慮しません）。

33.4.1 MCU デバイス ID コード

STM32L0x3 には MCU ID コードが内蔵されています。この ID は、ST 社製 MCU の部品番号とダイのリビジョンを識別します。

このコードは、ソフトウェアデバッグポート（2 本の ピン）、またはユーザソフトウェアによってアクセスできます。

DBG_IDCODE

アドレス : 0x4001 5800

32 ビットアクセスのみサポートされます。読み出し専用。

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

ビット 31:16 **REV_ID(15:0)** リビジョン識別子

このフィールドは、デバイスのリビジョンを示します。

リビジョン A : REV_ID = 0x1000

リビジョン Z : REV_ID = 0x1008

ビット 15:12 予約済み。0b0110 を読み出します。

ビット 11:0 **DEV_ID(11:0)** : デバイス識別子

このフィールドは、デバイス ID を示します。

デバイス ID は 0x417 です。

33.5 SWD ポート

33.5.1 SWD プロトコルの概要

この同期式シリアルプロトコルでは、次の 2 個のピンを使用します。

- SWCLK : ホストからターゲットへのクロック
- SWDIO : 双方向

このプロトコルでは、2 バンクのレジスタ (DPACC レジスタと APACC レジスタ) の読み出し／書き込みが可能です。

ビットは、ワイヤ上を LSB ファーストで転送されます。

SWDIO を双方向管理するには、ボード上でラインがプルアップされている必要があります (ARM の推奨値は 100 k Ω)。

プロトコルで SWDIO の方向が変化するたびに、ラインがホストからもターゲットからも駆動されないターンアラウンド時間が挿入されます。このターンアラウンド時間のデフォルトは 1 ビット時間ですが、SWCLK 周波数の設定によって調整できます。

33.5.2 SWD プロトコルシーケンス

各シーケンスは 3 つのフェーズで構成されます。

1. ホストによって送信されるパケットリクエスト (8 ビット)
2. ターゲットによって送信される確認応答 (3 ビット)
3. ホストまたはターゲットによって送信されるデータ転送フェーズ (33 ビット)

表 144. パケットリクエスト (8 ビット)

| ビット | 名称 | 説明 |
|-----|--------|--|
| 0 | Start | “1”である必要があります。 |
| 1 | APnDP | 0 : DP アクセス 1 : AP アクセス |
| 2 | RnW | 0 : 書き込みリクエスト 1 : 読み出しリクエスト |
| 4:3 | A[3:2] | DP/AP レジスタのアドレスフィールド (表 148 (881 ページ) を参照) |
| 5 | Parity | 先行するビットの 1 ビットパリティ |
| 6 | STOP | 0 |
| 7 | Park | ホストによって駆動されません。プルアップの効果で、ターゲットには「1」と読み出される必要があります。 |

DPACC レジスタと APACC レジスタの詳細については、Cortex[®]-M0+ TRM を参照してください。

パケットリクエストの後には、必ずホストもターゲットもラインを駆動しないターンアラウンド時間 (デフォルトでは 1 ビット) が続きます。

表 145. ACK 応答 (3 ビット)

| ビット | 名称 | 説明 |
|-----|------------|---------------------------------------|
| 0~2 | ACK (確認応答) | 001 : FAULT 010 : WAIT 100 : OK |

読み出しトランザクションの場合や、受信した ACK 応答が"WAIT" または"FAULT" の場合にのみ、ACK 応答の後にターンアラウンド時間が続く必要があります。

表 146. データ転送 (33 ビット)

| ビット | 名称 | 説明 |
|------|-----------------|----------------------|
| 0~31 | WDATA または RDATA | 書き込み/読み出しデータ |
| 32 | Parity | 32 データビットの 1 ビットパリティ |

読み出しトランザクションの場合にのみ、データ転送の後にターンアラウンド時間が続く必要があります。

33.5.3 SW-DP ステートマシン (リセット、アイドル状態、ID コード)

SW-DP のステートマシンには、SW-DP を識別する内部 ID コードがあります。これは JEP-106 規格に準じています。この ID コードは、ARM のデフォルトであり、**0x0BB11477** (Cortex[®]-M0+に対応) がセットされています。

注 : **SW-DP ステートマシンは、ターゲットがこの ID コードを読み出すまで無効であることに注意してください。**

- パワーオンリセット後、またはラインが 50 サイクルより長い間ハイレベルにあったあとでは、SW-DP ステートマシンはリセット状態になります。
- リセット状態のあと、ラインが 2 サイクル以上の間ローレベルであれば、SW-DP ステートマシンはアイドル状態になります。
- リセット状態のあとは、まずアイドル状態に入り、次に DP-SW ID CODE レジスタの読み出しアクセスを行う **必要があります**。そうしないと、ターゲットは、他のトランザクションに対して ACK 応答の"FAULT"を発行します。

SW-DP ステートマシンの詳細については、Cortex[®]-M0+ TRM および CoreSight Design Kit r1p0 TRMを参照してください。

33.5.4 DP と AP の読み出し／書き込みアクセス

- DP への読み出しアクセスはポストされません。つまり、ターゲットは、ACK 応答が"OK"の場合はただちに応答し、ACK 応答が"WAIT"の場合は遅れて応答します。
- AP への読み出しアクセスはポストされます。つまり、アクセスの結果は次の転送時に返されます。次のアクセスが AP アクセスでない場合、結果を得るには DP-RDBUFF レジスタを読み出す必要があります。
AP 読み出しアクセスが成功したかどうかを判断するため、DP-CTRL/STAT レジスタの READOK フラグは、AP 読み出しアクセスまたは RDBUFF 読み出しリクエストのたびに更新されます。
- SW-DP は、DP と AP の両方の書き込みに使用できる書き込みバッファを実装しているため、たとえ他のトランザクションが未処理であっても、書き込み動作を受け付けることができます。書き込みバッファがフルのとき、ターゲットの ACK 応答は"WAIT"です。例外として、IDCODE 読み出し、CTRL/STAT 読み出し、または ABORT 書き込みは、書き込みバッファがフルであっても受け付けられます。
- 非同期クロックドメイン SWCLK と HCLK によって、書き込みを内部的に有効にするには、書き込みトランザクション後（パリティビット後）に SWCLK の 2 サイクルが余分に必要となります。これらのサイクルは、ラインをローレベルに駆動している間（アイドル状態）に適用してください。
これは、パワーアップリクエストのために CTRL/STAT の書き込みを行う際に特に重要です。パワーアップを必要とする次のトランザクションがただちに発生すると、そのトランザクションは失敗します。

33.5.5 SW-DP レジスタ

これらのレジスタへのアクセスは、APnDP = 0 のときに開始されます。

表 147. SW-DP レジスタ

| A[3:2] | 読み出し／書き込み | SELECT レジスタの CTRLSEL ビット | レジスタ | 注 |
|--------|-----------|--------------------------------|--------------|--|
| 00 | 読み出し | | IDCODE | 製造者コードは、Cortex®-M0+ のデフォルトの ARM コードに設定されます： 0x0BB11477 (SW-DP を識別)。 |
| 00 | 書き込み | | ABORT | |
| 01 | 読み出し／書き込み | 0 | DP CTRL/STAT | 用途は以下のとおりです。 – システム／デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – ブッシュ比較／ブッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読み出し |
| 01 | 読み出し／書き込み | 1 | WIRE CONTROL | 物理的なシリアルポートプロトコルの設定（ターンアラウンド時間など）を行います。 |
| 10 | 読み出し | | READ RESEND | 元の AP 転送を反復しなくても、破壊されたデバッグ転送からの読み出しデータの復旧を可能にします。 |
| 10 | 書き込み | | SELECT | 現在のアクセスポートと有効な 4 ワードレジスタウィンドウを選択します。 |
| 11 | 読み出し／書き込み | | 読み出しバッファ | AP アクセスはポストされるため、この読み出しバッファは効果的です (AP 読み出しリクエストの結果は、次の AP トランザクションで取得できる)。この読み出しバッファは、新しいトランザクションを開始することなく、前回の読み出しの結果として AP から出力されるデータをキャプチャします。 |

33.5.6 SW-AP レジスタ

これらのレジスタへのアクセスは、APnDP = 1 のときに開始されます。

次の項目の組み合わせで、多くの AP レジスタをアドレス指定します。

- シフトされた値 A[3:2]
- DP SELECT レジスタの現在値

表 148. シフトされた値 A[3:2] によってアドレス指定される 32 ビットデバッグポートレジスタ

| アドレス | A[3:2] の値 | 説明 |
|------|-----------|---|
| 0x0 | 00 | 予約済みであり、リセット値に保持する必要があります。 |
| 0x4 | 01 | DP CTRL/STAT レジスタ：次の目的で使用されます。 – システム/デバッグパワーアップのリクエスト – AP アクセス用の転送動作の設定 – プッシュ比較/プッシュ検証動作の制御 – ステータスフラグ（オーバーラン、パワーアップの確認応答）の読み出し |
| 0x8 | 10 | DP SELECT レジスタ：現在のアクセスポートと有効な 4 ワードレジスタウィンドウの選択に使用されます。 – ビット 31:24 (APSEL)：現在の AP を選択します。 – ビット 23:8：予約済み – ビット 7:4 (APBANKSEL)：現在の AP で有効な 4 ワードレジスタウィンドウを選択します。 – ビット 3:0：予約済み |
| 0xC | 11 | DP RDBUFF レジスタ：一連の動作のあとで（新しい JTAG-DP 動作をリクエストせずに）、デバッガが最終結果を得られるようにします。 |

33.6 コアデバッグ

コアデバッグはコアデバッグレジスタを通じてアクセスされます。これらのレジスタへのデバッグアクセスには、デバッグアクセスポートを使用します。コアデバッグは 4 個のレジスタで構成されています。

表 149. コアデバッグレジスタ

| レジスタ | 説明 |
|-------|--|
| DHCSR | 32 ビットのデバッグホールド制御/ステータスレジスタ： レジスタプロセッサの状態についてのステータス情報を提供し、コアデバッグを有効にし、プロセッサのホールドとステップ実行を行います。 |
| DCRSR | 17 ビットのデバッグコアレジスタセクタレジスタ： データの転送先または転送元となるプロセッサレジスタを選択します。 |
| DCRDR | 32 ビットのデバッグコアレジスタデータレジスタ： DCRSR（セクタ）レジスタによって選択されたプロセッサとの間でレジスタの読み出しおよび書き込みに使用するデータを保持します。 |
| DEMCR | 32 ビットのデバッグ例外/モニタ制御レジスタ： ベクタキャッチとデバッグモニタの制御を行います。 |

これらのレジスタは、システムリセットによってはリセットされません。パワーオンリセットによってのみリセットされます。詳細については、Cortex®-M0+ TRM を参照してください。

リセット時に停止させるには、以下の手順が必要です。

- デバッグ例外／モニタ制御レジスタのビット 0 (VC_CORRESET) を有効にします。
- デバッグホールド制御／ステータスレジスタのビット 0 (C_DEBUGEN) を有効にします。

33.7 BPU (ブレイクポイントユニット)

Cortex®-M0+ BPU を実装することで、4 つのブレイクポイントレジスタが提供されます。BPU は、ARMv7-M (Cortex-M3 および Cortex-M4) で使用可能なフラッシュバッチブレイクポイント (FPB) ブロックのサブセットです。

33.7.1 BPU の機能

プロセッサのブレイクポイントは PC ベースのブレイクポイント機能を実装しています。

BPU CoreSight 識別レジスタ、およびそれらのアドレスやアクセスタイプに関する詳細は、ARMv6-M ARM and the ARM CoreSight Components Technical Reference Manual を参照してください。

33.8 DWT (データウォッチポイント)

Cortex®-M0+ DWT を実装することで、2 つのウォッチポイントレジスタが提供されます。

33.8.1 DWT の機能

プロセッサのウォッチポイントは、データアドレスと PC ベースのウォッチポイント機能を果たす PC サンプリングレジスタの両方を実装し、コンパレータのアドレスマスキングをサポートしています (ARMv6-M ARMを参照)。

33.8.2 DWT プログラムカウンタサンプルレジスタ

データウォッチポイントユニットを実装しているプロセッサは、ARMv6-M のオプション品である DWT プログラムカウンタサンプルレジスタ (DWT_PCSR)も実装しています。このレジスタによって、デバッグは、プロセッサを停止せずに定期的に PC をサンプリングすることができます。これにより、粒度の荒いプロファイリングが提供されます。詳細については、ARMv6-M ARM を参照してください。

Cortex®-M0+ DWT_PCSR は、条件コードに成功した命令と失敗した命令の両方を記録します。

33.9 MCU デバッグコンポーネント (DBG)

MCU デバッグコンポーネントは、デバッグによる以下のサポート機能を支援します。

- 低電力モード
- ブレークポイントにおける、タイマ、ウォッチドッグ、および I2C のクロック制御

33.9.1 低電力モードのデバッグサポート

低電力モードに入るには、WFI または WFE 命令を実行する必要があります。

MCU はいくつかの低電力モードを実装しており、CPU クロックを無効にしたり、CPU の消費電力を低減したりすることができます。

デバッグセッション中には、コアは FCLK や HCLK をオフにすることはできません。デバッグ時のデバッグ接続に必要なため、これらを有効な状態に保つ必要があります。MCU は、ユーザが低電力モードでソフトウェアをデバッグするための特殊な手段を備えています。

このため、デバッグホストは、最初にいくつかのデバッグ設定レジスタをセットして、低電力モード動作を変更する必要があります。

- SLEEP モードの場合、FCLK および HCLK は有効のままです。したがって、このモードは標準的なデバッグ機能になんら制限を加えないことが分かります。
- STOP および STANDBY モードの場合、DBG_STOP ビットをデバッグによって事前にセットする必要があります。

これによって、内部 RC オシレータクロックが、STOP モードで FCLK と HCLK を供給できます。

33.9.2 タイマ、ウォッチドッグ、および I²C のデバッグサポート

ブレークポイントにおいては、以下に示すタイマのカウンタやウォッチドッグの動作方法を選択する必要があります。

- ブレークポイントの中でもカウントを継続できます。この動作は、たとえば、PWM がモータを制御しているときに一般的に必要です。
- ブレークポイントの中でカウントを停止できます。この動作はウォッチドッグ用に必要です。

I²C の場合、ブレークポイントにおいては SMBUS タイムアウトをブロックするように選択できます。

33.9.3 デバッグ MCU 設定レジスタ (DBG_CR)

このレジスタを使用して、デバッグ中に MCU を設定できます。次の設定が可能です。

- 低電力モードのサポート

この DBG_CR はアドレス 0x4001 5804 に配置されます。

このレジスタは PORESET によって非同期にリセットされます(システムリセットではリセットされません)。システムリセット中に、デバッグによって書き込みが可能です。

デバッグホストがこれらの機能をサポートしない場合でも、ユーザソフトウェアによってこれらのレジスタへの書き込みが可能です。

アドレス : 0x04

32 ビットアクセスのみサポートされます。

POR リセット : 0x0000 0000 (システムリセットではリセットされません)

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

ビット 31:3 予約済みであり、リセット値に保持する必要があります。

ビット 2 **DBG_STANDBY** : デバッグ STANDBY モード

0 : (FCLK=オフ、HCLK=オフ) デジタル部全体が電源オフになります。

ソフトウェアから見て、STANDBY モードから抜けることは、リセットベクタのフェッチと同じになります (ただし、いくつかのステータスビットは MCU が STANDBY モードから再開していることを示す)。

1 : (FCLK=オン、HCLK=オン) この場合、デジタル部は電源オフ状態ではなく、FCLK と HCLK は引き続きアクティブ状態の内部 RC オシレータから供給されます。さらに、MCU は STANDBY モード中にシステムリセットを生成するため、STANDBY モードから抜けることはリセットからのフェッチと同じになります。

ビット 1 **DBG_STOP** : デバッグ STOP モード

0 : (FCLK=オフ、HCLK=オフ) STOP モードでは、クロックコントローラがすべてのクロック (HCLK と FCLK を含む) を無効にします。STOP モードから抜けると、クロック設定はリセット後の設定と同じになります。したがって、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります。

1 : (FCLK=オン、HCLK=オン) この場合、STOP モードに入ると、FCLK と HCLK は STOP モードでもアクティブ状態を維持する内部 RC オシレータから供給されます。STOP モードから抜けるとき、ソフトウェアは、クロックコントローラを再プログラムして、PLL やクリスタル発振子などを有効にする必要があります (DBG_STOP = 0 の場合と同様です)。

ビット 0 **DBG_SLEEP** : デバッグ SLEEP モード

0 : SLEEP モードでは、FCLK はソフトウェアによって事前に設定されたシステムクロックによって駆動され、HCLK は無効にされます。クロックコントローラの設定はリセットされず、事前にプログラムされた状態のままです。したがって、SLEEP モードから抜けるときに、ソフトウェアでクロックコントローラを再設定する必要はありません。

1 : この場合、SLEEP モードに入ると、HCLK には FCLK と同じクロック (ソフトウェアによって事前に設定されたシステムクロック) が供給されます。

33.9.4 デバッグ MCU APB1 凍結レジスタ (DBG_APB1_FZ)

DBG_APB1_FZ レジスタは、デバッグにおいて MCU の設定に使用します。以下に示す APB ペリフェラルに関する設定が可能です。

- タイマクロックカウンタの凍結
- I2C SMBUS タイムアウトの凍結
- システムウィンドウ型ウォッチドッグおよび独立型ウォッチドッグのカウンタの凍結サポート

この DBG_APB1_FZ はアドレス 0x4001 5808 に配置されます。

このレジスタは POR によって非同期にリセットされます (システムリセットではリセットされません)。システムリセット中に、デバッグによって書込みが可能です。

アドレスオフセット : 0X08

32 ビットアクセスのみサポートされます。

パワーオンリセット (POR) : 0x0000 0000 (システムリセットではリセットされません)

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|------------------|------|------|---------------|---------------|--------------|------|------|------|---------------|---------------|---------------|------|------|------|---------------|
| DBG_LPTIMER_STOP | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_I2C2_STOP | DBG_I2C1_STOP | Res. | Res. | Res. | Res. | Res. |
| rw | | | | | | | | | rw | rw | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | DBG_IWDG_STOP | DBG_WWDG_STOP | DBG_RTC_STOP | Res. | Res. | Res. | Res. | Res. | DBG_TIM6_STOP | Res. | Res. | Res. | DBG_TIM2_STOP |
| | | | rw | rw | rw | | | | | | rw | | | | rw |

ビット 31 **DBG_LPTIMER_STOP** : コア停止時に LPTIM1 カウンタは停止

- 0 : コアが停止しても、LPTIM1 カウンタのクロックが供給されます。
- 1 : コア停止時に RTC カウンタのクロックは停止します。

ビット 30:23 予約済みであり、リセット値に保持する必要があります。

ビット 22 **DBG_I2C2_STOP** : コア停止時に I2C2 SMBUS タイムアウトモードは停止

- 0 : 通常モードと同じ動作です。
- 1 : I2C2 SMBUS タイムアウトは行われません。

ビット 21 **DBG_I2C1_STOP** : コア停止時に I2C1 SMBUS タイムアウトモードは停止

- 0 : 通常モードと同じ動作です。
- 1 : I2C1 SMBUS タイムアウトは行われません。

ビット 20:13 予約済みであり、リセット値に保持する必要があります。

ビット 12 **DBG_IWDG_STOP** : コア停止時にデバッグ独立型ウォッチドッグは停止

- 0 : コアが停止しても独立型ウォッチドッグカウンタのクロックは継続されます。
- 1 : コア停止時に独立型ウォッチドッグカウンタのクロックは停止します。

ビット 11 **DBG_WWDG_STOP** : コア停止時にデバッグウィンドウ型ウォッチドッグは停止

- 0 : コアが停止してもウィンドウ型ウォッチドッグカウンタのクロックは継続されます。
- 1 : コア停止時にウィンドウ型ウォッチドッグカウンタのクロックは停止します。



- ビット 10 **DBG_RTC_STOP** : コア停止時にデバッグ RTC は停止
- 0 : コアが停止しても当該 RTC カウンタのクロックは供給されます。
 - 1 : コア停止時に当該 RTC カウンタのクロックは停止します。
- ビット 9:5 予約済みであり、リセット値に保持する必要があります。
- ビット 4 **DBG_TIM6_STOP** : コア停止時に TIM6 カウンタは停止
- 0 : コアが停止しても TIM6 カウンタのクロックが供給されます。
 - 1 : コア停止時に TIM6 カウンタのクロックは停止します。
- ビット 3:1 予約済みであり、リセット値に保持する必要があります。
- ビット 0 **DBG_TIM2_STOP** : コア停止時に TIM2 カウンタは停止
- 0 : コアが停止しても TIM2 カウンタのクロックが供給されます。
 - 1 : コア停止時に TIM2 カウンタのクロックは停止します。

33.9.5 デバッグ MCU APB2 凍結レジスタ (DBG_APB2_FZ)

DBG_APB2_FZ レジスタは、デバッグにおいて MCU の設定に使用します。以下に示す APB ペリフェラルに関する設定が可能です。

- タイマクロックカウンタの凍結

このレジスタはアドレス 0x4001580C に配置されます。

このレジスタは POR によって非同期にリセットされます (システムリセットではリセットされません)。システムリセット中に、デバッグによって書込みが可能です。

アドレス : 0x0C

32 ビットアクセスのみサポートされます。

POR : 0x0000 0000 (システムリセットではリセットされません)

| | | | | | | | | | | | | | | | |
|------|------|------|------|------|------|------|------|------|----------------|------|------|------|----------------|------|------|
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. |
| | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_TIM22_STOP | Res. | Res. | Res. | DBG_TIM21_STOP | Res. | Res. |
| | | | | | | | | | nw | | | | nw | | |

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

ビット 6 **DBG_TIM22_STOP** : コア停止時に TIM22 カウンタは停止

- 0 : コアが停止しても TIM22 カウンタのクロックは供給されます。
- 1 : コア停止時に TIM22 カウンタのクロックは停止します。

ビット 2 **DBG_TIM21_STOP** : コア停止時に TIM21 カウンタは停止

- 0 : コアが停止しても TIM21 カウンタのクロックは供給されます。
- 1 : コア停止時に TIM21 カウンタのクロックは停止します。

ビット 1:0 予約済みであり、リセット値に保持する必要があります。

33.10 DBG レジスタマップ

次の表にデバッグレジスタの一覧を示します。

表 150. DBG レジスタマップとリセット値

| アドレス | レジスタ | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------------|----------------------|------------------|------|------|------|------|------|------|------|------|---------------|---------------|------|------|------|------|------|------|------|------|---------------|---------------|--------------|------|------|------|------|----------------|------|---------------|----------------|----------|-----------|
| 0x40015800 | DBG_IDCODE | REV_ID | | | | | | | | | | | | | | | | Res. | Res. | Res. | Res. | DEV_ID | | | | | | | | | | | |
| | リセット値 ⁽¹⁾ | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | | | | | X | X | X | X | X | X | X | X | X | X | X | X |
| 0x40015804 | DBG_CR | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_STANDBY | DBG_STOP | DBG_SLEEP |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 |
| 0x40015808 | DBG_APB1_FZ | DBG_LPTIMER_STOP | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_I2C2_STOP | DBG_I2C1_STOP | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_IWDG_STOP | DBG_WWDG_STOP | DBG_RTC_STOP | Res. | Res. | Res. | Res. | Res. | Res. | DBG_TIM6_STOP | Res. | Res. | Res. |
| | リセット値 | 0 | | | | | | | | | 0 | 0 | | | | | | | | | 0 | 0 | 0 | | | | | | 0 | | | | 0 |
| 0x4001580C | DBG_APB2_FZ | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | Res. | DBG_TIM22_STOP | Res. | Res. | DBG_TIM21_STOP | Res. | Res. |
| | リセット値 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | 0 | | |

1. リセット値は製品によって異なります。詳細については、[セクション 33.4.1 : MCU デバイス ID コード](#)を参照してください。



34 デバイス電子署名

このセクションは、特に指定がない限り、STM32L0x3 ファミリ全体に適用されます。

電子署名は、フラッシュメモリモジュールのシステムメモリ領域に格納され、JTAG/SWD または CPU を使用して読み出すことができます。電子署名に含まれる出荷時にプログラムされた識別データを使用すれば、ユーザファームウェアやその他の外部デバイスは、そのインタフェースを STM32L0x3 マイクロコントローラの特性に自動的に整合させることができます。

34.1 メモリサイズレジスタ

34.1.1 フラッシュサイズレジスタ

ベースアドレス : 0x1FF8 007C

読み出し専用 = 0xFFFF (X は出荷時にプログラムされます)

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| F_SIZE | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 15:0 F_SIZE : フラッシュメモリサイズ

このフィールドに格納されている値は、デバイスのフラッシュメモリサイズを KB 単位で示します。

例 : 0x0040 = 64 KB

34.2 ユニークデバイス ID レジスタ (96 ビット)

このユニークデバイス識別子は、以下の用途に最適です。

- シリアル番号として使用
- 内部フラッシュメモリをプログラムする前に、このユニーク ID をソフトウェア暗号プリミティブやプロトコルと組み合わせて使用する際に、フラッシュメモリ内のコードのセキュリティを高めるためのセキュリティキーとして使用
- セキュアなブートプロセスなどの起動に使用

96 ビットのユニークデバイス識別子は、状況やデバイスの違いとは無関係にユニークなリファレンス番号を提供します。ユーザは、これらのビットを変更できません。

96 ビットのユニークデバイス識別子は、さまざまな方法で 1 バイト/ハーフワード/ワード単位で読み出し、カスタムアルゴリズムを使用して連結することもできます。

ベースアドレス : 0x1FF8 0050
アドレスオフセット : 0x00
読み出し専用 = 0xFFFF (X は出荷時にプログラムされます)

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

ビット 31:0 **U_ID(31:0)** : 31:0 ユニーク ID ビット

アドレスオフセット : 0x04
読み出し専用 = 0xFFFF (X は出荷時にプログラムされます)

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 63 | 62 | 61 | 60 | 59 | 58 | 57 | 56 | 55 | 54 | 53 | 52 | 51 | 50 | 49 | 48 |
| U_ID(63:48) | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |
| 47 | 46 | 45 | 44 | 43 | 42 | 41 | 40 | 39 | 38 | 37 | 36 | 35 | 34 | 33 | 32 |
| U_ID(47:32) | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 63:32 **U_ID(63:32)** : 63:32 ユニーク ID ビット

アドレスオフセット : 0x14
読み出し専用 = 0xFFFF XXXX (X は出荷時にプログラムされます)

| | | | | | | | | | | | | | | | |
|-------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 95 | 94 | 93 | 92 | 91 | 90 | 89 | 88 | 87 | 86 | 85 | 84 | 83 | 82 | 81 | 80 |
| U_ID(95:80) | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |
| 79 | 78 | 77 | 76 | 75 | 74 | 73 | 72 | 71 | 70 | 69 | 68 | 67 | 66 | 65 | 64 |
| U_ID(79:64) | | | | | | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

ビット 95:64 **U_ID(95:64)** : 95:64 ユニーク ID ビット



この資料は、STMicroelectronics NV 並びにその子会社 (以下 ST) が英文で記述した資料 (以下、「正規英語版資料」) を、皆様のご理解の一助として頂くために ST マイクロエレクトロニクス株式が英文から和文へ翻訳して作成したものです。この資料は現行の正規英語版資料の近時の更新に対応していない場合があります。この資料は、あくまでも正規英語版資料をご理解頂くための補助的参考資料のみにご利用下さい。この資料で説明される製品のご検討及びご採用にあたりましては、必ず最新の正規英語版資料を事前にご確認下さい。ST 及び ST マイクロエレクトロニクス株式は、現行の正規英語版資料の更新により製品に関する最新の情報を提供しているにも関わらず、当該英語版資料に対応した更新がなされていないこの資料の情報に基づいて発生した問題や障害などにつきましては如何なる責任も負いません。

索引

A

| | |
|------------|---------|
| ADC_CCR | 307 |
| ADC_CFGR1 | 299 |
| ADC_CFGR2 | 302 |
| ADC_CHSELR | 305 |
| ADC_CR | 297 |
| ADC_DR | 305-306 |
| ADC_IER | 295 |
| ADC_ISR | 294 |
| ADC_SMPR | 303 |
| ADC_TR | 304 |
| AES_CR | 396 |
| AES_DINR | 399 |
| AES_DOUTR | 399 |
| AES_IVR0 | 401 |
| AES_IVR1 | 402 |
| AES_IVR2 | 403 |
| AES_IVR3 | 403 |
| AES_KEYR1 | 400 |
| AES_KEYR2 | 401 |
| AES_KEYR3 | 401 |
| AES_KEYRx | 400 |
| AES_SR | 398 |

C

| | |
|-----------|-----|
| COMP1_CSR | 327 |
| COMP2_CSR | 328 |
| CRC_CR | 99 |
| CRC_DR | 98 |
| CRC_IDR | 99 |
| CRC_INIT | 100 |
| CRC_POL | 100 |
| CRS_CFGR | 200 |
| CRS_CR | 198 |
| CRS_ICR | 203 |
| CRS_ISR | 201 |

D

| | |
|-------------|-----|
| DAC_CR | 317 |
| DAC_DHR12L1 | 321 |
| DAC_DHR12R1 | 320 |
| DAC_DHR8R1 | 321 |
| DAC_DOR1 | 321 |
| DAC_SR | 322 |
| DAC_SWTRIGR | 320 |
| DBG_APB1_FZ | 885 |

| | |
|-------------|---------|
| DBG_APB2_FZ | 887 |
| DBG_CR | 884 |
| DBG_IDCODE | 876 |
| DBGMCU_CR | 884 |
| DMA_CCRx | 242 |
| DMA_CMARx | 245-246 |
| DMA_CNDTRx | 244 |
| DMA_CPARx | 244 |
| DMA_IFCR | 241 |
| DMA_ISR | 240 |

E

| | |
|------------|-----|
| EXTI_EMR | 259 |
| EXTI_FTSR | 260 |
| EXTI_IMR | 258 |
| EXTI_PR | 262 |
| EXTI_RTSR | 259 |
| EXTI_SWIER | 261 |

F

| | |
|---------------|--------|
| FLASH_ACR | 81 |
| FLASH_CR | 86 |
| FLASH_KEYR | 83 |
| FLASH_OPTKEYR | 86-87 |
| FLASH_OTPR | 90 |
| FLASH_PDKEYR | 86 |
| FLASH_PECR | 83 |
| FLASH_PEKEYR | 86 |
| FLASH_PRGKEYR | 86 |
| FLASH_SR | 86, 88 |
| FLASH_WRPROT | 92 |
| FW_CR | 112 |
| FW_CSL | 110 |
| FW_CSSA | 109 |
| FW_NVDSL | 111 |
| FW_NVDSA | 110 |
| FW_VDSL | 112 |
| FW_VDSA | 111 |

G

| | |
|------------|-----|
| GPIOx_AFRH | 219 |
| GPIOx_AFRL | 219 |
| GPIOx_BRR | 220 |
| GPIOx_BSRR | 217 |
| GPIOx_IDR | 216 |
| GPIOx_LCKR | 218 |

| | |
|---------------------|-----|
| GPIOx_MODER | 214 |
| GPIOx_ODR | 217 |
| GPIOx_OSPEEDR | 215 |
| GPIOx_OTYPER | 215 |
| GPIOx_PUPDR | 216 |

I

| | |
|---------------------|--------------|
| I2C_ISR | 681 |
| I2Cx_CR1 | 109-111, 671 |
| I2Cx_CR2 | 110-113, 674 |
| I2Cx_ICR | 683 |
| I2Cx_OAR1 | 677 |
| I2Cx_OAR2 | 678 |
| I2Cx_PECR | 684 |
| I2Cx_RXDR | 685 |
| I2Cx_TIMEOUTR | 680 |
| I2Cx_TIMINGR | 679 |
| I2Cx_TXDR | 685 |
| IWDG_KR | 562 |
| IWDG_PR | 563 |
| IWDG_RLR | 564 |
| IWDG_SR | 565 |
| IWDG_WINR | 566 |

L

| | |
|-------------------|-----|
| LCD_CLR | 356 |
| LCD_CR | 350 |
| LCD_RAM | 357 |
| LPTIMx_ARR | 556 |
| LPTIMx_CFGR | 552 |
| LPTIMx_CMP | 556 |
| LPTIMx_CNT | 557 |
| LPTIMx_CR | 555 |
| LPTIMx_ICR | 550 |
| LPTIMx_IER | 551 |
| LPTIMx_ISR | 549 |
| LPUART_ICR | 789 |

P

| | |
|---------------|-----|
| PWR_CR | 138 |
| PWR_CSR | 141 |

R

| | |
|---------------------|-----|
| RCC_AHBENR | 175 |
| RCC_AHBSTR | 170 |
| RCC_AHBSMENR | 182 |
| RCC_APB1ENR | 179 |
| RCC_APB1RSTR | 172 |
| RCC_APB1SMENR | 184 |

| | |
|---------------------|-----|
| RCC_APB2ENR | 177 |
| RCC_APB2RSTR | 171 |
| RCC_APB2SMENR | 183 |
| RCC_CCIPR | 186 |
| RCC_CFGR | 161 |
| RCC_CICR | 167 |
| RCC_CIER | 164 |
| RCC_CIFR | 166 |
| RCC_CR | 157 |
| RCC_CRRCR | 161 |
| RCC_CSR | 187 |
| RCC_ICSCR | 160 |
| RCC_IOPENR | 174 |
| RCC_IOPRSTR | 169 |
| RCC_IOPSMENR | 181 |
| REF_CFGR3 | 227 |
| RNG_CR | 407 |
| RNG_DR | 408 |
| RNG_SR | 407 |
| RTC_ALRMAR | 601 |
| RTC_ALRMBR | 602 |
| RTC_ALRMBSSR | 614 |
| RTC_BKxR | 616 |
| RTC_CALR | 609 |
| RTC_CR | 593 |
| RTC_DR | 592 |
| RTC_ISR | 596 |
| RTC_OR | 615 |
| RTC_PRER | 599 |
| RTC_SHIFTR | 605 |
| RTC_SSR | 604 |
| RTC_TR | 591 |
| RTC_TSDR | 607 |
| RTC_TSSSR | 608 |
| RTC_TSTR | 606 |
| RTC_WPR | 603 |
| RTC_WUTR | 600 |

S

| | |
|--------------------|-----|
| SPI_CR1 | 830 |
| SPI_CR2 | 832 |
| SPI_CRCPR | 835 |
| SPI_DR | 834 |
| SPI_I2SCFGR | 836 |
| SPI_I2SPR | 838 |
| SPI_RXCR | 835 |
| SPI_SR | 833 |
| SPI_TXCR | 836 |
| SYSCFG_CFGR1 | 224 |
| SYSCFG_CFGR2 | 225 |



| | |
|----------------------|-----|
| SYSCFG_EXTICR1 | 229 |
| SYSCFG_EXTICR2 | 229 |
| SYSCFG_EXTICR3 | 230 |
| SYSCFG_EXTICR4 | 230 |

T

| | |
|-------------------|------------------------|
| TIM2_OR | 470 |
| TIM21_OR | 523 |
| TIM22_OR | 524 |
| TIMx_ARR | 465, 521, 538 |
| TIMx_CCER | 463, 520 |
| TIMx_CCMR1 | 459, 517 |
| TIMx_CCMR2 | 462 |
| TIMx_CCR1 | 466, 522 |
| TIMx_CCR2 | 466, 522 |
| TIMx_CCR3 | 467 |
| TIMx_CCR4 | 467 |
| TIMx_CNT | 465, 521, 537 |
| TIMx_CR1 | 450, 508, 523-524, 535 |
| TIMx_CR2 | 452, 510, 536 |
| TIMx_DCR | 468 |
| TIMx_DIER | 455, 514, 536 |
| TIMx_DMAR | 468 |
| TIMx_EGR | 458, 516, 537 |
| TIMx_PSC | 465, 521, 538 |
| TIMx_SMCR | 453, 511 |
| TIMx_SR | 456, 514, 537 |
| TSC_CR | 369 |
| TSC_ICR | 372 |
| TSC_IER | 371 |
| TSC_IOASCR | 374 |
| TSC_IOCCR1 | 375 |
| TSC_I OGCSR | 376 |
| TSC_I OGxCR | 376 |
| TSC_I OHCR | 374 |
| TSC_I OSCR1 | 375 |
| TSC_ISR | 373 |

U

| | |
|-------------------|-----|
| USART_BRR | 784 |
| USART_CR1 | 777 |
| USART_CR2 | 780 |
| USART_CR3 | 782 |
| USART_DR | 790 |
| USART_ISR | 785 |
| USART_SR | 785 |
| USARTx_BRR | 742 |
| USARTx_CR1 | 730 |
| USARTx_CR2 | 733 |
| USARTx_CR3 | 737 |
| USARTx_GTPR | 743 |

| | |
|---------------------|----------|
| USARTx_ICR | 751 |
| USARTx_ISR | 746 |
| USARTx_RDR | 752 |
| USARTx_RQR | 745 |
| USARTx_RTOR | 744 |
| USARTx_TDR | 753 |
| USB_ADDRn_RX | 869 |
| USB_ADDRn_TX | 868 |
| USB_BTABLE | 860, 862 |
| USB_CNTR | 854, 861 |
| USB_COUNTn_RX | 870 |
| USB_COUNTn_TX | 869 |
| USB_DADDR | 860 |
| USB_EPnR | 863 |
| USB_FNR | 859 |
| USB_ISTR | 856 |

W

| | |
|----------------|-----|
| WWDG_CFR | 572 |
| WWDG_CR | 571 |
| WWDG_SR | 572 |

35 改版履歴

表 151. 文書改版履歴

| 日付 | 版 | 変更内容 |
|-----------------|---|--|
| 2014 年 2 月 11 日 | 1 | 初版リリース。 |
| 2014 年 4 月 29 日 | 2 | <p>TSC、RNG、ASE、および通信のセクションの並べ替え。</p> <p>システムおよびメモリの概要</p> <p>セクション 2.3: 内蔵 SRAM および セクション 2.4: ブート設定を更新。</p> <p>フラッシュメモリ/データ EEPROM</p> <p>図 6: RDP レベルを更新。</p> <p>セクション 3.4.2: PcROP (独自仕様コード読み出し保護) の内容を修正。</p> <p>ファイアウォール</p> <p>EEPROM PROG、PROG EEPROM、PROG/PROGRAM MEMORY の名称をフラッシュプログラムメモリに変更。</p> <p>PWR</p> <p>VREF+ に関する注 3 を セクション 図 10.: 電源の概要の下に追加。</p> <p>セクション 6.1.1: 独立した A/D および DAC コンバータ用電源と基準電圧を更新。</p> <p>RCC</p> <p>図 16: 簡略化されたリセット回路図および 図 17: クロックツリーを更新。</p> <p>セクション 7.2.4: HSI48 クロックを更新。</p> <p>MCOSSEL[2:0] を MCOSSEL[3:0] に変更、セクション 7.2.14: クロック信号出力の定義を更新 (セクション 7.3.4: クロック設定レジスタ (RCC_CFGR))。</p> <p>TOUCHRST を TSCRST に名称変更 (セクション 7.3.9: AHB ペリフェラルリセットレジスタ (RCC_AHBSTR))。</p> <p>TOUCHRST を TSCRST に名称変更 (セクション 7.3.17: SLEEP モード時 AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBSMENR))。HSI48MSEL を HSI48SEL に名称変更 (セクション 7.3.20: クロック設定レジスタ (RCC_CCIPR))。</p> |



表 151. 文書改版履歴（続き）

| 日付 | 版 | 変更内容 |
|-------------------------|---|---|
| 2014 年 4 月 29 日 （続き） | 2 | <p>SYSCFG : セクション 10.1 : 概要を更新。 REF_CTRL レジスタ (セクション 10.2.3) を REF_CFGR3 に名称変更、ENBUFLP_BGAP_COMP を ENBUF_VREFINT_COMP に変更、ENBUF_BGAP_ADC を ENBUF_VREFINT_ADC に変更、ENBUF_SENSOR_ADC を ENBUF_TSENSE_ADC に変更。EN_BGAP を EN_VREFINT に変更し、説明を更新。</p> <p>割り込み 表 45 : ベクタテーブルを更新。</p> <p>ADC : 全セクションから JADSTART を削除。 ADC 供給要件を更新 (セクション 14.2 : ADC の主な機能 および 図 32 : ADC ブロック図)。 セクション名「Analog Reference from the PMU」を セクション : ADC 内部電圧レギュレータのアナログ基準 に変更し、内容を更新。セクション 14.4.2 : 較正 (ADCAL) の内容を修正。 セクション 14.10 : 温度センサと内部基準電圧 で、110 °C から 130 °C に変更。 SCANDIR の説明を更新 (セクション 14.13.4 : ADC 設定レジスタ 1 (ADC_CFGR1))。</p> <p>DAC : セクション : 1 つの LFSR 生成による独立トリガ および セクション : 1 つの三角波生成による独立トリガ を セクション 15.3.4 : DAC チャンネル変換 に追加。</p> <p>COMP : 図 61 : コンパレータ 1 および 2 のブロック図 を更新。 図 61 : コンパレータ 1 および 2 のブロック図 に挙げられているビットを更新。 COMP1_CSR を COMP2_CSR ビットに名称変更。</p> <p>LCD COM および SEG 出力ピンの機能を更新 (セクション 17.3.7 : COM および SEG の多重化 (セクション 17.6.1 : LCD 制御レジスタ (LCD_CR)))。</p> <p>TIM2 : セクション 21.4.19 : TIM2 オプションレジスタ (TIM2_OR) を更新。 すべてのレジスタを 16 ビット長に変更。</p> |

表 151. 文書改版履歴（続き）

| 日付 | 版 | 変更内容 |
|-------------------------|---|--|
| 2014 年 4 月 29 日 （続き） | 2 | <p>TIM21/22 : 図 142 : 汎用タイマのブロック図 (TIM21/22) を更新。 セクション 22.4.2 : TIM21/22 制御レジスタ 2 (TIMx_CR2) を追加。</p> <p>USART CTSE ビットの説明を更新 (セクション 29.7.3 : 制御レジスタ 3 (USARTx_CR3))。</p> <p>DEBUG DBG_TIM20_STOP と DBG_TIM21_STOP をそれぞれ DBG_TIM21_STOP と DBG_TIM22_STOP に変更。 REV_ID を更新 (セクション 33.4.1 : MCU デバイス ID コード)。</p> |

表 152. 文書改版履歴（日本語版）

| 日付 | 版 | 変更内容 |
|------------|---|-------------|
| 2015年8月10日 | 1 | 日本語版初版リリース。 |

重要なお知らせ（よくお読み下さい）

STMicroelectronics NV およびその子会社（以下、ST）は、ST製品及び本書の内容をいつでも予告なく変更、修正、改善、改定及び改良する権利を留保します。購入される方は、発注前にST製品に関する最新の関連情報を必ず入手してください。ST製品は、注文請書発行時点で有効なSTの販売条件に従って販売されます。

ST製品の選択並びに使用については購入される方が全ての責任を負うものとします。購入される方の製品上の操作や設計に関してSTは一切の責任を負いません。

明示又は黙示を問わず、STは本書においていかなる知的財産権の実施権も許諾致しません。

本書で説明されている情報とは異なる条件でST製品が再販された場合、その製品についてSTが与えたいかなる保証も無効となります。

ST およびST ロゴはSTMicroelectronics の商標です。その他の製品またはサービスの名称は、それぞれの所有者に帰属します。

本書の情報は本書の以前のバージョンで提供された全ての情報に優先し、これに代わるものです。

© 2015 STMicroelectronics - All rights reserved