

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。STM32L151xx、STM32L152xx、STM32L162xx および STM32L100xx マイクロコントローラのメモリおよびペリフェラルの使用方法について詳細に説明しています。特に記載がない限り、このマニュアルでは STM32L151xx、STM32L152xx、STM32L162xx および STM32L100xx のバリエーションの総称として STM32L1xxxx と記載しています。

STM32L1xxxx は異なるメモリサイズ、パッケージ、ペリフェラルを持ったマイクロコントローラの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、各データシートを参照してください。

内蔵不揮発性メモリのプログラミング、消去、および保護については、[セクション 3: フラッシュプログラムメモリとデータEEPROM \(フラッシュ\)](#) を参照してください。

ARM® Cortex®-M3 コアの詳細な情報に関しては、Cortex®-M3 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

次の資料が www.arm.com から入手できます。

- Cortex®-M3 テクニカルリファレンスマニュアルは、Web サイト (<http://infocenter.arm.com>) から入手できます。

次の資料が www.st.com から入手できます。

- STM32L151xx STM32L152xx データシート
- STM32L162xx データシート
- STM32L100xx データシート
- STM32F10xxx/20xxx/21xxx/L1xxxx Cortex®-M3 プログラミングマニュアル
- STM32L15xx6/8/B から STM32L15xx6/8/B-A および STM32L100x6/8/B から STM32L100x6/8/B-A (TN1176) への移行
- STM32L15/6xRC-A から STM32L15/6xRC および STM32L15/6xVC-A から STM32L15/6xVC (TN1177) への移行
- STM32L15/6xxD から STM32L15/6xxE (TN1178) への移行

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	使用可能なペリフェラル	38
1.3	用語	39
1.4	製品カテゴリの定義	40
2	システムアーキテクチャとメモリの概要	42
2.1	システムアーキテクチャ	42
2.2	メモリ構成	46
2.3	メモリマップ	47
2.4	内蔵 SRAM	50
2.5	フラッシュメモリの概要	50
2.6	ビットバンディング	50
2.7	ブート設定	51
3	フラッシュプログラムメモリとデータ EEPROM（フラッシュ）	53
3.1	フラッシュメモリの概要	53
3.2	フラッシュメモリ構成	53
3.3	読み出しインタフェース	58
3.3.1	CPU クロック周波数とフラッシュメモリ読み出し時間との関係	58
3.3.2	フラッシュアクセスが 64 ビットのときの命令プリフェッチ	60
3.3.3	データ管理	61
3.4	メモリ命令	61
3.4.1	メモリのロック / アンロック	61
3.4.2	メモリの消去	63
3.4.3	メモリへの書き込み	64
3.4.4	書き込み中読出し（RWW）	70
3.5	オプションバイトの説明	73
3.5.1	オプションバイトブロックの書き込み	76
3.6	書き込み／消去機能のクイックリファレンス	77
3.7	メモリ保護	79
3.7.1	プログラム EEPROM およびデータ EEPROM の読み出し保護機能（RDP）	79

3.7.2	プログラムメモリの書き込み保護機能 (WRP)	81
3.7.3	書き込み保護機能エラーフラグ	81
3.7.4	PCROP	81
3.8	割り込み	82
3.9	レジスタの説明	83
3.9.1	アクセス制御レジスタ (FLASH_ACR)	83
3.9.2	書き込み/消去制御レジスタ (FLASH_PECR)	84
3.9.3	パワーダウンキーレジスタ (FLASH_PDKEYR)	86
3.9.4	書き込み/消去キーレジスタ (FLASH_PEKEYR)	87
3.9.5	プログラムメモリキーレジスタ (FLASH_PRGKEYR)	87
3.9.6	オプションバイトキーレジスタ (FLASH_OPTKEYR)	88
3.9.7	ステータスレジスタ (FLASH_SR)	88
3.9.8	オプションバイトレジスタ (FLASH_OBR)	90
3.9.9	書き込み保護機能レジスタ (FLASH_WRPx)	91
3.9.10	レジスタマップ	92
4	CRC 計算ユニット	95
4.1	CRC の概要	95
4.2	CRC の主な特徴	95
4.3	CRC の機能説明	96
4.4	CRC レジスタ	96
4.4.1	データレジスタ (CRC_DR)	96
4.4.2	独立型データレジスタ (CRC_IDR)	97
4.4.3	制御レジスタ (CRC_CR)	97
4.4.4	CRC レジスタマップ	97
5	電源コントローラ (PWR)	99
5.1	電源	99
5.1.1	A/D と DAC コンバータの独立した電源と基準電圧	101
5.1.2	独立した LCD 電源	101
5.1.3	RTC レジスタと RTC バックアップレジスタ	102
5.1.4	電圧レギュレータ	102
5.1.5	ダイナミック電圧スケーリング管理	103
5.1.6	ダイナミック電圧スケーリング設定	104
5.1.7	2.0 V 以下に VDD 低下した場合の電圧レギュレータとクロックの管理	105
5.1.8	VCORE 範囲を変更する際の電圧レギュレータとクロックの管理	105

5.2	電源供給スーパーバイザ	106
5.2.1	パワーオンリセット (POR) / パワーダウンリセット (PDR)	108
5.2.2	ブラウナウトリセット	109
5.2.3	プログラマブル電圧検出器 (PVD)	110
5.2.4	内部電圧基準 (VREFINT)	111
5.3	低消費電力モード	111
5.3.1	低消費電力モードでのクロック動作	113
5.3.2	システムクロックの低下	113
5.3.3	ペリフェラルクロックゲーティング	113
5.3.4	低消費電力 run モード (LP run)	114
5.3.5	SLEEP モード	114
5.3.6	低消費電力 sleep モード (LP スリープ)	116
5.3.7	Stop モード	118
5.3.8	STANDBY モード	120
5.3.9	RTC とコンパレータを使用した、Stop モードおよび Standby モードからの デバイスのウェイクアップ	121
5.4	電源制御レジスタ	123
5.4.1	PWR 電源制御レジスタ (PWR_CR)	123
5.4.2	PWR 電源制御/ステータスレジスタ (PWR_CSR)	126
5.4.3	PWR レジスタマップ	128
6	リセットおよびクロック制御 (RCC)	129
6.1	リセット	129
6.1.1	システムリセット	129
6.1.2	電源リセット	130
6.1.3	RTC レジスタとバックアップレジスタのリセット	130
6.2	クロック	131
6.2.1	HSE クロック	133
6.2.2	HSI クロック	134
6.2.3	MSI クロック	135
6.2.4	PLL	136
6.2.5	LSE クロック	136
6.2.6	LSI クロック	137
6.2.7	システムクロック (SYSCLK) の選択	137
6.2.8	システムクロックソース周波数対電圧範囲	137
6.2.9	クロックセキュリティシステム (CSS)	138
6.2.10	LSE のクロックセキュリティシステム	138

6.2.11	RTC クロックと LCD クロック	139
6.2.12	ウォッチドッグクロック	139
6.2.13	クロック出力機能	139
6.2.14	TIM9/TIM10/TIM11 による内部 / 外部クロックの測定	140
6.2.15	TIM9/TIM10/TIM11 用のクロック非依存システムクロックソース	141
6.3	RCC レジスタ	142
6.3.1	クロック制御レジスタ (RCC_CR)	142
6.3.2	内部クロックソース校正レジスタ (RCC_ICSCR)	144
6.3.3	クロック設定レジスタ (RCC_CFGR)	145
6.3.4	クロック割り込みレジスタ (RCC_CIR)	148
6.3.5	AHB ペリフェラルリセットレジスタ (RCC_AHBSTR)	151
6.3.6	APB2 ペリフェラルリセットレジスタ (RCC_APB2RSTR)	153
6.3.7	APB1 ペリフェラルリセットレジスタ (RCC_APB1RSTR)	154
6.3.8	AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBENR)	157
6.3.9	APB2 ペリフェラルクロックイネーブルレジスタ (RCC_APB2ENR)	159
6.3.10	APB1 ペリフェラルクロックイネーブルレジスタ (RCC_APB1ENR)	161
6.3.11	低消費電力モードレジスタでの AHB ペリフェラルクロックイネーブル (RCC_AHBLPENR) 163	
6.3.12	低消費電力モードレジスタでの APB2 ペリフェラルクロックイネーブル (RCC_APB2LPENR)	165
6.3.13	低消費電力モードレジスタでの APB1 ペリフェラルクロックイネーブル (RCC_APB1LPENR)	167
6.3.14	制御/ステータスレジスタ (RCC_CSR)	169
6.3.15	RCC レジスタマップ	173
7	汎用 I/O (GPIO)	175
7.1	GPIO の概要	175
7.2	GPIO の主な機能	175
7.3	GPIO 機能の説明	175
7.3.1	汎用 I/O (GPIO)	178
7.3.2	I/O ピンマルチプレクサおよびマッピング	178
7.3.3	I/O ポート制御レジスタ	180
7.3.4	I/O ポートデータレジスタ	180
7.3.5	I/O データのビット単位の処理	181
7.3.6	GPIO ロック機構	181
7.3.7	I/O オルタネート機能の入力/出力	181
7.3.8	外部割り込み/ウェイクアップライン	182
7.3.9	入力設定	182

7.3.10	出力設定	183
7.3.11	オルタネート機能設定	184
7.3.12	アナログ設定	185
7.3.13	OSC32_IN / OSC32_OUT ピンの GPIO PC14 / PC15 ポートピンとしての使用	185
7.3.14	OSC_IN/OSC_OUT ピンの GPIO PH0/PH1 ポートピンとしての使用	186
7.3.15	RTC_AF1 オルタネート機能の選択	186
7.4	GPIO レジスタ	187
7.4.1	GPIO ポートモードレジスタ (GPIOx_MODER) (x = A..H)	187
7.4.2	GPIO ポート出力タイプレジスタ (GPIOx_OTYPER) (x = A..H)	188
7.4.3	GPIO ポート出力速度レジスタ (GPIOx_OSPEEDR) (x = A..H)	188
7.4.4	GPIO ポートプルアップ / プルダウンレジスタ (GPIOx_PUPDR) (x = A..H)	189
7.4.5	GPIO ポート入力データレジスタ (GPIOx_IDR) (x = A..H)	189
7.4.6	GPIO ポート出力データレジスタ (GPIOx_ODR) (x = A..H)	190
7.4.7	GPIO ポートビットセット / リセットレジスタ (GPIOx_BSRR) (x = A..H)	190
7.4.8	GPIO ポート設定ロックレジスタ (GPIOx_LCKR) (x = A..H)	191
7.4.9	GPIO オルタネート機能ローレジスタ (GPIOx_AFR1) (x = A..H)	192
7.4.10	GPIO オルタネート機能ハイレジスタ (GPIOx_AFRH) (x = A..H)	193
7.4.11	GPIO ポートビットリセットレジスタ (GPIOx_BRR) (x=A..H)	193
7.4.12	GPIO レジスタマップ	194
8	システム設定コントローラ (SYSCFG) とルーティングインタフェース (RI)	197
8.1	SYSCFG と RI について	197
8.2	RI の主な特徴	197
8.3	RI の機能詳細	201
8.3.1	特別 I/O 設定	201
8.3.2	入力キャプチャのルーティング	205
8.3.3	基準電圧のルーティング	206
8.4	RI レジスタ	207
8.4.1	RI 入力キャプチャレジスタ (RI_ICR)	207
8.4.2	RI アナログスイッチ制御レジスタ (RI_ASCR1)	209
8.4.3	RI アナログスイッチ制御レジスタ 2 (RI_ASCR2)	211

8.4.4	RI ヒステリシス制御レジスタ (RI_HYSCR1)	212
8.4.5	RI ヒステリシス制御レジスタ (RI_HYSCR2)	212
8.4.6	RI ヒステリシス制御レジスタ (RI_HYSCR3)	213
8.4.7	RI ヒステリシス制御レジスタ (RI_HYSCR4)	214
8.4.8	アナログスイッチモードレジスタ (RI_ASMR1)	214
8.4.9	チャネルマスクレジスタ (RI_CMR1)	215
8.4.10	キャプチャレジスタ (RI_CICR1) のチャネル識別	215
8.4.11	アナログスイッチモードレジスタ (RI_ASMR2)	216
8.4.12	チャネルマスクレジスタ (RI_CMR2)	216
8.4.13	キャプチャレジスタ (RI_CICR2) のチャネル識別	217
8.4.14	アナログスイッチモードレジスタ (RI_ASMR3)	217
8.4.15	チャネルマスクレジスタ (RI_CMR3)	218
8.4.16	キャプチャレジスタ (RI_CICR3) のチャネル識別	218
8.4.17	アナログスイッチモードレジスタ (RI_ASMR4)	219
8.4.18	チャネルマスクレジスタ (RI_CMR4)	219
8.4.19	キャプチャレジスタ (RI_CICR4) のチャネル識別	220
8.4.20	アナログスイッチモードレジスタ (RI_ASMR5)	220
8.4.21	チャネルマスクレジスタ (RI_CMR5)	221
8.4.22	キャプチャレジスタ (RI_CICR5) のチャネル識別	221
8.4.23	RI レジスタマップ	222
8.5	SYSCFG レジスタ	224
8.5.1	SYSCFG メモリ再配置レジスタ (SYSCFG_MEMRMP)	224
8.5.2	SYSCFG ペリフェラルモード設定レジスタ (SYSCFG_PMC)	225
8.5.3	SYSCFG 外部割り込み設定レジスタ 1 (SYSCFG_EXTICR1)	226
8.5.4	SYSCFG 外部割り込み設定レジスタ 2 (SYSCFG_EXTICR2)	227
8.5.5	SYSCFG 外部割り込み設定レジスタ 3 (SYSCFG_EXTICR3)	227
8.5.6	SYSCFG 外部割り込み設定レジスタ 4 (SYSCFG_EXTICR4)	228
8.5.7	SYSCFG レジスタマップ	229
9	タッチセンシング I/O	231
9.1	概要	231
9.2	主な特長	231
9.3	機能の説明	232
9.3.1	表面電荷移動アクイジションの概要	232

9.3.2	電荷移動アクイジションの管理	234
9.4	タッチセンシングライブラリ	236
10	割り込みとイベント	237
10.1	ネスト化されたベクタ割り込みコントローラ (NVIC)	237
10.1.1	SysTick 較正值レジスタ	237
10.1.2	割り込みベクタおよび例外ベクタ	238
10.2	外部割り込み/イベントコントローラ (EXTI)	244
10.2.1	主な機能	244
10.2.2	ブロック図	245
10.2.3	ウェイクアップイベント管理	245
10.2.4	機能の説明	246
10.2.5	外部割り込み/イベントラインのマッピング	246
10.3	EXTI レジスタ	248
10.3.1	EXTI 割り込みマスクレジスタ (EXTI_IMR)	248
10.3.2	EXTI イベントマスクレジスタ (EXTI_EMR)	249
10.3.3	EXTI 立ち上がりエッジトリガ 選択レジスタ (EXTI_RTSSR)	249
10.3.4	立ち下がりエッジトリガ 選択レジスタ (EXTI_FTSR)	250
10.3.5	EXTI ソフトウェア割り込みイベントレジスタ (EXTI_SWIER)	250
10.3.6	EXTI ペンディングレジスタ (EXTI_PR)	251
10.3.7	EXTI レジスタマップ	252
11	ダイレクトメモリアクセス (DMA) コントローラ	253
11.1	DMA の概要	253
11.2	DMA の主な機能	253
11.3	DMA 機能の説明	256
11.3.1	DMA トランザクション	257
11.3.2	アービタ	257
11.3.3	DMA チャネル	258
11.3.4	プログラミング可能なデータ幅、データの配置、およびエンディアン	259
11.3.5	エラー管理	260
11.3.6	割り込み	260
11.3.7	DMA リクエストのマッピング	261
11.4	DMA レジスタ	265
11.4.1	DMA 割り込みステータスレジスタ (DMA_ISR)	265
11.4.2	DMA 割り込みフラグクリアレジスタ (DMA_IFCR)	266

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)	267
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)	268
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)	269
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)	269
11.4.7	DMA レジスタマップ	270
12	A/D コンバータ (ADC)	273
12.1	ADC の概要	273
12.2	ADC の主な機能	273
12.3	ADC の機能詳細	274
12.3.1	ADC 電源のオン / オフ制御	277
12.3.2	ADC クロック	278
12.3.3	チャンネルの選択	278
12.3.4	シングル変換モード	279
12.3.5	連続変換モード	279
12.3.6	タイミング図	280
12.3.7	アナログウォッチドッグ	281
12.3.8	スキャンモード	282
12.3.9	インジェクトチャンネルの管理	282
12.3.10	不連続モード	283
12.4	データアライメント	284
12.5	チャンネル単位で設定可能なサンプリングタイム	285
12.6	外部トリガによる変換	286
12.7	変換の中止	288
12.7.1	インジェクトチャンネル	288
12.7.2	レギュラチャンネル	288
12.8	変換分解能	288
12.9	ハードウェアフリーズと低速変換用の遅延挿入モード	289
12.9.1	各レギュラ変換後の遅延挿入	290
12.9.2	自動インジェクト変換の各シーケンスの後への遅延挿入	291
12.10	省電力	292
12.11	データ管理とオーバーラン検出	293
12.11.1	DMA の使用	293

12.11.2	DMA を使用しない変換シーケンスの管理	294
12.11.3	全データを読み出さない変換	294
12.11.4	オーバーラン検出	294
12.12	温度センサと内部基準電圧	295
12.13	内部基準電圧 (V_{REFINT}) の変換	297
12.14	ADC 割り込み	297
12.15	ADC レジスタ	298
12.15.1	ADC ステータスレジスタ (ADC_SR)	298
12.15.2	ADC 制御レジスタ 1 (ADC_CR1)	300
12.15.3	ADC 制御レジスタ 2 (ADC_CR2)	303
12.15.4	ADC サンプルタイムレジスタ 1 (ADC_SMPR1)	306
12.15.5	ADC サンプルタイムレジスタ 2 (ADC_SMPR2)	307
12.15.6	ADC サンプルタイムレジスタ 3 (ADC_SMPR3)	308
12.15.7	ADC インジェクトチャネルデータオフセットレジスタ x (ADC_JOFRx) (x=1..4)	309
12.15.8	ADC ウォッチドッグ高閾値レジスタ (ADC_HTR)	309
12.15.9	ADC ウォッチドッグ低閾値レジスタ (ADC_LTR)	310
12.15.10	ADC レギュラシーケンスレジスタ 1 (ADC_SQR1)	310
12.15.11	ADC レギュラシーケンスレジスタ 2 (ADC_SQR2)	311
12.15.12	ADC レギュラシーケンスレジスタ 3 (ADC_SQR3)	311
12.15.13	ADC レギュラシーケンスレジスタ 4 (ADC_SQR4)	312
12.15.14	ADC レギュラシーケンスレジスタ 5 (ADC_SQR5)	312
12.15.15	ADC インジェクトシーケンスレジスタ (ADC_JSQR)	313
12.15.16	ADC インジェクトデータレジスタ x (ADC_JDRx) (x= 1..4)	314
12.15.17	ADC レギュラデータレジスタ (ADC_DR)	314
12.15.18	ADC サンプルタイムレジスタ 0 (ADC_SMPR0)	315
12.15.19	ADC 共通 ステータスレジスタ (ADC_CSR)	316
12.15.20	ADC 共通制御レジスタ (ADC_CCR)	317
12.15.21	ADC レジスタマップ	318
13	D/A コンバータ (DAC)	321
13.1	DAC の概要	321
13.2	DAC の主な機能	321
13.3	DAC 機能の説明	323
13.3.1	DAC チャネルイネーブル	323
13.3.2	DAC 出力バッファイネーブル	323
13.3.3	DAC データフォーマット	323

13.3.4	DAC 変換	324
13.3.5	DAC 出力電圧	325
13.3.6	DAC トリガ選択	325
13.3.7	DMA リクエスト	326
13.3.8	ノイズ発生	326
13.3.9	三角波生成	327
13.4	デュアル DAC チャンネル変換	328
13.4.1	波形生成なしの独立トリガ	328
13.4.2	1 つの LFSR 生成による独立トリガ	329
13.4.3	異なる LFSR 生成による独立トリガ	329
13.4.4	1 つの三角波生成による独立トリガ	329
13.4.5	異なる三角波生成による独立トリガ	330
13.4.6	同時ソフトウェア開始	330
13.4.7	波形生成なしの同時トリガ	330
13.4.8	1 つの LFSR 生成による同時トリガ	331
13.4.9	異なる LFSR 生成による同時トリガ	331
13.4.10	1 つの三角波生成による同時トリガ	331
13.4.11	異なる三角波生成による同時トリガ	332
13.5	DAC レジスタ	333
13.5.1	DAC 制御レジスタ (DAC_CR)	333
13.5.2	DAC ソフトウェアトリガレジスタ (DAC_SWTRIGR)	336
13.5.3	DAC チャンネル 1 12 ビット右詰めデータ保持レジスタ (DAC_DHR12R1)	337
13.5.4	DAC チャンネル 1 12 ビット左詰めデータ保持レジスタ (DAC_DHR12L1)	338
13.5.5	DAC チャンネル 1 8 ビット右詰めデータ保持レジスタ (DAC_DHR8R1)	338
13.5.6	DAC チャンネル 2 12 ビット右詰めデータ保持レジスタ (DAC_DHR12R2)	339
13.5.7	DAC チャンネル 2 12 ビット左詰めデータ保持レジスタ (DAC_DHR12L2)	339
13.5.8	DAC チャンネル 2 8 ビット右詰めデータ保持レジスタ (DAC_DHR8R2)	340
13.5.9	デュアル DAC 12 ビット右詰めデータ保持レジスタ (DAC_DHR12RD)	340
13.5.10	デュアル DAC 12 ビット左詰めデータ保持レジスタ (DAC_DHR12LD)	341
13.5.11	デュアル DAC 8 ビット右詰めデータ保持レジスタ (DAC_DHR8RD)	342

13.5.12	DAC チャンネル 1 データ出力レジスタ (DAC_DOR1)	342
13.5.13	DAC チャンネル 2 データ出力レジスタ (DAC_DOR2)	343
13.5.14	DAC ステータスレジスタ (DAC_SR)	343
13.5.15	DAC レジスタマップ	344
14	コンパレータ (COMP)	345
14.1	概要	345
14.2	主な機能	345
14.3	COMP クロック	346
14.4	コンパレータ 1 (COMP1)	346
14.5	コンパレータ 2 (COMP2)	348
14.6	ウィンドウモードのコンパレータ	350
14.7	低消費電力モード	350
14.8	割り込み	351
14.9	COMP レジスタ	351
14.9.1	COMP コンパレータ制御およびステータスレジスタ (COMP_CSR)	351
14.9.2	COMP レジスタマップ	354
15	オペアンプ (OPAMP)	355
15.1	OPAMP の概要	355
15.2	OPAMP の主な機能	355
15.3	OPAMP の機能詳細	355
15.3.1	信号ルーティング	356
15.3.2	ADC 入力として OPAMP 出力を使用	357
15.3.3	較正	357
15.4	OPAMP レジスタ	359
15.4.1	OPAMP 制御/ステータスレジスタ (OPAMP_CSR)	359
15.4.2	通常モードの OPAMP オフセットトリミングレジスタ (OPAMP_OTR)	362
15.4.3	低消費電力モードの OPAMP オフセットトリミングレジスタ (OPAMP_LPOTR)	363
15.4.4	OPAMP レジスタマップ	364

16	液晶ディスプレイコントローラ (LCD)	365
16.1	概要	365
16.2	LCD の主な機能	366
16.3	用語	367
16.4	LCD 機能の説明	368
16.4.1	概要	368
16.4.2	周波数ジェネレータ	369
16.4.3	コモンドライバ	371
16.4.4	セグメントドライバ	374
16.4.5	電圧ジェネレータ	378
16.4.6	デッドタイム	380
16.4.7	ダブルバッファメモリ	381
16.4.8	COM と SEG のマルチプレクシング	381
16.4.9	フローチャート	385
16.5	LCD レジスタ	386
16.5.1	LCD 制御レジスタ (LCD_CR)	386
16.5.2	LCD フレーム制御レジスタ (LCD_FCR)	387
16.5.3	LCD ステータスレジスタ (LCD_SR)	389
16.5.4	LCD クリアレジスタ (LCD_CLR)	390
16.5.5	LCD ディスプレイメモリ (LCD_RAM)	391
16.5.6	LCD レジスタマップ	392
17	汎用タイマ (TIM2 から TIM5)	395
17.1	TIM2 から TIM5 の概要	395
17.2	TIM2 から TIM5 の主な機能	395
17.3	TIM2 から TIM5 の機能詳細	397
17.3.1	タイムベースユニット	397
17.3.2	カウンタモード	399
17.3.3	クロック選択	407
17.3.4	キャプチャ／比較チャネル	410
17.3.5	入力キャプチャモード	412
17.3.6	PWM 入力モード	413
17.3.7	強制出力モード	414
17.3.8	出力比較モード	414
17.3.9	PWM モード	415
17.3.10	ワンパルスモード	418

17.3.11	外部イベントによる OCxREF 信号のクリア	419
17.3.12	エンコーダインタフェースモード	420
17.3.13	タイマ入力 XOR 機能	422
17.3.14	タイマと外部トリガの同期	422
17.3.15	タイマの同期	426
17.3.16	デバッグモード	431
17.4	TIMx レジスタ	432
17.4.1	TIMx 制御レジスタ 1 (TIMx_CR1)	432
17.4.2	TIMx 制御レジスタ 2 (TIMx_CR2)	434
17.4.3	TIMx スレーブモード制御レジスタ (TIMx_SMCR)	435
17.4.4	TIMx DMA / 割り込み有効レジスタ (TIMx_DIER)	437
17.4.5	TIMx ステータスレジスタ (TIMx_SR)	438
17.4.6	TIMx イベント生成レジスタ (TIMx_EGR)	440
17.4.7	TIMx およびキャプチャ/比較モードレジスタ 1 (TIMx_CCMR1)	441
17.4.8	TIMx およびキャプチャ/比較モードレジスタ 2 (TIMx_CCMR2)	444
17.4.9	TIMx キャプチャ/比較有効レジスタ (TIMx_CCER)	445
17.4.10	TIMx カウンタ (TIMx_CNT)	447
17.4.11	TIMx プリスケアラ (TIMx_PSC)	447
17.4.12	TIMx 自動再ロードレジスタ (TIMx_ARR)	447
17.4.13	TIMx キャプチャ/比較レジスタ 1 (TIMx_CCR1)	448
17.4.14	TIMx キャプチャ/比較レジスタ 2 (TIMx_CCR2)	448
17.4.15	TIMx キャプチャ/比較レジスタ 3 (TIMx_CCR3)	449
17.4.16	TIMx キャプチャ/比較レジスタ 4 (TIMx_CCR4)	449
17.4.17	TIMx DMA 制御レジスタ (TIMx_DCR)	450
17.4.18	完全転送の TIMx DMA アドレス (TIMx_DMAR)	451
17.4.19	TIM2 オプションレジスタ (TIM2_OR)	452
17.4.20	TIM3 オプションレジスタ (TIM3_OR)	452
17.4.21	TIMx レジスタマップ	453
18	汎用タイマ (TIM9/10/11)	455
18.1	TIM9/10/11 の概要	455
18.2	TIM9/10/11 の主な機能	455
18.2.1	TIM9 の主な機能	455
18.2.2	TIM10/TIM11 の主な機能	457
18.3	TIM9/10/11 の機能説明	459
18.3.1	タイムベースユニット	459
18.3.2	カウンタ・モード	460

18.3.3	クロック選択	463
18.3.4	キャプチャ／比較チャネル	465
18.3.5	入力キャプチャモード	467
18.3.6	PWM 入力モード (TIM9 のみ)	468
18.3.7	強制出力モード	469
18.3.8	出力比較モード	469
18.3.9	PWM モード	470
18.3.10	ワンパルスモード	472
18.3.11	TIM9 外部トリガ同期	473
18.3.12	タイマ同期 (TIM9)	476
18.3.13	デバッグモード	476
18.3.14	エンコーダインタフェースモード (TIM9 の場合)	476
18.4	TIM9 レジスタ	476
18.4.1	TIM9 制御レジスタ 1 (TIMx_CR1)	477
18.4.2	TIM9 制御レジスタ 2 (TIMx_CR2)	479
18.4.3	TIM9 スレーブモード制御レジスタ (TIMx_SMCR)	480
18.4.4	TIM9 割り込みイネーブルレジスタ (TIMx_DIER)	482
18.4.5	TIM9 ステータスレジスタ (TIMx_SR)	483
18.4.6	TIM イベント発生レジスタ (TIMx_EGR)	485
18.4.7	TIMx キャプチャ／比較モードレジスタ 1 (TIMx_CCMR1)	486
18.4.8	TIM9 キャプチャ／比較イネーブルレジスタ (TIMx_CCER)	489
18.4.9	TIM9 カウンタ (TIMx_CNT)	490
18.4.10	TIM9 プリスケラ (TIMx_PSC)	490
18.4.11	TIM9 自動再ロードレジスタ (TIMx_ARR)	490
18.4.12	TIM9 キャプチャ／比較レジスタ 1 (TIMx_CCR1)	491
18.4.13	TIM9 キャプチャ／比較レジスタ 2 (TIMx_CCR1)	491
18.4.14	TIM9 オプションレジスタ 1 (TIM9_OR)	492
18.4.15	TIM9 レジスタマップ	493
18.5	TIM10/11 レジスタ	495
18.5.1	TIM10/11 制御レジスタ 1 (TIMx_CR1)	495
18.5.2	TIM10/11 スレーブモード制御レジスタ 1 (TIMx_SMCR)	496
18.5.3	TIM10/11 割り込みイネーブルレジスタ (TIMx_DIER)	497
18.5.4	TIM10/11 ステータスレジスタ (TIMx_SR)	497
18.5.5	TIM10/11 イベント発生レジスタ (TIMx_EGR)	499
18.5.6	TIM10/11 キャプチャ／比較 モードレジスタ 1 (TIMx_CCMR1)	500
18.5.7	TIM10/11 キャプチャ／比較イネーブルレジスタ (TIMx_CCER)	503
18.5.8	TIM10/11 カウンタ (TIMx_CNT)	504

18.5.9	TIM10/11 プリスケアラ (TIMx_PSC)	504
18.5.10	TIM10/11 自動再ロードレジスタ (TIMx_ARR)	504
18.5.11	TIM10/11 キャプチャ/比較 レジスタ 1 (TIMx_CCR1)	505
18.5.12	TIM10 オプションレジスタ 1 (TIM10_OR)	506
18.5.13	TIM11 オプションレジスタ 1 (TIM11_OR)	507
18.5.14	TIM10/11 レジスタマップ	508
19	基本タイマ (TIM6、TIM7)	511
19.1	TIM6 および TIM7 の概要	511
19.2	TIM6 および TIM7 の主な機能	511
19.3	TIM6 および TIM7 の機能詳細	512
19.3.1	タイムベースユニット	512
19.3.2	カウントモード	513
19.3.3	クロックソース	516
19.3.4	デバッグモード	516
19.4	TIM6 および TIM7 レジスタ	517
19.4.1	TIM6 および TIM7 制御レジスタ 1 (TIMx_CR1)	517
19.4.2	TIM6 および TIM7 制御レジスタ 2 (TIMx_CR2)	519
19.4.3	TIM6 および TIM7 DMA/ 割り込みイネーブルレジスタ (TIMx_DIER)	519
19.4.4	TIM6 および TIM7 ステータスレジスタ (TIMx_SR)	520
19.4.5	TIM6 および TIM7 イベント生成レジスタ (TIMx_EGR)	520
19.4.6	TIM6 および TIM7 カウンタ (TIMx_CNT)	520
19.4.7	TIM6 および TIM7 プリスケアラ (TIMx_PSC)	521
19.4.8	TIM6 および TIM7 自動リロードレジスタ (TIMx_ARR)	521
19.4.9	TIM6 および TIM7 レジスタマップ	522
20	リアルタイムクロック (RTC)	523
20.1	概要	523
20.2	RTC の主な機能	523
20.3	RTC の機能詳細	526
20.3.1	クロックとプリスケアラ	526
20.3.2	リアルタイムクロックとカレンダー	527
20.3.3	プログラム可能なアラーム	527
20.3.4	定期的な自動ウェイクアップ	528
20.3.5	RTC の初期化と設定	529
20.3.6	カレンダーの読み出し	531

20.3.7	RTC のリセット	532
20.3.8	RTC 同期 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)	532
20.3.9	RTC 基準クロック検出	533
20.3.10	RTC の簡易デジタル較正	534
20.3.11	RTC の高精度デジタル較正 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)	535
20.3.12	タイムスタンプ機能	537
20.3.13	タンパ検出	538
20.3.14	較正されたクロックの出力	539
20.3.15	アラーム出力	540
20.4	RTC および低電力モード	540
20.5	RTC 割り込み	541
20.6	RTC レジスタ	542
20.6.1	RTC 時間レジスタ (RTC_TR)	542
20.6.2	RTC データレジスタ (RTC_DR)	543
20.6.3	RTC 制御レジスタ (RTC_CR)	544
20.6.4	RTC 初期化およびステータスレジスタ (RTC_ISR)	547
20.6.5	RTC プリスケアラレジスタ (RTC_PRER)	549
20.6.6	RTC ウェイクアップタイマレジスタ (RTC_WUTR)	551
20.6.7	RTC 較正レジスタ (RTC_CALIBR)	552
20.6.8	RTC アラーム A レジスタ (RTC_ALRMAR)	553
20.6.9	RTC アラーム B レジスタ (RTC_ALRMBR)	554
20.6.10	RTC 書き込み保護レジスタ (RTC_WPR)	555
20.6.11	RTC サブセカンドレジスタ (RTC_SSR)	555
20.6.12	RTC シフト制御レジスタ (RTC_SHIFTR)	556
20.6.13	RTC タイムスタンプ時刻レジスタ (RTC_TSTR)	557
20.6.14	RTC タイムスタンプ日付レジスタ (RTC_TSDR)	558
20.6.15	RTC タイムスタンプサブセカンドレジスタ (RTC_TSSSR)	558
20.6.16	RTC 較正レジスタ (RTC_CALR)	559
20.6.17	RTC タンパおよび代替機能設定レジスタ (RTC_TAFCR)	560
20.6.18	RTC アラーム A サブセカンドレジスタ (RTC_ALRMASR)	562
20.6.19	RTC アラーム B サブセカンドレジスタ (RTC_ALRMBSSR)	563
20.6.20	RTC バックアップレジスタ (RTC_BKPxR)	564
20.6.21	RTC レジスタマップ	565

21	独立型ウォッチドッグ (IWDG)	567
21.1	IWDG の概要	567
21.2	IWDG の主な機能	567
21.3	IWDG の機能詳細	567
21.3.1	ハードウェアウォッチドッグ	567
21.3.2	レジスタのアクセス保護	568
21.3.3	デバッグモード	568
21.4	IWDG レジスタ	569
21.4.1	キーレジスタ (IWDG_KR)	569
21.4.2	プリスケアラレジスタ (IWDG_PR)	570
21.4.3	再ロードレジスタ (IWDG_RLR)	570
21.4.4	ステータスレジスタ (IWDG_SR)	571
21.4.5	IWDG レジスタマップ	572
22	ウィンドウ型ウォッチドッグ (WWDG)	573
22.1	WWDG の概要	573
22.2	WWDG の主な機能	573
22.3	WWDG の機能詳細	573
22.4	ウォッチドッグタイムアウトをプログラム方法	575
22.5	デバッグモード	576
22.6	WWDG レジスタ	577
22.6.1	制御レジスタ (WWDG_CR)	577
22.6.2	設定レジスタ (WWDG_CFR)	578
22.6.3	ステータスレジスタ (WWDG_SR)	578
22.6.4	WWDG レジスタマップ	579
23	高度暗号化標準ハードウェアアクセラレータ (AES)	581
23.1	概要	581
23.2	AES の主な機能	581
23.3	AES の機能詳細	582
23.4	暗号化キーと復号化キー	583
23.5	AES 連結アルゴリズム	584
23.5.1	ECB (電子コードブック)	584
23.5.2	CBC (Cipher Block Chaining)	585
23.5.3	CTR (カウンタモード)	588

23.6	データタイプ	590
23.7	動作モード	592
23.7.1	モード 1 : 暗号化	592
23.7.2	モード 2 : キー派生	593
23.7.3	モード 3 : 復号化	593
23.7.4	モード 4 : キー派生と復号化	594
23.8	AES DMA インタフェース	595
23.9	エラーフラグ	596
23.10	処理時間	596
23.11	AES 割り込み	597
23.12	AES レジスタ	598
23.12.1	AES 制御レジスタ (AES_CR)	598
23.12.2	AES ステータスレジスタ (AES_SR)	600
23.12.3	AES データ入力レジスタ (AES_DINR)	601
23.12.4	AES データ出力レジスタ (AES_DOUTR)	601
23.12.5	AES キーレジスタ 0 (AES_KEYR0) (LSB : キー [31:0])	602
23.12.6	AES キーレジスタ 1 (AES_KEYR0) (キー [63:32])	602
23.12.7	AES キーレジスタ 2 (AES_KEYR2) (キー [95:64])	603
23.12.8	AES キーレジスタ 3 (AES_KEYR3) (MSB : キー [127:96])	603
23.12.9	AES 初期化ベクタレジスタ 0 (AES_IVR0) (LSB : IVR[31:0])	603
23.12.10	AES 初期化ベクタレジスタ 1 (AES_IVR0) (IVR[63:32])	604
23.12.11	AES 初期化ベクタレジスタ 2 (AES_IVR2) (IVR[95:64])	605
23.12.12	AES 初期化ベクタレジスタ 3 (AES_IVR3) (MSB : IVR[127:96])	605
23.12.13	AES レジスタマップ	606
24	ユニバーサルシリアルバス (USB) フルスピードデバイス インタフェース	607
24.1	USB の概要	607
24.2	USB の主な機能	607
24.3	USB の機能詳細	608
24.3.1	USB ブロックの説明	609
24.4	プログラミングの際の注意	611
24.4.1	一般的な USB デバイスのプログラミング	611
24.4.2	システムリセットとパワーオンリセット	611
24.4.3	ダブルバッファエンドポイント	616
24.4.4	アイソクロナス転送	619

24.4.5	サスペンド／レジュームイベント	620
24.5	USB レジスタ	622
24.5.1	一般レジスタ	622
24.5.2	エンドポイント固有のレジスタ	629
24.5.3	バッファディスクリプタテーブル	634
24.5.4	USB レジスタマップ	637
25	フレキシブルスタティックメモリコントローラ (FSMC)	639
25.1	FSMC の主な機能	639
25.2	ブロック図	640
25.3	AHB インタフェース	640
25.3.1	サポートされるメモリおよびトランザクション	641
25.4	外部デバイスアドレスマッピング	641
25.4.1	NOR/PSRAM アドレスマッピング	642
25.5	NOR 型フラッシュ／PSRAM コントローラ	643
25.5.1	外部メモリインタフェース信号	644
25.5.2	サポートされるメモリおよびトランザクション	646
25.5.3	一般的なタイミング規則	647
25.5.4	NOR 型フラッシュ／PSRAM コントローラ非同期トランザクション	647
25.5.5	同期トランザクション	665
25.5.6	NOR/PSRAM 制御レジスタ	671
25.5.7	FSMC レジスタマップ	678
26	I²C(Inter-integrated Circuit) インタフェース	679
26.1	I ² C の概要	679
26.2	I ² C の主な機能	680
26.3	I ² C 機能の詳細	681
26.3.1	モード選択	681
26.3.2	I2C スレーブモード	683
26.3.3	I2C マスタモード	686
26.3.4	エラー条件	691
26.3.5	SDA/SCL ライン制御	692
26.3.6	SMBus	692
26.3.7	DMA リクエスト	695
26.3.8	パケットエラーチェック	697

26.4	I ² C 割り込み	698
26.5	I ² C デバッグモード	700
26.6	I ² C レジスタ	700
26.6.1	I ² C 制御レジスタ 1(I2C_CR1)	700
26.6.2	I ² C 制御レジスタ 2(I2C_CR2)	703
26.6.3	I ² C 自己アドレスレジスタ 1 (I2C_OAR1)	705
26.6.4	I ² C 自己アドレスレジスタ 2 (I2C_OAR2)	705
26.6.5	I ² C データレジスタ (I2C_DR)	706
26.6.6	I ² C ステータスレジスタ 1(I2C_SR1)	707
26.6.7	I ² C ステータスレジスタ 2(I2C_SR2)	711
26.6.8	I ² C クロック制御レジスタ (I2C_CCR)	713
26.6.9	I ² C TRISE レジスタ (I2C_TRISE)	714
26.6.10	I2C レジスタマップ	715
27	USART (Universal synchronous asynchronous receiver transmitter)	717
27.1	USART の概要	717
27.2	USART の主な機能	718
27.3	USART 機能の説明	719
27.3.1	USART キャラクタの説明	721
27.3.2	トランスミッタ	722
27.3.3	レシーバ	725
27.3.4	フラクショナルボーレート生成	730
27.3.5	クロック偏差に対する USART レシーバの許容誤差	739
27.3.6	マルチプロセッサ通信	740
27.3.7	パリティ制御	742
27.3.8	LIN (local interconnection network) モード	743
27.3.9	USART 同期モード	745
27.3.10	単線半二重通信	747
27.3.11	スマートカード	748
27.3.12	IrDA SIR ENDEC ブロック	750
27.3.13	DMA を使用した連続通信	752
27.3.14	ハードウェアフロー制御	754

27.4	USART 割り込み	756
27.5	USART モード設定	757
27.6	USART レジスタ	757
27.6.1	ステータスレジスタ (USART_SR)	758
27.6.2	データレジスタ (USART_DR)	760
27.6.3	ボーレートレジスタ (USART_BRR)	760
27.6.4	制御レジスタ 1 (USART_CR1)	761
27.6.5	制御レジスタ 2 (USART_CR2)	763
27.6.6	制御レジスタ 3 (USART_CR3)	764
27.6.7	ガード時間 / プリスケアラレジスタ (USART_GTPR)	766
27.6.8	USART レジスタマップ	767
28	シリアルペリフェラルインタフェース (SPI)	769
28.1	SPI の概要	769
28.2	SPI と I ² S の主な機能	770
28.2.1	SPI 機能	770
28.2.2	I ² S 機能	771
28.3	SPI 機能詳細	772
28.3.1	概要	772
28.3.2	SPI のスレーブモード設定	776
28.3.3	SPI のマスタモード設定	779
28.3.4	SPI の半二重通信への設定	781
28.3.5	データの送受信手順	782
28.3.6	CRC の計算	788
28.3.7	ステータスフラグ	790
28.3.8	SPI の無効化	791
28.3.9	DMA (Direct Memory Addressing) を使用する SPI 通信	792
28.3.10	エラーフラグ	794
28.3.11	SPI 割り込み	795
28.4	I ² S 機能の詳細	796
28.4.1	I ² S の概要	796
28.4.2	サポートされるオーディオプロトコル	797
28.4.3	クロックジェネレータ	805
28.4.4	I ² S マスタモード	807
28.4.5	I ² S スレーブモード	809
28.4.6	ステータスフラグ	810

28.4.7	エラーフラグ	811
28.4.8	I ² S 割り込み	812
28.5	SPI レジスタと I ² S レジスタ	813
28.5.1	SPI 制御レジスタ 1 (SPI_CR1) (I ² S モードでは使用しません)	813
28.5.2	SPI 制御レジスタ 2 (SPI_CR2)	815
28.5.3	SPI ステータスレジスタ (SPI_SR)	817
28.5.4	SPI データレジスタ (SPI_DR)	818
28.5.5	SPI CRC 多項式レジスタ (SPI_CRCPR) (I ² S モードでは不使用)	819
28.5.6	SPI RX CRC レジスタ (SPI_RXCRCR) (I ² S モードでは不使用)	820
28.5.7	SPI TX CRC レジスタ (SPI_TXCRCR) (I ² S モードでは不使用)	820
28.5.8	SPI_I ² S 設定レジスタ (SPI_I2SCFGR)	821
28.5.9	SPI_I ² S プリスケアラレジスタ (SPI_I2SPR)	822
28.5.10	SPI レジスタマップ	823
29	セキュアデジタル入力 / 出力インタフェース (SDIO)	825
29.1	SDIO の主な機能	825
29.2	SDIO バストポロジ	826
29.3	SDIO の機能詳細	828
29.3.1	SDIO アダプタ	829
29.3.2	SDIO APB2 インタフェース	839
29.4	カード機能詳細	840
29.4.1	カード識別モード	840
29.4.2	カードリセット	840
29.4.3	動作電圧範囲の検証	840
29.4.4	カード識別プロセス	841
29.4.5	ブロック書き込み	842
29.4.6	ブロック読み出し	842
29.4.7	ストリームアクセス、ストリーム書き込み、およびストリーム読み出し (MultiMediaCard のみ)	843
29.4.8	消去：グループ消去とセクタ消去	844
29.4.9	ワイドバス選択または選択解除	845
29.4.10	保護管理	845
29.4.11	カードステータスレジスタ	848
29.4.12	SD ステータスレジスタ	851
29.4.13	SD I/O モード	855
29.4.14	コマンドとレスポンス	856

29.5	レスポンスフォーマット	859
29.5.1	R1 (ノーマルレスポンスコマンド)	859
29.5.2	R1b	860
29.5.3	R2 (CID、CSD レジスタ)	860
29.5.4	R3 (OCR レジスタ)	860
29.5.5	R4 (高速 I/O)	861
29.5.6	R4b	861
29.5.7	R5 (割り込みリクエスト)	862
29.5.8	R6	862
29.6	SDIO I/O カード固有の操作	863
29.6.1	SDIO_D2 信号による SDIO I/O 読み出しウェイト操作	863
29.6.2	SDIO_CK 停止による SDIO 読み出しウェイト操作	863
29.6.3	SDIO サスペンド/レジューム操作	863
29.6.4	SDIO 割り込み	864
29.7	CE-ATA 固有の操作	864
29.7.1	コマンド完了信号ディセーブル	864
29.7.2	コマンド完了信号イネーブル	864
29.7.3	CE-ATA 割り込み	864
29.7.4	CMD61 のアポート	864
29.8	HW フロー制御	865
29.9	SDIO レジスタ	865
29.9.1	SDIO 電源制御レジスタ (SDIO_POWER)	865
29.9.2	SDI クロック制御レジスタ (SDIO_CLKCR)	866
29.9.3	SDIO 引数レジスタ (SDIO_ARG)	867
29.9.4	SDIO コマンドレジスタ (SDIO_CMD)	868
29.9.5	SDIO コマンドレスポンスレジスタ (SDIO_RESPCMD)	869
29.9.6	SDIO レスポンス 1.4 レジスタ (SDIO_RESPx)	869
29.9.7	SDIO データタイマレジスタ (SDIO_DTIMER)	870
29.9.8	SDIO データ長レジスタ (SDIO_DLEN)	870
29.9.9	SDIO データ制御レジスタ (SDIO_DCTRL)	871
29.9.10	SDIO データカウンタレジスタ (SDIO_DCOUNT)	872
29.9.11	SDIO ステータスレジスタ (SDIO_STA)	873
29.9.12	SDIO 割り込みクリアレジスタ (SDIO_ICR)	874
29.9.13	SDIO マスクレジスタ (SDIO_MASK)	876
29.9.14	SDIO FIFO カウンタレジスタ (SDIO_FIFOCNT)	879
29.9.15	SDIO データ FIFO レジスタ (SDIO_FIFO)	879
29.9.16	SDIO レジスタマップ	880

30	デバッグサポート (DBG)	883
30.1	概要	883
30.2	ARM® リファレンス資料	884
30.3	SWJ デバッグポート (シリアルワイヤと JTAG)	885
30.3.1	JTAG-DP または SW-DP の選択メカニズム	885
30.4	ピン名とデバッグポートピン	886
30.4.1	SWJ デバッグポートピン	886
30.4.2	フレキシブルな SWJ-DP ピン配置	886
30.4.3	JTAG ピンでの内部プルアップ / プルダウン	887
30.4.4	シリアルワイヤの使用と、未使用のデバッグピンを GPIO として解放する方法	888
30.5	STM32L1xxxx JTAG TAP 接続	889
30.6	ID コードとロック機構	890
30.6.1	MCU デバイス ID コード	890
30.6.2	バウンダリスキャン TAP	891
30.6.3	Cortex®-M3 TAP	891
30.6.4	Cortex®-M3 JEDEC-106 ID コード	891
30.7	JTAG デバッグポート	892
30.8	SW デバッグポート	893
30.8.1	SW プロトコルの概要	893
30.8.2	SW プロトコルシーケンス	894
30.8.3	SW-DP ステートマシン (リセット、アイドル状態、ID コード)	895
30.8.4	DP と AP の読み出し / 書き込みアクセス	895
30.8.5	SW-DP レジスタ	896
30.8.6	SW-AP レジスタ	896
30.9	AHB-AP (AHB アクセスポート) - JTAG-DP と SW-DP の両方に有効	897
30.10	コアデバッグ	898
30.11	システムリセット中のデバッグホスト接続機能	899
30.12	FPB (フラッシュパッチブレイクポイント)	899
30.13	DWT (データウォッチポイントトリガ)	900
30.14	ITM (計測トレースマクロセル)	900
30.14.1	概要	900
30.14.2	タイムスタンプパケット、同期およびオーバーフローパケット	901
30.15	ETM (組み込みトレースマクロセル)	902
30.15.1	概要	902

30.15.2	信号プロトコル、パケットタイプ	903
30.15.3	主な ETM レジスタ	903
30.15.4	設定例	903
30.16	MCU デバッグコンポーネント (DBGMCU)	904
30.16.1	低電力モードのデバッグサポート	904
30.16.2	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	904
30.16.3	デバッグ MCU 設定レジスタ	905
30.16.4	デバッグ MCU APB1 凍結レジスタ (DBGMCU_APB1_FZ)	907
30.16.5	デバッグ MCU APB2 凍結レジスタ (DBGMCU_APB2_FZ)	909
30.17	TPIU (トレースポートインタフェースユニット)	910
30.17.1	概要	910
30.17.2	TRACE ピンの割り当て	911
30.17.3	TPIU フォーマッタ	912
30.17.4	TPIU フレーム同期パケット	913
30.17.5	同期フレームパケットの送信	913
30.17.6	同期モード	913
30.17.7	非同期モード	914
30.17.8	内部の TRACECLKIN 接続 STM32L1xxxx	914
30.17.9	TPIU レジスタ	915
30.17.10	設定例	915
30.18	DBG レジスタマップ	916
31	デバイス電子署名	917
31.1	メモリサイズレジスタ	917
31.1.1	フラッシュサイズレジスタ	917
31.2	ユニークデバイス ID レジスタ (96 ビット)	918
32	改版履歴	923

表一覧

表 1.	製品カテゴリとメモリサイズの概要	40
表 2.	STM32L100xx 製品カテゴリ	40
表 3.	STM32L15xxx 製品カテゴリ	40
表 4.	STM32L162xx 製品カテゴリ	41
表 5.	レジスタ境界アドレス	47
表 6.	ブートモード	51
表 7.	メモリマッピング対ブートモード／物理リマップ	52
表 8.	フラッシュメモリモジュール構成 (Cat.1 と Cat.2 デバイス)	54
表 9.	フラッシュメモリモジュール構成 (Cat.3 デバイス)	55
表 10.	フラッシュメモリモジュール構成 (Cat.4 デバイス)	56
表 11.	フラッシュメモリモジュール構成 (Cat.5 デバイス)	57
表 12.	CPU クロック (HCLK) 周波数に対応したウェイトステート数 (WS)	58
表 13.	FLASH_ACR に対する許容設定	61
表 14.	データ EEPROM 書き込み時間	69
表 15.	リードホワイルライトの要約	71
表 16.	禁止命令	72
表 17.	オプションバイト構成	73
表 18.	オプションバイトの説明	74
表 19.	書き込み／消去機能 (Cat.1、Cat.2、Cat.3 デバイス)	77
表 20.	書き込み／消去機能 (Cat.4 と Cat.5 デバイス)	78
表 21.	RDP とその相補に従うフラッシュメモリモジュール保護機能	80
表 22.	割り込み	82
表 23.	レジスタマップとリセット値	92
表 24.	CRC 計算ユニットのレジスタ・マップとリセット値	97
表 25.	性能対 VCORE 範囲	103
表 26.	低消費電力モードの一覧	112
表 27.	Sleep-now	115
表 28.	Sleep-on-exit	115
表 29.	Sleep-now	117
表 30.	Sleep-on-exit	117
表 31.	Stop モード	119
表 32.	Standby モード	120
表 33.	PWR - レジスタマップとリセット値	128
表 34.	システムクロックソース周波数	137
表 35.	RCC レジスタマップとリセット値	173
表 36.	ポートビット設定表	177
表 37.	フレキシブルな SWJ-DP ピン配置	179
表 38.	RTC_AF1 ピン	186
表 39.	GPIO レジスタマップとリセット値	194
表 40.	I/O グループと選択	202
表 41.	入力キャプチャの割り当て	205
表 42.	タイマ選択	206
表 43.	入力キャプチャの選択	206
表 44.	RI レジスタマップとリセット値	222
表 45.	SYSCFG レジスタマップとリセット値	229
表 46.	アキュイジションスイッチングシーケンスの概要	233
表 47.	チャンネルとサンプリングコンデンサ I/O の設定概要	234
表 48.	ベクタテーブル (Cat.1 および Cat.2 デバイス)	238
表 49.	ベクタテーブル (Cat.3 デバイス)	240



表 50.	ベクタテーブル (Cat.4 および Cat.5 デバイス)	242
表 51.	外部割り込み/イベントコントローラレジスタマップとリセット値	252
表 52.	プログラミング可能なデータ幅とエンディアン動作 (ビット PINC = MINC = 1 のとき)	259
表 53.	DMA 割り込みリクエスト	260
表 54.	各チャネル に対する DMA1 リクエストの一覧	263
表 55.	各チャネルに対する DMA2 リクエストの一覧	264
表 56.	DMA レジスタマップとリセット値	270
表 57.	ADC ピン	275
表 58.	アナログウォッチドッグチャネル選択	281
表 59.	トリガエッジ検出の設定	286
表 60.	レギュラチャネルの外部 トリガ	286
表 61.	インジェクトチャネルの外部 トリガ	287
表 62.	ADC 割り込み	298
表 63.	ADC グローバルレジスタマップ	318
表 64.	ADC のレジスタマップおよびリセット値	318
表 65.	ADC のレジスタマップおよびリセット値 (共通レジスタ)	320
表 66.	DAC のピン	322
表 67.	外部トリガ	325
表 68.	DAC レジスタマップ	344
表 69.	低消費電力モードでのコンパレータ動作	350
表 70.	COMP レジスタマップとリセット値	354
表 71.	動作モードと較正	358
表 72.	OPAMP レジスタマップ	364
表 73.	フレームレートの計算例	370
表 74.	ブリック周波数	378
表 75.	GPIO ピン への V _{LCDrail} の接続	380
表 76.	再配置機能	382
表 77.	LCD レジスタマップとリセット値	392
表 78.	カウント方向とエンコーダ信号	421
表 79.	TIMx 内部トリガ接続	436
表 80.	標準 OCx チャネルの出力制御ビット	446
表 81.	TIMx レジスタマップとリセット値	453
表 82.	TIMx 内部トリガ接続	481
表 83.	標準 OCx チャネルの出力制御ビット	490
表 84.	TIM9 のレジスタマップとリセット値	493
表 85.	標準 OCx チャネルの出力制御ビット	503
表 86.	TIM10/11 のレジスタマップとリセット値	508
表 87.	TIM6 および TIM7 のレジスタマップとリセット値	522
表 88.	低電力モードが RTC に与える影響	540
表 89.	割り込み制御ビット	541
表 90.	RTC レジスタマップとリセット値	565
表 91.	37 kHz での最小 / 最大 IWDG タイムアウト期間 (LSI)	568
表 92.	IWDG レジスタマップとリセット値	572
表 93.	タイムアウトの最小値と最大値 @32 MHz (f _{PCLK1})	575
表 94.	WWDG レジスタマップとリセット値	579
表 95.	処理時間 (クロックサイクル単位)	596
表 96.	AES 割り込みリクエスト	597
表 97.	AES レジスタマップ	606
表 98.	ダブルバッファリングのバッファフラグ定義	617
表 99.	バルクダブルバッファリングメモリのバッファ使用	618
表 100.	アイソクロナスメモリバッファ	619
表 101.	レジャーイベント検出	621

表 102.	受信ステータスエンコーディング	632
表 103.	エンドポイントタイプエンコーディング	633
表 104.	エンドポイントの種類の意味	633
表 105.	送信ステータスのエンコーディング	633
表 106.	割り当てられるバッファメモリの定義	637
表 107.	USB レジスタマップとリセット値	637
表 108.	NOR/PSRAM バンク選択	642
表 109.	外部メモリアドレス	642
表 110.	プログラマブル NOR/PSRAM アクセスパラメータ	643
表 111.	非マルチプレクス I/O NOR 型フラッシュ	644
表 112.	マルチプレクス I/O NOR 型フラッシュ	644
表 113.	非マルチプレクス I/O、PSRAM/SRAM	645
表 114.	マルチプレクス I/O PSRAM	645
表 115.	NOR 型フラッシュ／PSRAM コントローラ：サポートされるメモリとトランザクションの例	646
表 116.	FSMC_BCRx ビットフィールド	649
表 117.	FSMC_BTRx ビットフィールド	650
表 118.	FSMC_BCRx ビットフィールド	651
表 119.	FSMC_BTRx ビットフィールド	652
表 120.	FSMC_BWTRx ビットフィールド	652
表 121.	FSMC_BCRx ビットフィールド	654
表 122.	FSMC_BTRx ビットフィールド	655
表 123.	FSMC_BWTRx ビットフィールド	655
表 124.	FSMC_BCRx ビットフィールド	657
表 125.	FSMC_BTRx ビットフィールド	657
表 126.	FSMC_BWTRx ビットフィールド	658
表 127.	FSMC_BCRx ビットフィールド	659
表 128.	FSMC_BTRx ビットフィールド	660
表 129.	FSMC_BWTRx ビットフィールド	660
表 130.	FSMC_BCRx ビットフィールド	662
表 131.	FSMC_BTRx ビットフィールド	662
表 132.	FSMC_BCRx ビットフィールド	667
表 133.	FSMC_BTRx ビットフィールド	668
表 134.	FSMC_BCRx ビットフィールド	669
表 135.	FSMC_BTRx ビットフィールド	670
表 136.	FSMC レジスタマップ	678
表 137.	SMBus と I2C	693
表 138.	I2C 割り込みリクエスト	698
表 139.	I2C レジスタマップとリセット値	715
表 140.	サンプリングされたデータからのノイズ検出	729
表 141.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 8 \text{ MHz}$ または $f_{PCLK} = 12 \text{ MHz}$)、 16 倍のオーバーサンプリング	732
表 142.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 8 \text{ MHz}$ または $f_{PCLK} = 12 \text{ MHz}$)、 8 倍のオーバーサンプリング	733
表 143.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 24 \text{ MHz}$)、 16 倍のオーバーサンプリング	733
表 144.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 24 \text{ MHz}$)、 8 倍のオーバーサンプリング	734
表 145.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、 16 倍のオーバーサンプリング	735
表 146.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、 8 倍のオーバーサンプリング	735

表 147.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、 16 倍のオーバーサンプリング.....	736
表 148.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、 8 倍のオーバーサンプリング.....	737
表 149.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、 8 倍のオーバーサンプリング.....	737
表 150.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、 16 倍のオーバーサンプリング.....	738
表 151.	プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、 8 倍のオーバーサンプリング.....	739
表 152.	USART レシーバの許容誤差 (DIV Fraction = 0 の場合).....	740
表 153.	USART レシーバの許容誤差 (DIV Fraction が 0 でない場合).....	740
表 154.	フレームフォーマット.....	742
表 155.	USART 割り込みリクエスト.....	756
表 156.	USART モード設定.....	757
表 157.	USART レジスタマップとリセット値.....	767
表 158.	SPI 割り込みリクエスト.....	795
表 159.	標準の 8 MHz HSE を使用したオーディオ周波数精度 (Cat.3、Cat.4、Cat.5 デバイスの場合).....	806
表 160.	I ² S 割り込みリクエスト.....	812
表 161.	SPI レジスタマップとリセット値.....	823
表 162.	SDIO I/O 定義.....	829
表 163.	コマンドフォーマット.....	833
表 164.	ショートレスポンスフォーマット.....	833
表 165.	ロングレスポンスフォーマット.....	834
表 166.	コマンドパスステータスフラグ.....	834
表 167.	データトークンフォーマット.....	837
表 168.	送信 FIFO ステータスフラグ.....	838
表 169.	受信 FIFO ステータスフラグ.....	839
表 170.	カードステータス.....	849
表 171.	SD ステータス.....	851
表 172.	速度クラスコードフィールド.....	852
表 173.	移動パフォーマンスフィールド.....	853
表 174.	AU_SIZE フィールド.....	853
表 175.	最大 AU サイズ.....	853
表 176.	消去サイズフィールド.....	854
表 177.	消去タイムアウトフィールド.....	854
表 178.	消去オフセットフィールド.....	854
表 179.	ブロック指向書き込みコマンド.....	857
表 180.	ブロック指向書き込み保護コマンド.....	857
表 181.	消去コマンド.....	858
表 182.	I/O モードコマンド.....	858
表 183.	カードのロック.....	858
表 184.	アプリケーション固有コマンド.....	859
表 185.	R1 レスポンス.....	859
表 186.	R2 レスポンス.....	860
表 187.	R3 レスポンス.....	860
表 188.	R4 レスポンス.....	861
表 189.	R4b レスポンス.....	861
表 190.	R5 レスポンス.....	862
表 191.	R6 レスポンス.....	862
表 192.	レスポンスタイプと SDIO_RESPx レジスタ.....	869
表 193.	SDIO レジスタマップ.....	880

表 194.	SWJ デバッグポートピン	886
表 195.	フレキシブルな SWJ-DP ピン配置	886
表 196.	JTAG デバッグポートのデータレジスタ	892
表 197.	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	893
表 198.	パケットリクエスト (8 ビット)	894
表 199.	ACK 応答 (3 ビット)	894
表 200.	DATA 転送 (33 ビット)	894
表 201.	SW-DP レジスタ	896
表 202.	Cortex®-M3 AHB-AP レジスタ	897
表 203.	コアデバッグレジスタ	898
表 204.	主な ITM レジスタ	901
表 205.	主な ETM レジスタ	903
表 206.	非同期 TRACE ピンの割り当て	911
表 207.	同期 TRACE ピンの割り当て	911
表 208.	フレキシブルな TRACE ピンの割り当て	912
表 209.	重要な TPIU レジスタ	915
表 210.	DBG レジスタマップとリセット値	916
表 211.	改版履歴	923

図一覧

図 1.	システムアーキテクチャ (Cat.1 と Cat.2 デバイス).....	42
図 2.	システムアーキテクチャ (Cat.3 デバイス).....	43
図 3.	システムアーキテクチャ (Cat.4 デバイス).....	44
図 4.	システムアーキテクチャ (Cat.5 デバイス).....	45
図 5.	連続 32 ビット命令の実行.....	60
図 6.	RDP レベル.....	80
図 7.	CRC 計算ユニットのブロック図.....	95
図 8.	電源の概要.....	100
図 9.	STM32L1xxxx 性能対 VDD と VCore の範囲.....	104
図 10.	電源供給スーパバイザ.....	107
図 11.	パワーオンリセット/パワーダウンリセット波形.....	108
図 12.	BOR の閾値.....	110
図 13.	PVD の閾値.....	110
図 14.	簡略化されたリセット回路図.....	130
図 15.	クロックツリー.....	132
図 16.	HSE/LSE クロックソース.....	133
図 17.	TIM9/TIM10/TIM11 チャンネル 1 入力キャプチャを使用する周波数測定.....	140
図 18.	標準 I/O ポートビットの基本構造.....	176
図 19.	5 V 耐性 I/O ポートビットの基本構造.....	176
図 20.	オルタネート機能の選択.....	180
図 21.	入力フローティング/プルアップ/プルダウン設定.....	182
図 22.	出力設定.....	183
図 23.	オルタネート機能設定.....	184
図 24.	ハインピーダンスアナログ設定.....	185
図 25.	Cat.1 および Cat.2 製品のルーティングインタフェース (RI) ブロック図.....	198
図 26.	Cat.3 製品のルーティングインタフェース (RI) ブロック図.....	199
図 27.	Cat.4 および Cat.5 製品のルーティングインタフェース (RI) ブロック図.....	200
図 28.	内部基準電圧の出力.....	206
図 29.	表面電荷移動アナログ I/O グループ構造.....	232
図 30.	サンプリングコンデンサ電荷の概要.....	233
図 31.	タイマモードアクイジションのロジック.....	235
図 32.	外部割り込み/イベントコントローラのブロック図.....	245
図 33.	外部割り込み/イベントの GPIO マッピング.....	247
図 34.	Cat.1 および Cat.2 STM32L1xxxx デバイスの DMA ブロック図.....	254
図 35.	Cat.3STM32L1xxxx デバイスの DMA ブロック図.....	255
図 36.	およびデバイスの Cat.4 DMA ブロック図 Cat.5STM32L1xxxx.....	256
図 37.	DMA1 リクエストのマッピング.....	262
図 38.	DMA2 リクエストのマッピング.....	264
図 39.	ADC ブロック図 (Cat.1 および Cat.2 デバイス).....	274
図 40.	ADC ブロック図 (Cat.3、Cat.4、Cat.5 デバイス).....	276
図 41.	タイミング図 (通常モード、PDI=0).....	280
図 42.	アナログウォッチドッグの保護領域.....	281
図 43.	インジェクト変換遅延.....	283
図 44.	右詰め 12 ビットデータ.....	284
図 45.	左詰め 12 ビットデータ.....	285
図 46.	左詰め 6 ビットデータ.....	285
図 47.	ADC フリーズモード.....	289
図 48.	連続レギュラ変換 (遅延あり).....	290
図 49.	連続変換 (各変換の間の遅延あり).....	291

図 50.	自動パワーダウン制御：例 1.....	292
図 51.	自動パワーダウン制御：例 2.....	292
図 52.	自動パワーダウン制御：例 3.....	293
図 53.	温度センサと VREFINT チャンネルのブロック図.....	295
図 54.	ADC フラグと割り込み.....	297
図 55.	DAC チャンネルブロック図.....	322
図 56.	シングル DAC チャンネルモードのデータレジスタ.....	324
図 57.	デュアル DAC チャンネルモードのデータレジスタ.....	324
図 58.	トリガ無効 (TEN = 0) 時の変換タイミング図.....	325
図 59.	DAC LFSR レジスタ計算アルゴリズム.....	326
図 60.	LFSR 波形生成による DAC 変換 (SW トリガ有効).....	327
図 61.	DAC 三角波生成.....	327
図 62.	三角波生成による DAC 変換 (SW トリガ有効).....	328
図 63.	COMP1 の接続 (Cat.1 と Cat.2 デバイス).....	346
図 64.	COMP1 の接続 (Cat.3、Cat.4、Cat.5 デバイス).....	347
図 65.	COMP2 の接続 (Cat.1 と Cat.2 デバイス).....	348
図 66.	COMP2 の接続 (Cat.3、Cat.4、Cat.5 デバイス).....	348
図 67.	COMP2 出力のリダイレクト.....	349
図 68.	ウィンドウモードのコンパレータ.....	350
図 69.	OPAMP1 の信号ルーティング.....	356
図 70.	OPAMP2 の信号ルーティング.....	356
図 71.	OPAMP3 信号ルーティング (Cat.4 デバイスの場合).....	357
図 72.	LCD コントローラのブロック図.....	368
図 73.	1/3 バイアス、1/4 デューティ.....	371
図 74.	スタティックデューティ.....	372
図 75.	スタティックデューティ.....	373
図 76.	1/2 デューティ、1/2 バイアス.....	374
図 77.	1/3 デューティ、1/3 バイアス.....	375
図 78.	1/4 デューティ、1/3 バイアス.....	376
図 79.	1/8 デューティ、1/4 バイアス.....	377
図 80.	LCD 電圧制御.....	379
図 81.	デッドタイム.....	380
図 82.	SEG/COM マルチプレクス機能の例.....	384
図 83.	フローチャートの例.....	385
図 84.	汎用タイマのブロック図.....	396
図 85.	プリスケアラ分周比が 1 から 2 に変化する時のカウンタのタイミング図.....	398
図 86.	プリスケアラ分周比が 1 から 4 に変化する時のカウンタのタイミング図.....	398
図 87.	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.....	399
図 88.	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.....	400
図 89.	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.....	400
図 90.	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.....	400
図 91.	ARPE=0 (TIMx_ARR はプリロードされない) のときの更新イベントのカウンタタイミング図.....	401
図 92.	ARPE=1 (TIMx_ARR はプリロードされる) のときの更新イベントのカウンタタイミング図.....	401
図 93.	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.....	402
図 94.	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.....	403
図 95.	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.....	403
図 96.	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.....	403
図 97.	更新イベントのカウンタのタイミング図.....	404
図 98.	内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタタイミング図.....	405
図 99.	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.....	405
図 100.	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.....	406
図 101.	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.....	406

図 102.	ARPE=1(カウンタアンダーフロー) のときの更新イベント時のカウンタタイミング図	406
図 103.	ARPE=1 (カウンタオーバーフロー) のときの更新イベント時のカウンタのタイミング図	407
図 104.	内部クロック分周比 1 の場合の、通常モードの制御回路	407
図 105.	TI2 外部クロックの接続例	408
図 106.	外部クロックモード 1 の制御回路	409
図 107.	外部トリガ入力ブロック	409
図 108.	外部クロックモード 2 の制御回路	410
図 109.	キャプチャ/比較チャネル (例: チャネル 1 入力ステージ)	410
図 110.	キャプチャ/比較チャネル 1 メイン回路	411
図 111.	キャプチャ/比較チャネル (チャネル 1) の出力ステージ	411
図 112.	PWM 入力モードのタイミング	413
図 113.	出力比較モード、OC1 のトグル	415
図 114.	エッジアライン PWM 波形 (ARR=8)	416
図 115.	センターアライン PWM 波形 (ARR=8)	417
図 116.	ワンパルスモードの例	418
図 117.	TIMx OCxREF のクリア	420
図 118.	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	421
図 119.	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	422
図 120.	リセットモードの制御回路	423
図 121.	ゲートモードの制御回路	424
図 122.	トリガモードの制御回路	424
図 123.	外部クロックモード 2 + トリガモードの制御回路	425
図 124.	マスタ/スレーブタイマの例	426
図 125.	TIM3 の OC1REF による TIM2 のゲート操作	427
図 126.	TIM3 の有効化による TIM2 のゲート操作	428
図 127.	TIM3 の更新による TIM2 のトリガ	429
図 128.	TIM3 の有効化による TIM2 のトリガ	430
図 129.	TIM3 の TI1 入力による TIM3 および TIM2 のトリガ	431
図 130.	汎用タイマのブロック図 (TIM9)	456
図 131.	汎用タイマのブロック図 (TIM10)	457
図 132.	汎用タイマのブロック図 (TIM11)	458
図 133.	プリスケアラプリスケアラ比が 1 から 2 に変化したときのカウンタのタイミング図	460
図 134.	プリスケアラプリスケアラ比が 1 から 4 に変化したときのカウンタのタイミング図	460
図 135.	カウンタタイミング図、内部クロック 1 分周	461
図 136.	カウンタタイミング図、内部クロック 2 分周	461
図 137.	カウンタタイミング図、内部クロック 4 分周	462
図 138.	内部クロックプリスケアラ比が N の場合のカウンタのタイミング図	462
図 139.	ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のカウンタのタイミング図	462
図 140.	カウンタタイミング図、ARPE=1 時の更新イベント (TIMx_ARR preloaded)	463
図 141.	内部クロックプリスケアラ比 1 の場合の、通常モードの制御回路	464
図 142.	TI2 外部クロックの接続例	464
図 143.	外部クロックモード 1 の制御回路	465
図 144.	キャプチャ/比較チャネル (例: チャネル 1 入力ステージ)	465
図 145.	キャプチャ/比較チャネル 1 メイン回路	466
図 146.	キャプチャ/比較チャネル (チャネル 1) の出力ステージ	466
図 147.	PWM 入力モードタイミング	468
図 148.	出力比較モード、OC1 のトグル	470
図 149.	エッジアライン PWM 波形 (ARR=8)	471
図 150.	ワンパルスモードの例	472
図 151.	リセットモードの制御回路	474
図 152.	ゲートモードの制御回路	474
図 153.	トリガモードの制御回路	475

図 154.	基本タイマブロック図	511
図 155.	プリスケアラ比が 1 から 2 に変化したときのカウンタのタイミング図	512
図 156.	プリスケアラ比が 1 から 4 に変化したときのカウンタのタイミング図	513
図 157.	内部クロック分周比が 1 の場合のカウンタタイミング図	514
図 158.	カウンタタイミング図、内部クロック 2 分周	514
図 159.	カウンタタイミング図、内部クロック 4 分周	514
図 160.	内部クロックプリスケアラ比が N の場合のカウンタのタイミング図	515
図 161.	ARPE=0 (TIMx_ARR はプリロードされない) のときの更新イベント時のカウンタの タイミング図	515
図 162.	カウンタタイミング図、ARPE=1 時の更新イベント (TIMx_ARR preloaded)	516
図 163.	内部クロックプリスケアラ比 1 の場合の、通常モードの制御回路	516
図 164.	RTC のブロック図 (Cat.1 製品)	525
図 165.	RTC のブロック図 (Cat.2、Cat.3、Cat.4 および Cat.5 製品)	525
図 166.	独立型ウォッチドッグのブロック図	568
図 167.	ウォッチドッグのブロック図	573
図 168.	ウィンドウ型ウォッチドッグのタイミング図	575
図 169.	ブロック図	582
図 170.	ECB 暗号化モード	584
図 171.	ECB 復号化モード	585
図 172.	CBC モードの暗号化	586
図 173.	CBC モードの復号化	586
図 174.	サスペンドモードの管理例	587
図 175.	CTR モードの暗号化	588
図 176.	CTR モードの復号化	589
図 177.	32 ビットカウンタ + ノンスの構成	589
図 178.	データタイプに準じた 128 ビットブロックの構成	591
図 179.	データタイプに準じた 128 ビットブロックの構成 (続き)	592
図 180.	モード 1 : 暗号化	592
図 181.	モード 2 : キー派生	593
図 182.	モード 3 : 復号化	594
図 183.	モード 4 : キー派生と復号化	594
図 184.	入力フェーズ中の DMA リクエストとデータ転送 (AES_IN)	595
図 185.	出力フェーズ中の DMA リクエスト (AES_OUT)	596
図 186.	USB peripheral のブロック図	608
図 187.	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	613
図 188.	FSMC のブロック図	640
図 189.	FSMC メモリバンク	642
図 190.	モード 1 読み出しアクセス	648
図 191.	モード 1 書き込みアクセス	649
図 192.	モード A 読み出しアクセス	650
図 193.	モード A 書き込みアクセス	651
図 194.	モード 2 およびモード B 読み出しアクセス	653
図 195.	モード 2 書き込みアクセス	653
図 196.	モード B 書き込みアクセス	654
図 197.	モード C 読み出しアクセス	656
図 198.	モード C 書き込みアクセス	656
図 199.	モード D 読み出しアクセス	658
図 200.	モード D 書き込みアクセス	659
図 201.	マルチプレクス読み出しアクセス	661
図 202.	マルチプレクス書き込みアクセス	661
図 203.	読み出しアクセス中の非同期ウェイト	664
図 204.	書き込みアクセス中の非同期ウェイト	664

図 205.	ウェイト設定	666
図 206.	同期マルチプレクス読み出しモード NOR、PSRAM (CRAM)	667
図 207.	同期マルチプレクス書き込みモード NOR、PSRAM (CRAM)	669
図 208.	I2C バスプロトコル	681
図 209.	I2C ブロック図	682
図 210.	スレーブトランスミッタの転送シーケンス図	684
図 211.	スレーブレシーバの転送シーケンス図	685
図 212.	マスタトランスミッタの転送シーケンス図	688
図 213.	マスタレシーバの転送シーケンス図	690
図 214.	I2C 割り込みマッピング図	699
図 215.	USART のブロック図	720
図 216.	ワード長のプログラミング	721
図 217.	設定可能なストップビット	723
図 218.	送信時の TC/TXE の動作	724
図 219.	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	725
図 220.	データサンプリング (16 倍のオーバーサンプリング)	728
図 221.	データサンプリング (8 倍のオーバーサンプリング)	729
図 222.	アイドルライン検出を使用したミュートモード	741
図 223.	アドレスマーク検出を使用したミュートモード	742
図 224.	LIN モードでのブレーク検出 (11 ビットブレーク長、LBDL=1)	744
図 225.	LIN モードでのブレーク検出とフレーミングエラー検出	745
図 226.	USART の同期送信の例	746
図 227.	USART データクロックタイミング図 (M=0)	746
図 228.	USART データクロックタイミング図 (M=1)	747
図 229.	RX データのセットアップ/ホールド時間	747
図 230.	ISO 7816-3 非同期プロトコル	748
図 231.	1.5 ストップビットを使用したパリティエラー検出	749
図 232.	IrDA SIR ENDEC- ブロック図	751
図 233.	IrDA データ変調 (3/16) - 通常モード	751
図 234.	DMA を使用した送信	753
図 235.	DMA を使用した受信	754
図 236.	2 つの USART 間のハードウェアフロー制御	754
図 237.	RTS フロー制御	755
図 238.	CTS フロー制御	755
図 239.	USART 割り込みマッピング図	756
図 240.	SPI のブロック図	772
図 241.	シングルマスタ/シングルスレーバアプリケーション	773
図 242.	データクロックのタイミング図	775
図 243.	TI モード - スレーブモード、シングル送信	777
図 244.	TI モード - スレーブモード、連続送信	778
図 245.	TI モード - マスタモード、シングル転送	780
図 246.	TI モード - マスタモード、連続転送	780
図 247.	連続転送の場合におけるマスタ/全二重モード (BIDIMODE=0 および RXONLY=0) での TXE/RXNE/BSY の動作	784
図 248.	連続転送の場合におけるスレーブ/全二重モード (BIDIMODE=0 および RXONLY=0) での TXE/RXNE/BSY の動作	784
図 249.	連続転送の場合におけるマスタ送信専用モード (BIDIMODE=0 および RXONLY=0) での TXE/BSY の動作	785
図 250.	連続転送の場合におけるスレーブ送信専用モード (BIDIMODE=0 および RXONLY=0) での TXE/BSY の動作	786
図 251.	連続転送の場合における受信専用モード (BIDIROMODE=0 および RXONLY=1) での RXNE の動作	787

図 252.	不連続転送の場合における送信時 (BIDIRMODE=0 および RXONLY=0) の TXE/BSY の動作	788
図 253.	DMA を使用した送信	793
図 254.	DMA を使用した受信	793
図 255.	TI モードフレームフォーマットエラーの検出	795
図 256.	I ² S のブロック図	796
図 257.	I ² S フィリップスプロトコルの波形 (16/32 ビットフル精度、CPOL = 0)	798
図 258.	I ² S フィリップス標準の波形 (24 ビットフレーム、CPOL = 0)	798
図 259.	0x8EAA33 の送信	799
図 260.	0x8EAA33 の受信	799
図 261.	I ² S フィリップス標準 (32 ビットパケットフレームへ 16 ビット拡張、CPOL = 0)	799
図 262.	例	800
図 263.	MSB 詰め 16 ビットまたは 32 ビットフル精度長、CPOL = 0	800
図 264.	MSB 詰め 24 ビットフレーム長、CPOL = 0	800
図 265.	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	801
図 266.	LSB 詰め 16 ビットまたは 32 ビットのフル精度、CPOL = 0	801
図 267.	LSB 詰め 24 ビットフレーム長、CPOL = 0	802
図 268.	0x3478AE を送信するために必要な動作	802
図 269.	0x3478AE の受信に必要な動作	802
図 270.	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	803
図 271.	32 ビットパケットフレームに拡張された LSB 詰め 16 ビットの例	803
図 272.	PCM 標準の波形 (16 ビット)	804
図 273.	PCM 標準の波形 (32 ビットパケットフレームに拡張された 16 ビット)	804
図 274.	オーディオサンプリング周波数の定義	805
図 275.	I ² S クロックジェネレータのアーキテクチャ	805
図 276.	SDIO "no response" と "no data" 操作	826
図 277.	SDIO (複数) ブロック読み出し操作	826
図 278.	SDIO (複数) ブロック書き込み操作	827
図 279.	SDIO 順次読み出し操作	827
図 280.	SDIO 順次書き込み操作	827
図 281.	SDIO ブロック図	828
図 282.	SDIO アダプタ	829
図 283.	制御ユニット	830
図 284.	SDIO アダプタコマンドパス	831
図 285.	コマンドパスステートマシン (CPSM)	832
図 286.	SDIO コマンド転送	832
図 287.	データパス	835
図 288.	データパスステートマシン (DPSM)	836
図 289.	STM32 MCU と Cortex®-M3- レベルのブロック図デバッグサポート	883
図 290.	SWJ デバッグポート	885
図 291.	JTAG TAP 接続	889
図 292.	TPIU ブロック図	910

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、ソフトウェアから読み出し／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。このビットに“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。
読み出し専用書き込みトリガ (rt_w)	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。“0”または“1”を書き込むと、イベントがトリガされますが、ビットの値は変化しません。
反転 (t)	ソフトウェアからこのビットへ“1”を書き込んで、トグルさせることができます。“0”を書き込んでも、影響ありません。
予約済み (Res.)	予約済みのビットであり、リセット値のままにしておかなければなりません。

1.2 使用可能なペリフェラル

すべての STM32L1xxxx 販売タイプで使用可能なペリフェラルと型番については、STM32L1xxxx データシートを参照してください。

1.3 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。

- **フラッシュメモリ** : 不揮発性メモリ、このドキュメントの適用範囲としてフラッシュメモリはプログラムメモリブロック、データ EEPROM ブロック、情報ブロックをカバーします。
- **FLITF** : メモリインタフェースはフラッシュメモリの読み出し動作、書き込み動作、消去動作を管理します。
- **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 (インアプリケーションプログラミング)** : IAP はユーザプログラムの動作中にマイクロコントローラのフラッシュメモリを再書き込みする機能です。
- **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に書き込みを行う機能です。
- **オプションバイト** : フラッシュメモリに格納された製品構成ビット。
- **OBL** : オプションバイトローダ。
- **AHB** : アドバンストハイパフォーマンスバス。
- **APB** : 高機能ペリフェラルバス。

1.4 製品カテゴリの定義

デバイスは、Cat.1、Cat.2、Cat.3、Cat.4、Cat.5の5つのカテゴリで構成されています。表 1 に、メモリサイズと製品カテゴリの概要を示します。

表 2、表 3、表 4 に、提供中の部品番号のリストをカテゴリごとに示します。

表 1. 製品カテゴリとメモリサイズの概要

メモリ集積度レベル	Cat.1	Cat.2	Cat.3	Cat.4	Cat.5
32KB	x	x	-	-	-
64KB	x	x	-	-	-
128KB	x	x	-	-	-
256KB	-	-	x	-	-
384KB	-	-	-	x	-
512KB	-	-	-	-	x

表 2. STM32L100xx 製品カテゴリ

RPNs STM32L100xx	Cat.1	Cat.2	Cat.3	Cat.4	Cat.5	例
STM32L100C6	x	-	-	-	-	-
STM32L100R8	x	-	-	-	-	-
STM32L100RB	x	-	-	-	-	-
STM32L100C6-A	-	x	-	-	-	STM32L100C6xxA
STM32L100R8-A	-	x	-	-	-	STM32L100R8xxA
STM32L100RB-A	-	x	-	-	-	STM32L100RBxxA
STM32L100RC	-	-	x	-	-	-

表 3. STM32L15xxx 製品カテゴリ

RPNs STM32L15xxx	Cat.1	Cat.2	Cat.3	Cat.4	Cat.5	例
STM32L15xx6	x	-	-	-	-	-
STM32L15xx8	x	-	-	-	-	-
STM32L15xxB	x	-	-	-	-	-
STM32L15xx6-A	-	x	-	-	-	STM32L151C6T6A
STM32L15xx8-A	-	x	-	-	-	STM32L151R8T6A
STM32L15xxB-A	-	x	-	-	-	STM32L151RBT6A
STM32L15xCC	-	-	x	-	-	-
STM32L15xUC	-	-	x	-	-	-
STM32L15xRC	-	-	x	-	-	-
STM32L15xRCY	-	-	x	-	-	STM32L151RCY6
STM32L15xRC-A	-	-	x	-	-	STM32L151RCT6A
STM32L15xVC	-	-	x	-	-	-

表 3. STM32L15xxx 製品カテゴリ (続き)

RPNs STM32L15xxx	Cat.1	Cat.2	Cat.3	Cat.4	Cat.5	例
STM32L15xVC-A	-	-	x	-	-	STM32L151VCT6A
STM32L15xQC	-	-	x	-	-	-
STM32L15xZC	-	-	x	-	-	-
STM32L15xRD	-	-	-	x	-	-
STM32L15xVD	-	-	-	x	-	-
STM32L15xQD	-	-	-	x	-	-
STM32L15xZD	-	-	-	x	-	-
STM32L15xxE	-	-	-	-	x	-

表 4. STM32L162xx 製品カテゴリ

RPNs STM32L162xx	Cat.1	Cat.2	Cat.3	Cat.4	Cat.5	例
STM32L162RC	-	-	x	-	-	-
STM32L162RC-A	-	-	x	-	-	STM32L162RCT6A
STM32L162VC	-	-	x	-	-	-
STM32L162VC-A	-	-	x	-	-	STM32L162VCT6A
STM32L162QC	-	-	x	-	-	-
STM32L162ZC	-	-	x	-	-	-
STM32L162RD	-	-	-	x	-	-
STM32L162VD	-	-	-	x	-	-
STM32L162QD	-	-	-	x	-	-
STM32L162ZD	-	-	-	x	-	-
STM32L162xE	-	-	-	-	x	-

2 システムアーキテクチャとメモリの概要

2.1 システムアーキテクチャ

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

- 最大 5 個の マスタ:
 - Cortex[®]-M3 I-bus、D-bus、S-bus
 - DMA1 と DMA2
- 最大 5 個の スレーブ:
 - 内部フラッシュメモリ ICode
 - 内部フラッシュメモリ DCode
 - 内部 SRAM
 - すべての APB 周辺回路を接続するための AHB/APBx (APB1 または APB2)
 - フレキシブルスタティックメモリコントローラ

これらは、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されています。

図 1. システムアーキテクチャ (Cat.1とCat.2 デバイス)

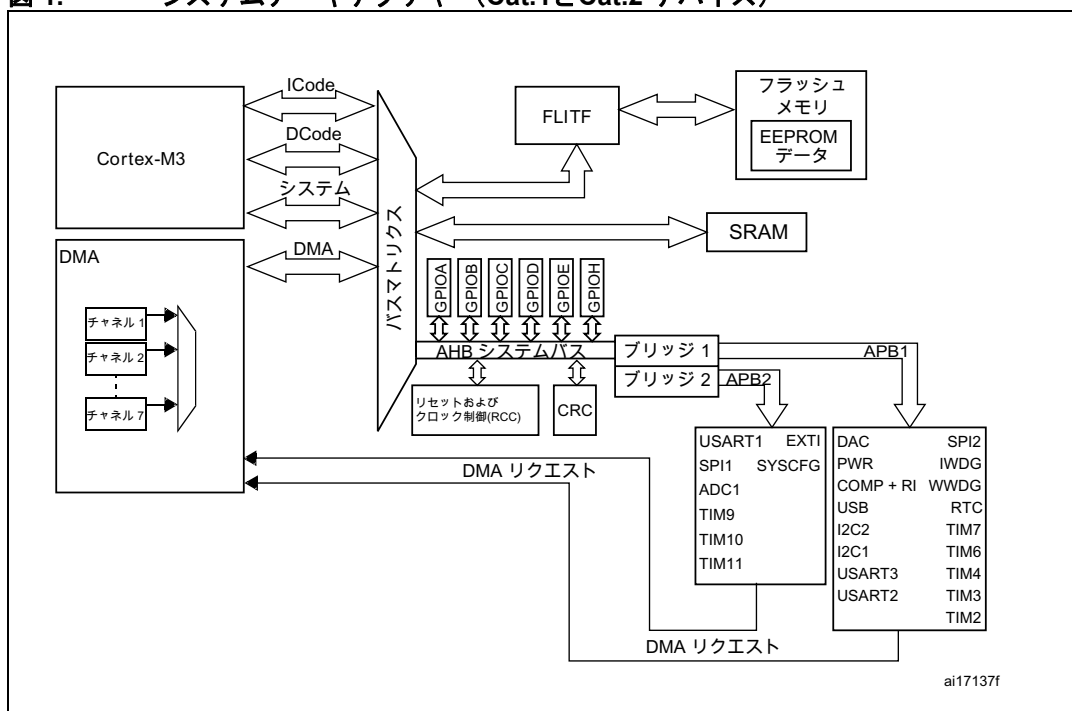


図 2. システムアーキテクチャ (Cat.3 デバイス)

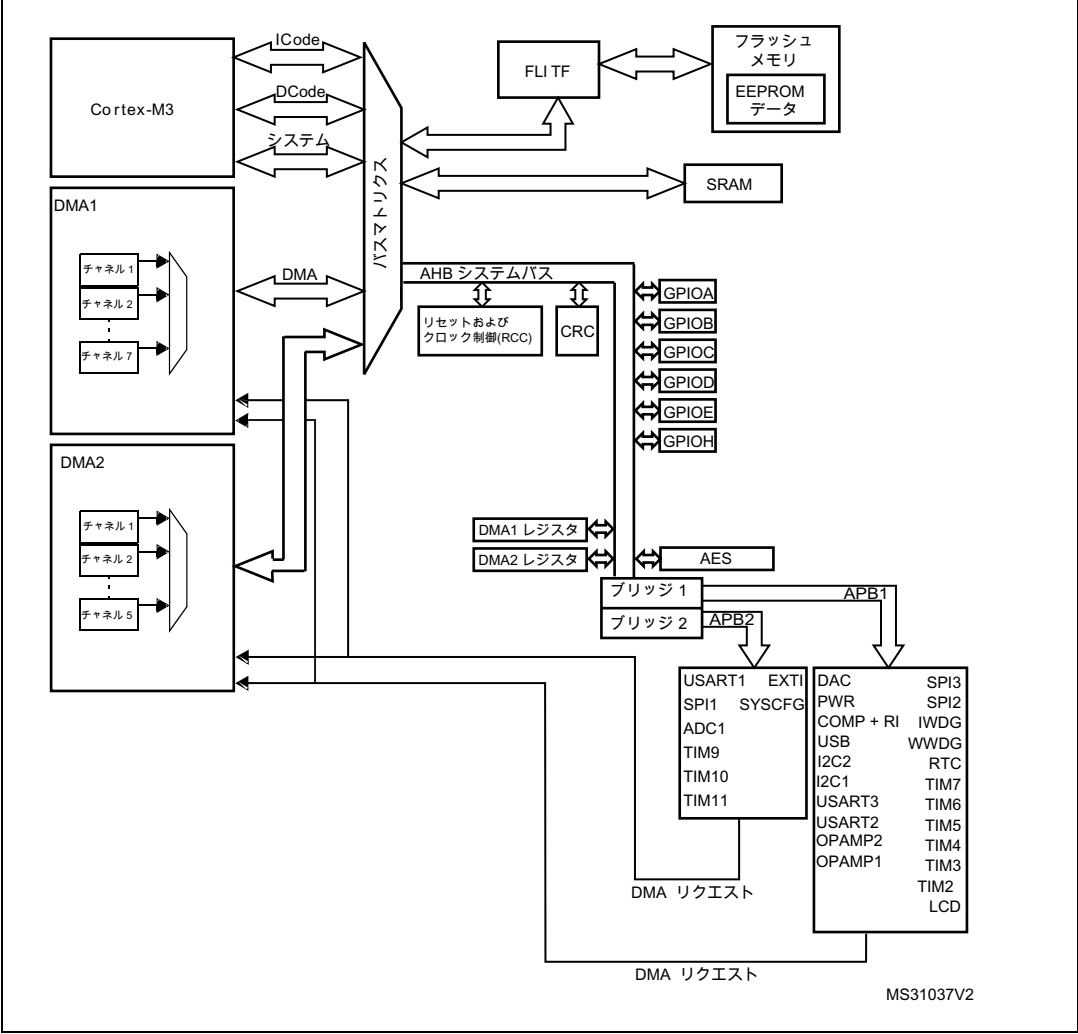


図 3. システムアーキテクチャ (Cat.4 デバイス)

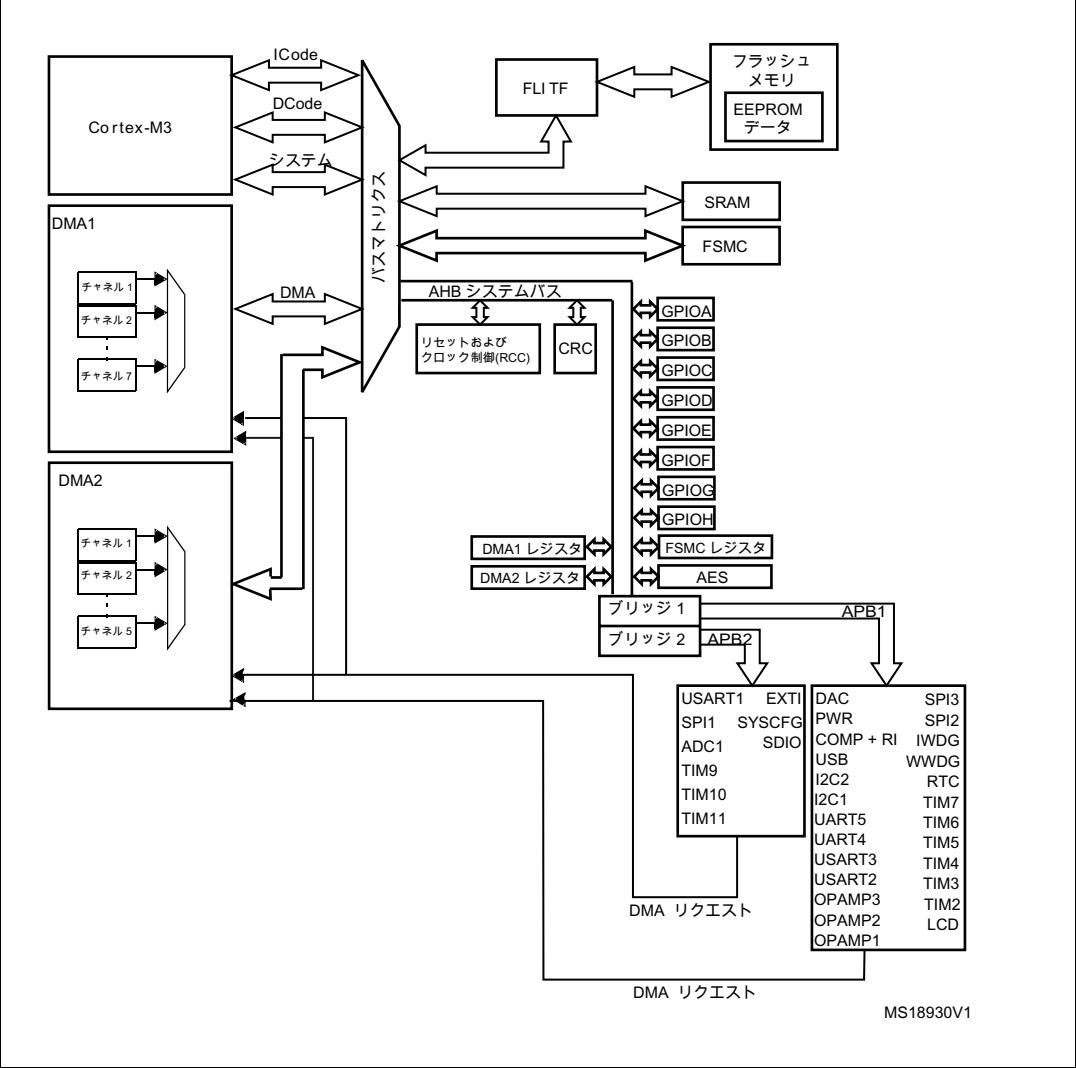
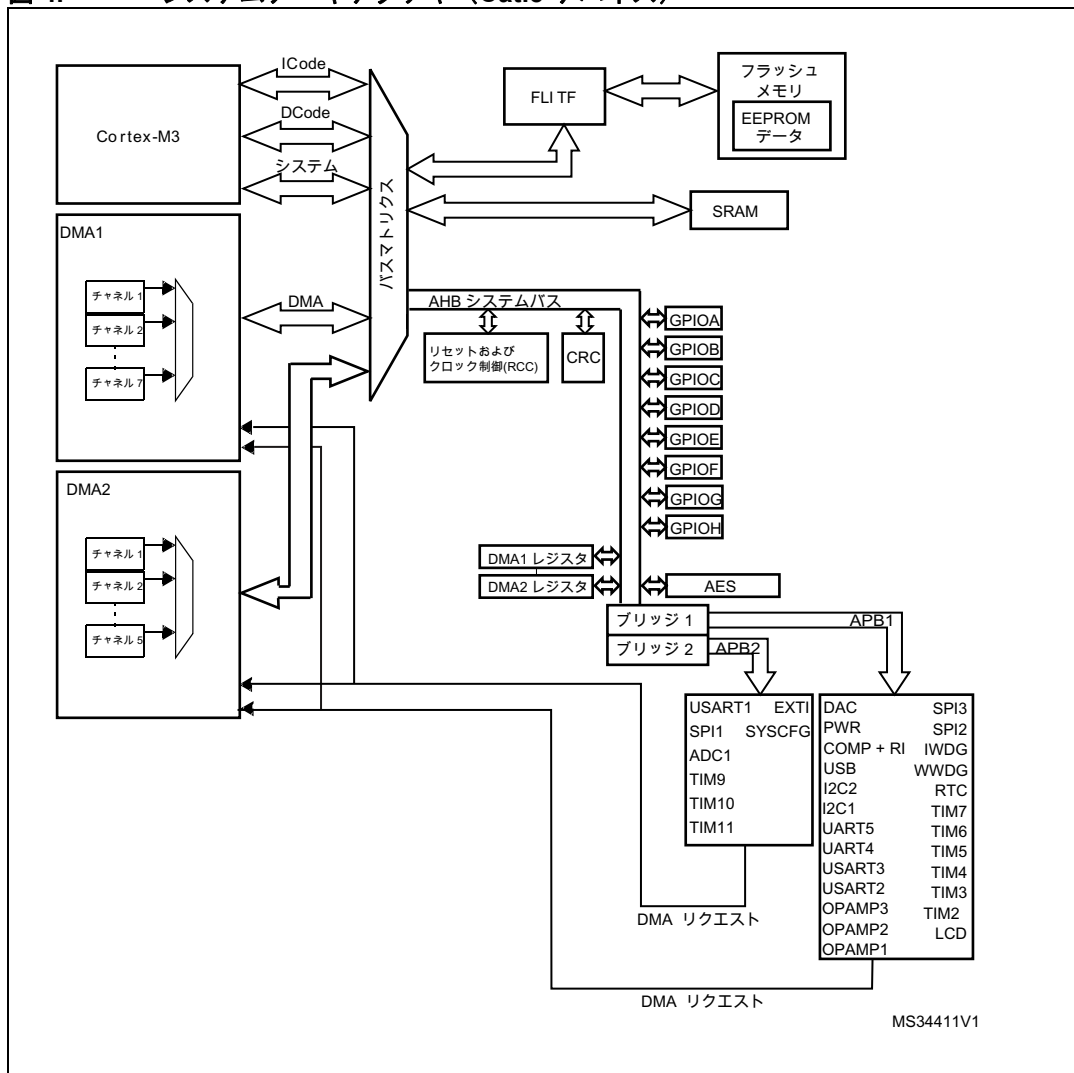


図 4. システムアーキテクチャ (Cat.5 デバイス)

**I-bus**

このバスによって、Cortex[®]-M3 コアの命令バスが、BusMatrix に接続されます。コアは、このバスを用いて命令をフェッチします。このバスの対象は、コードを格納するメモリ（内部フラッシュメモリまたは SRAM）です。

D-bus

このバスによって Cortex[®]-M3 のデータバスが BusMatrix に接続されます。このバスは、リテラルロードとデバッグアクセスのために、コアによって使用されます。このバスの対象は、コードまたはデータを格納するメモリ（内部フラッシュメモリまたは SRAM）です。

S-Bus

このバスによって、Cortex[®]-M3 コアの S-Bus が BusMatrix に接続されます。このバスを使用してペリフェラルまたは SRAM 内のデータにアクセスします。このバス上で、命令をフェッチすることもできます（ICode より低効率です）。このバスの対象は、内部 SRAM と AHB/APB ブリッジです。

DMA バス

このバスは、DMA の AHB マスタインタフェースをバスマトリックスに接続します。バスマトリックスは、CPU の D-bus と DMA から SRAM、フラッシュメモリ、およびペリフェラルへのアクセスを管理します。

バスマトリックス

バスマトリックスは、コア S-Bus と DMA マスタバスの間のアクセスアービトレーションを管理します。アービトレーションには、ラウンドロビン方式が使用されます。バスマトリックスは、5 個のマスタ (ICode、DCode、S-Bus、DMA1 バス、DMA2 バス) および 5 個のスレーブ (フラッシュ ICode インタフェース、フラッシュ DCode インタフェース、SRAM、FSMC、AHB2APB ブリッジ) から構成されています。

AHB ペリフェラルは、バスマトリックスを経由して S-Bus に接続され、DMA アクセスを可能にします。

AHB/APB ブリッジ (APB)

2 個の AHB/APB ブリッジは、AHB と 2 個の APB バスの間の、完全同期式の接続を提供しています。2 本の APB バスは、フルスピードで動作します (最大 32 MHz)。

AHB と APB のペリフェラルのアドレスマッピングについては、[表 5 \(47 ページ\)](#) 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化されます (SRAM とフラッシュのインタフェースは除きます)。ペリフェラルを使用する前に、RCC_AHBENR レジスタ、RCC_APB1ENR レジスタまたは RCC_APB2ENR レジスタでクロックを有効化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

プログラムメモリ、データメモリ、レジスタ、および入出力ポートは、同じリニア 4 G バイトアドレス空間内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。

ペリフェラルレジスタの詳細なマッピングについては、該当するセクションを参照してください。

アドレス可能なメモリ空間は、それぞれ 512 Mバイトの 8 つのメインブロックに分けられています。

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。STM32L1xxx データシートのメモリマップ図を参照してください。

2.3 メモリマップ

STM32L1xxxx データシートの包括的なメモリマップ図を参照してください。表 5 に、STM32L1xxxx 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 5. レジスタ境界アドレス

境界アドレス	ペリフェラル	バス	レジスタマップ
0xA000 0000 - 0xA000 0FFF	FSMC	AHB	セクション 25.5.7 : FSMC レジスタマップ (678 ページ)
0x5006 0000 - 0x5006 03FF	AES		セクション 23.12.13 : AES レジスタマップ (606 ページ)
0x4002 6400 - 0x4002 67FF	DMA2		セクション 11.4.7 : DMA レジスタマップ (270 ページ)
0x4002 6000 - 0x4002 63FF	DMA1		セクション 11.4.7 : DMA レジスタマップ (270 ページ)
0x4002 3C00 - 0x4002 3FFF	FLASH		セクション 3.9.10 : レジスタマップ (92 ページ)
0x4002 3800 - 0x4002 3BFF	RCC		セクション 6.3.15 : RCC レジスタマップ (173 ページ)
0x4002 3000 - 0x4002 33FF	CRC		セクション 4.4.4 : CRC レジスタマップ (97 ページ)
0x4002 1C00 - 0x4002 1FFF	GPIOG		セクション 7.4.12 : GPIO レジスタマップ (194 ページ)
0x4002 1800 - 0x4002 1BFF	GPIOF		
0x4002 1400 - 0x4002 17FF	GPIOH		
0x4002 1000 - 0x4002 13FF	GPIOE		
0x4002 0C00 - 0x4002 0FFF	GPIOD		
0x4002 0800 - 0x4002 0BFF	GPIOC		
0x4002 0400 - 0x4002 07FF	GPIOB		
0x4002 0000 - 0x4002 03FF	GPIOA		

表 5. レジスタ境界アドレス (続き)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x4001 3800 - 0x4001 3BFF	USART1	APB2	セクション 27.6.8 : USART レジスタマップ (767 ページ)
0x4001 3000 - 0x4001 33FF	SPI1		セクション 28.5.10 : SPI レジスタマップ (823 ページ)
0x4001 2C00 - 0x4001 2FFF	SDIO		セクション 29.9.16 : SDIO レジスタマップ (880 ページ)
0x4001 2400 - 0x4001 27FF	ADC		セクション 12.15.21 : ADC レジスタマップ (318 ページ)
0x4001 1000 - 0x4001 13FF	TIM11		セクション 17.4.21 : TIMx レジスタマップ (453 ページ)
0x4001 0C00 - 0x4001 0FFF	TIM10		セクション 17.4.21 : TIMx レジスタマップ (453 ページ)
0x4001 0800 - 0x4001 0BFF	TIM9		セクション 17.4.21 : TIMx レジスタマップ (453 ページ)
0x4001 0400 - 0x4001 07FF	EXTI		セクション 10.3.7 : EXTI レジスタマップ (252 ページ)
0x4001 0000 - 0x4001 03FF	SYSCFG		セクション 8.5.7 : SYSCFG レジスタマップ (229 ページ)

表 5. レジスタ境界アドレス (続き)

境界アドレス	ペリフェラル	バス	レジスタマップ
0x4000 7C00 - 0x4000 7C03	COMP	APB1	セクション 14.9.2 : COMP レジスタマップ (354 ページ)
0x4000 7C04 - 0x4000 7C5B	RI		セクション 8.5.7:SYSCFG レジスタマップ (229 ページ)
0x4000 7C5C - 0x4000 7FFF	OPAMP		セクション 15.4.4:OPAMP レジスタマップ (364 ページ)
0x4000 7400 - 0x4000 77FF	DAC		セクション 13.5.15:DAC レジスタマップ (344 ページ)
0x4000 7000 - 0x4000 73FF	PWR		セクション 5.4.3 : PWR レジスタマップ (128 ページ)
0x4000 6000 - 0x4000 63FF	USB デバイス FS SRAM 512バイト		セクション 24.5.4 : USB レジスタマップ (637 ページ)
0x4000 5C00 - 0x4000 5FFF	USB デバイス FS		
0x4000 5800 - 0x4000 5BFF	I2C2		セクション 26.6.10 : I2C レジスタマップ (715 ページ)
0x4000 5400 - 0x4000 57FF	I2C1		
0x4000 5000 - 0x4000 53FF	USART5		セクション 27.6.8:USART レジスタマップ (767 ページ)
0x4000 4C00 - 0x4000 4FFF	USART4		
0x4000 4800 - 0x4000 4BFF	USART3		
0x4000 4400 - 0x4000 47FF	USART2		
0x4000 3C00 - 0x4000 3FFF	SPI3		セクション 28.5.10 : SPI レジスタマップ (823 ページ)
0x4000 3800 - 0x4000 3BFF	SPI2		
0x4000 3000 - 0x4000 33FF	IWDG		セクション 21.4.5 : IWDG レジスタマップ (572 ページ)
0x4000 2C00 - 0x4000 2FFF	WWDG		セクション 22.6.4:WWDG レジスタマップ (579 ページ)
0x4000 2800 - 0x4000 2BFF	RTC		セクション 20.6.21:RTC レジスタマップ (565 ページ)
0x4000 2400 - 0x4000 27FF	LCD		セクション 16.5.6 : LCD レジスタマップ (392 ページ)
0x4000 1400 - 0x4000 17FF	TIM7		セクション 19.4.9:TIM6 および TIM7 レジスタマップ (522 ページ)
0x4000 1000 - 0x4000 13FF	TIM6		
0x4000 0C00 - 0x4000 0FFF	TIM5 (32ビット)		セクション 17.4.21 : TIMx レジスタマップ (453 ページ)
0x4000 0800 - 0x4000 0BFF	TIM4		
0x4000 0400 - 0x4000 07FF	TIM3		
0x4000 0000 - 0x4000 03FF	TIM2		

2.4 内蔵 SRAM

STM32L1xxxx は最大80 Kバイトの SRAM を内蔵しています。SRAM 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。SRAM の開始アドレスは、0x2000 0000 です。

読み出しと書き込みは、0 ウェイトステートの CPU 速度でアクセスされます。

SRAM からのブートが選択されている場合、または物理リマップが選択されている場合、CPU は S-Bus または ICode/D-bus を介して SRAM にアクセスすることができます（SYSCFG コントローラの [セクション 8.5.1 : SYSCFG メモリ再配置レジスタ \(SYSCFG_MEMRMP\)](#) レジスタ参照）。最適 SRAM 実行性能を得るためには、物理リマップ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）。

2.5 フラッシュメモリの概要

フラッシュメモリは、次の 3 つの物理領域から構成されています。

- フラッシュプログラムメモリブロック。アプリケーションプログラムとオプションのユーザデータを格納します。
- データ EEPROM メモリブロック。ユーザデータを格納します。
- 情報ブロック。次の 2 つの部分から構成されています。
 - － ハードウェアとメモリ保護機能ユーザ設定のオプションバイト。
 - － 当社独自のブートローダコードを格納するシステムメモリ。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 \(フラッシュ\)](#)を参照してください。

フラッシュメモリ命令とデータアクセスは、AHB バスを介して実行されます。プリフェッチブロックは I-bus を介する命令フェッチに使用されます。アービトレーションはフラッシュメモリインタフェース内で実行され、D-bus 上のデータアクセスが優先されます。また、フラッシュレジスタから制御されるフラッシュメモリ動作（書き込み/消去）の実行に必要なロジックも内蔵しています。

2.6 ビットバンディング

Cortex®-M3 メモリマップには、2 つのビットバンド領域が含まれています。これらの領域は、メモリのエイリアス領域の各ワードをメモリのビットバンド領域のビットにマップします。エイリアス領域のワードへの書き込みは、ビットバンド領域の対象ビットへの読み取り/変更/書き込み操作と同じ効果を持ちます。

STM32L1xxxx では、ペリフェラルレジスタと SRAM の両方がビットバンド領域にマッピングされます。これにより、シングルビットバンド書き込み/読み出し動作が可能になります。これらの動作は Cortex-M3 アクセスでのみ可能であり、他のバスマスタ（DMA など）からの動作ではできません。

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

$$bit\_word\_addr = bit\_band\_base + (byte\_offset \times 32) + (bit\_number \times 4)$$

ここで、

`bit_word_addr` は、対象ビットにマッピングされるエイリアスメモリ領域内のワードのアドレスです。

`bit_band_base` は、エイリアス領域の開始アドレスです

`byte_offset` は、対象ビットを含むビットバンド領域のバイトの番号

`bit_number` は、対象ビットのビット位置（0-7）

例：

エイリアス領域内の SRAM アドレス 0x2000 0300 にあるバイトのビット 2 をマッピングする方法を次の例で示します。

$$0x2200\ 6008 = 0x2200\ 0000 + (0x300 \times 32) + (2 \times 4)$$

アドレス 0x2200 6008 に対する書き込みは、SRAM アドレス 0x2000 0300 にあるバイトのビット 2 に対するリードモデファイライト命令と同じ効果があります。

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

ビットバンディングの詳細については、Cortex[®]-M3 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。

2.7 ブート設定

メモリマップは固定されているため、コード領域はアドレス 0x0000 0000 から始まり (ICode/D-bus 経由でアクセス)、データ領域 (SRAM) はアドレス 0x2000 0000 から始まります (S-Bus 経由でアクセス)。Cortex[®]-M3 CPU は、常に I-bus のリセットベクタをフェッチします。これは、ブート領域はコード領域 (一般にはフラッシュメモリ) でのみ使用可能なことを意味します。STM32L1xxxx マイクロコントローラは、フラッシュ以外の他のメモリ (内部 SRAM など) からのブートを可能にする特別なメカニズムを採用しています。

STM32L1xxxx では、BOOT[1 : 0] ピンを使って 3 種類のブートモードを選択することができます (表 6 参照)。

表 6. ブートモード

ブートモード選択ピン		ブートモード	エイリアシング
BOOT1	BOOT0		
x	0	メインフラッシュメモリ	メインフラッシュ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。

BOOT ピンの値は、リセット後、SYSCLK の 4 番目の立ち上がりエッジでラッチされます。ユーザはリセット後のブートモードを指定するために BOOT1 と BOOT0 ピンに必要な設定をする必要があります。

BOOT0 は専用ピンですが、BOOT1 は GPIO ピンとしても使われます。BOOT1 をサンプルした後、対応する GPIO ピンはアプリケーションから使用可能になります。

BOOT ピンは、STANDBY モードを終了するときにもサンプルされます。したがって、これらのピンは STANDBY モードでは必要なブートモード設定に保たれる必要があります。このスタートアップ遅延が経過した後、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 から始まるブートメモリからコードの実行を開始します。

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



物理リマップ

ブートピンを設定した後、アプリケーションソフトウェアからコード領域内でアクセス可能なメモリの内容を変更することができます（このため、S-Bus の代わりに ICode/DCode を介してコードを実行することができます）。このメモリ値の変更は、SYSCFG コントローラ内で [SYSCFG メモリ再配置レジスタ \(SYSCFG_MEMRMP\)](#) を設定することにより行われます。

その後で、次のメモリをリマップすることができます。

- メインフラッシュメモリ
- システムメモリ
- 内蔵 SRAM

注： オプションバイトで設定されたメモリ保護機能に応じて、使用できないブート／リマップ設定もあります（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（フラッシュ）の読み出し保護機能セクション](#)を参照してください）。

表 7. メモリマッピング対ブートモード／物理リマップ

アクセス	メインフラッシュメモリでのブート／リマップ	内蔵 SRAM でのブート／リマップ	システムメモリでのブート／リマップ
0x2000 0000 - 0x2000 BFFF	SRAM	SRAM	SRAM
0x1FF0 0000 - 0x1FF0 1FFF	システムメモリ	システムメモリ	システムメモリ
0x0802 0000 - 0x0FFF FFFF	予約済み	予約済み	予約済み
0x0800 0000 - 0x0805 FFFF	フラッシュメモリ	フラッシュメモリ	フラッシュメモリ
0x0002 0000 - 0x07FF FFFF	予約済み	予約済み	予約済み
0x0000 0000 - 0x0005 FFFF	フラッシュ (最大 512 KB) エイリアス	SRAM エイリアス	システムメモリ (8 KB) エイリアス

注： ブートメモリ空間でエイリアスされた場合でも、対応するメモリは元のメモリ空間でアクセスすることができます。

内蔵ブートローダ

内蔵ブートローダを使って、次のいずれかのインタフェースからフラッシュメモリに再書き込みを行います。

- Cat.1 および Cat.2 デバイスの場合：USART1 または USART2.
- Cat.3、Cat.4、Cat.5 デバイスの場合：USART1、USART2 または USB

このプログラムは、ST によって生産時にプログラムされ、システムメモリに配置されます。

3 フラッシュプログラムメモリとデータ EEPROM（フラッシュ）

3.1 フラッシュメモリの概要

- 最大512 KB のフラッシュプログラムメモリ
- メモリ構造（デュアルバンク、Cat.4とCat.5 デバイスの場合）：
 - － 最大512 KB のフラッシュプログラムメモリおよび最大16 KB のデータ EEPROM
 - － 最大8 KB のシステムメモリおよび最大64 バイトのオプションバイト

デュアルバンクデバイスの各バンクは、次のように構成されます。

- － 192/256 KB のプログラムメモリおよび 6/8 KB のデータ（Cat.4/Cat.5 デバイスの場合）
- － 4 KB のシステムメモリ、32 バイトのオプションバイト

フラッシュメモリインタフェース（FLITF）は次の機能を内蔵します。

- フラッシュメモリ読み出し動作：読み出しアクセスは 64ビットまたは 32ビットで行われます。
- フラッシュメモリ書き込み／消去動作
- バンクごとのリードホワイルライト（RWW）
- 読み出し/書き込み保護機能
- 書き込みアクセスは 32ビットで実行
- オプションバイトローダリセット
- 低消費電力モード：
 - － STM32L1xxxx が Standby モードまたは Stop モードのとき、フラッシュメモリはパワーダウンモードになります。
 - － STM32L1xxxx が Sleep モードのとき、フラッシュメモリはパワーダウンまたはアイドルモードになることができます。
 - － STM32L1xxxx が Run モードのとき、フラッシュメモリはパワーダウンまたはアイドルモードになることができます。

注： DMA は、読み出し命令を使ってフラッシュメモリモジュールだけをアクセスすることができます。

注： データ EEPROM からのコードの実行はできません。

3.2 フラッシュメモリ構成

メモリは、プログラムメモリブロック、データ EEPROM ブロック、情報ブロックとして構成されています。表 8、表 9、表 10に、メモリ構成を示します。

フラッシュプログラムメモリブロックは、4 KB のセクタに分割され、各セクタはさらに各ページ 256 バイトで構成される 16 ページに分割されます。セクタは、書き込み保護機能の単位でもあります。ページはプログラムメモリブロックの消去の単位です。

フラッシュプログラムメモリページは、ハーフページ書き込み命令または高速ワード書き込み命令を使って書き込むことができます。

データ EEPROM は次の単位で消去し書き込むことができます。

- ダブルワード
- ワード／高速ワード
- ハーフワード／高速ハーフワード
- バイト／高速バイト

フラッシュメモリに対する書き込み／消去動作中に（ただしハーフページ書き込みまたはダブルワード消去／書き込みを除く）、同じバンクのフラッシュメモリを読み出そうとすると、バスが停止してしまいます。読み出し命令は、書き込み命令が完了した後に正常に実行されます。これは、同じバンクで書き込み／消去動作の進行中に、コードまたはデータのフェッチを実行できないことを意味します。

詳細については、[セクション 3.4.2 : メモリの消去 \(63 ページ\)](#) と [セクション 3.4.3 : メモリへの書き込み \(64 ページ\)](#) を参照してください。

表 8. フラッシュメモリモジュール構成（Cat.1とCat.2 デバイス）

表 6. プログラムメモリセクタ構成 (Cat.1とCat.2 ヴァリエ)

ブロック	名称		メモリアドレス	サイズ
プログラムメモリ	セクタ 0	ページ 0	0x0800 0000 - 0x0800 00FF	256 バイト
		ページ 1	0x0800 0100 - 0x0800 01FF	256 バイト
		ページ 2	0x0800 0200 - 0x0800 02FF	256 バイト
		ページ 3	0x0800 0300 - 0x0800 03FF	256 バイト
		ページ 4~7	0x0800 0400 - 0x0800 07FF	1 KB
		ページ 8~11	0x0800 0800 - 0x0800 0BFF	1 KB
		ページ 12~15	0x0800 0C00 - 0x0800 0FFF	1 KB
	セクタ 1		0x0800 1000 - 0x0800 1FFF	4 KB
	セクタ 2		0x0800 2000 - 0x0800 2FFF	4 KB
	セクタ 3		0x0800 3000 - 0x0800 3FFF	4 KB
	.		.	.
	.		.	.
	.		.	.
セクタ 30		0x0801 E000 - 0x0801 EFFF	4 KB	
セクタ 31		0x0801 F000 - 0x0801 FFFF	4 KB	
データ EEPROM			0x0808 0000 - 0x0808 0FFF	4096 バイト

表 8. フラッシュメモリモジュール構成 (Cat.1とCat.2 デバイス) (続き)

ブロック	名称		メモリアドレス	サイズ
情報ブロック	システムメモリ	ページ 0	0x1FF0 0000 - 0x1FF0 00FF	256 バイト
		ページ 1	0x1FF0 0100 - 0x1FF0 01FF	256 バイト
		ページ 2	0x1FF0 0200 - 0x1FF0 02FF	256 バイト
		ページ 3	0x1FF0 0300 - 0x1FF0 03FF	256 バイト
		.	.	.
		.	.	.
		ページ 15	0x1FF0 0F00 - 0x1FF0 0FFF	256 バイト
	オプションバイト		0x1FF8 0000 - 0x1FF8 001F	32 バイト
	工場情報		0x1FF8 0020 - 0x1FF8 00FF	224 バイト

表 9. フラッシュメモリモジュール構成 (Cat.3 デバイス)

ブロック	名称		メモリアドレス	サイズ
プログラムメモリ	セクタ 0	ページ 0	0x0800 0000 - 0x0800 00FF	256 バイト
		ページ 1	0x0800 0100 - 0x0800 01FF	256 バイト
		ページ 2	0x0800 0200 - 0x0800 02FF	256 バイト
		ページ 3	0x0800 0300 - 0x0800 03FF	256 バイト
		ページ 4~7	0x0800 0400 - 0x0800 07FF	1 KB
		ページ 8~11	0x0800 0800 - 0x0800 0BFF	1 KB
		ページ 12~15	0x0800 0C00 - 0x0800 0FFF	1 KB
	セクタ 1	ページ 16~31	0x0800 1000 - 0x0800 1FFF	4 KB
	セクタ 2	ページ 32~47	0x0800 2000 - 0x0800 2FFF	4 KB
	セクタ 3	ページ 48~63	0x0800 3000 - 0x0800 3FFF	4 KB
	.	.	.	.
	.	.	.	.
	.	.	.	.
	セクタ 30	ページ 478~495	0x0801 E000 - 0x0801 EFFF	4 KB
	セクタ 31	ページ 496~511	0x0801 F000 - 0x0801 FFFF	4 KB
	セクタ 32~セクタ 47	ページ 512~767	0x0802 0000 - 0x0802 FFFF	64 KB
	セクタ 48~セクタ 63	ページ 768~1024	0x0803 0000 - 0x0803 FFFF	64 KB
	データ EEPROM		0x0808 0000 - 0x0808 1FFF	8 KB

表 9. フラッシュメモリモジュール構成（Cat.3 デバイス）（続き）

ブロック	名称		メモリアドレス	サイズ
情報ブロック	システムメモリ	ページ 0	0x1FF0 0000 - 0x1FF0 00FF	256 バイト
		ページ 1	0x1FF0 0100 - 0x1FF0 01FF	256 バイト
		ページ 2	0x1FF0 0200 - 0x1FF0 02FF	256 バイト
		ページ 3	0x1FF0 0300 - 0x1FF0 03FF	256 バイト
		.	.	.
		.	.	.
		.	.	.
		ページ 15	0x1FF0 0F00 - 0x1FF0 0FFF	256 バイト
		ページ 16~31	0x1FF0 1000 - 0x1FF0 1FFF	4 KB
	オプションバイト		0x1FF8 0000 - 0x1FF8 001F	32 バイト
	工場情報		0x1FF8 0020 - 0x1FF8 00FF	224 バイト

表 10. フラッシュメモリモジュール構成（Cat.4 デバイス）

ブロック	名称		メモリアドレス	サイズ
プログラムメモリ バンク 1	セクタ 0	ページ 0	0x0800 0000 - 0x0800 00FF	256 バイト
		ページ 1	0x0800 0100 - 0x0800 01FF	256 バイト
		ページ 2	0x0800 0200 - 0x0800 02FF	256 バイト
		ページ 3	0x0800 0300 - 0x0800 03FF	256 バイト
		ページ 4~7	0x0800 0400 - 0x0800 07FF	1 KB
		ページ 8~11	0x0800 0800 - 0x0800 0BFF	1 KB
		ページ 12~15	0x0800 0C00 - 0x0800 0FFF	1 KB
	セクタ 1	ページ 16~31	0x0800 1000 - 0x0800 1FFF	4 KB
	セクタ 2	ページ 32~47	0x0800 2000 - 0x0800 2FFF	4 KB
	セクタ 3	ページ 48~63	0x0800 3000 - 0x0800 3FFF	4 KB
	.	.	.	.
	.	.	.	.
	.	.	.	.
	セクタ 30	ページ 478~495	0x0801 E000 - 0x0801 EFFF	4 KB
	セクタ 31	ページ 496~511	0x0801 F000 - 0x0801 FFFF	4 KB
	セクタ 32~ セクタ 47	ページ 512~767	0x0802 0000 - 0x0802 FFFF	64 KB
プログラムメモリ バンク 2	セクタ 48~セ クタ 79	ページ 768~1279	0x0803 0000 - 0x0804 FFFF	128 KB
	セクタ 80~ セクタ 95	ページ 1278~1535	0x0805 0000 - 0x0805 FFFF	64 KB
データ EEPROM バンク 1			0x0808 0000 - 0x0808 17FF	6 KB
データ EEPROM バンク 2			0x0808 1800 - 0x0808 2FFF	6 KB

表 10. フラッシュメモリモジュール構成 (Cat.4 デバイス) (続き)

ブロック	名称	メモリアドレス	サイズ
システムメモリバンク 1	ページ 0	0x1FF0 0000 - 0x1FF0 00FF	256 バイト
	ページ 1	0x1FF0 0100 - 0x1FF0 01FF	256 バイト
	ページ 2	0x1FF0 0200 - 0x1FF0 02FF	256 バイト
	ページ 3	0x1FF0 0300 - 0x1FF0 03FF	256 バイト
	.	.	.
	ページ 15	0x1FF0 0F00 - 0x1FF0 0FFF	256 バイト
システムメモリバンク 2	ページ 16~31	0x1FF0 1000 - 0x1FF0 1FFF	4 KB
オプションバイトバンク 1		0x1FF8 0000 - 0x1FF8 001F	32 バイト
工場情報バンク 1		0x1FF8 0020 - 0x1FF8 007F	96 バイト
オプションバイトバンク 2		0x1FF8 0080 - 0x1FF8 009F	32 バイト
工場情報バンク 2		0x1FF8 00A0 - 0x1FF8 00FF	96 バイト

表 11. フラッシュメモリモジュール構成 (Cat.5 デバイス)

ブロック	名称		メモリアドレス	サイズ
プログラムメモリ バンク 1	セクタ 0	ページ 0	0x0800 0000 - 0x0800 00FF	256 バイト
		ページ 1	0x0800 0100 - 0x0800 01FF	256 バイト
		ページ 2	0x0800 0200 - 0x0800 02FF	256 バイト
		ページ 3	0x0800 0300 - 0x0800 03FF	256 バイト
		ページ 4～7	0x0800 0400 - 0x0800 07FF	1 KB
		ページ 8～11	0x0800 0800 - 0x0800 0BFF	1 KB
		ページ 12～15	0x0800 0C00 - 0x0800 0FFF	1 KB
	セクタ 1	ページ 16～31	0x0800 1000 - 0x0800 1FFF	4 KB
	セクタ 2	ページ 32～47	0x0800 2000 - 0x0800 2FFF	4 KB
	セクタ 3	ページ 48～63	0x0800 3000 - 0x0800 3FFF	4 KB
	.	.	.	.
	.	.	.	.
	.	.	.	.
	セクタ 30	ページ 478～495	0x0801 E000 - 0x0801 EFFF	4 KB
セクタ 31	ページ 496～511	0x0801 F000 - 0x0801 FFFF	4 KB	
セクタ 32～ セクタ 63	ページ 512～1023	0x0802 0000 - 0x0803 FFFF	128 KB	
プログラムメモリ バンク 2	セクタ 64～ セクタ 95	ページ 1024～1535	0x0804 0000 - 0x0805 FFFF	128 KB
	セクタ 96～ セクタ 127	ページ 1536～2047	0x0806 0000 - 0x0807 FFFF	128 KB
データ EEPROM バンク 1			0x0808 0000 - 0x0808 1FFF	8 KB

表 11. フラッシュメモリモジュール構成（Cat.5 デバイス）（続き）

ブロック	名称	メモリアドレス	サイズ
データ EEPROM バンク 2		0x0808 2000 - 0x0808 3FFF	8 KB
システムメモリバンク 1	ページ 0	0x1FF0 0000 - 0x1FF0 00FF	256 バイト
	ページ 1	0x1FF0 0100 - 0x1FF0 01FF	256 バイト
	ページ 2	0x1FF0 0200 - 0x1FF0 02FF	256 バイト
	ページ 3	0x1FF0 0300 - 0x1FF0 03FF	256 バイト
	.	.	.
	ページ 15	0x1FF0 0F00 - 0x1FF0 0FFF	256 バイト
システムメモリバンク 2	ページ 16~31	0x1FF0 1000 - 0x1FF0 1FFF	4 KB
オプションバイトバンク 1		0x1FF8 0000 - 0x1FF8 001F	32 バイト
工場情報バンク 1		0x1FF8 0020 - 0x1FF8 007F	96 バイト
オプションバイトバンク 2		0x1FF8 0080 - 0x1FF8 009F	32 バイト
工場情報バンク 2		0x1FF8 00A0 - 0x1FF8 00FF	96 バイト

3.3 読み出しインタフェース

3.3.1 CPU クロック周波数とフラッシュメモリ読み出し時間との関係

フラッシュメモリは、64 ビット単位または 32 ビット単位で読み出されます。

フラッシュアクセス制御レジスタ（FLASH_ACR）の ACC64 ビットをセットすると、64 ビットアクセスが設定されます。このアクセスモードは、書き込み動作の実行を高速化します。プリフェッチは、1 CPU サイクル内にフラッシュメモリをアクセスできない場合に役立ちます。この場合、CPU クロック（HCLK）の周波数とデバイス電源電圧に従って、フラッシュアクセス制御レジスタ（FLASH_ACR）にウェイトステート数（LATENCY）を正しく設定する必要があります。表 12 に、ウェイトステート数と CPU クロック周波数との対応を示します。

表 12. CPU クロック（HCLK）周波数に対応したウェイトステート数（WS）

HCLK 周波数（MHz）			ウェイトステート数 （LATENCY）
V _{DD} 範囲 1.65 V～3.6 V		V _{DD} 範囲 1.71 V～3.6 V	
範囲 3	範囲 2	範囲 1	
f _{HCLK} ≤ 2.1 MHz （Cat.1 デバイス） f _{CPU} ≤ 4.2 MHz （Cat.2、Cat.3、Cat.4、 Cat.5 デバイス）	f _{HCLK} ≤ 8 MHz	f _{HCLK} ≤ 16 MHz	0 WS（1 HCLK サイクル）
f _{HCLK} ≤ 4.2 MHz （Cat.1 デバイス） f _{HCLK} ≤ 8（in Cat.2、Cat.3、 Cat.4、Cat.5 デバイス）	f _{HCLK} ≤ 16 MHz	f _{HCLK} ≤ 32 MHz	1 WS（2 HCLK サイクル）

フラッシュメモリに 32 ビット単位でアクセスすることも可能です。FLASH_ACR の ACC64 ビットをクリアすると、これを行うことができます。この場合、プリフェッチを無効化する必要があります。32 ビットアクセスでは消費電力が少なくなるため、CPU 周波数が低いとき使用されます。この場合、ウェイトステート数は 0 である必要があります。

リセット後に使用されるクロックは、FLASH_ACR レジスタで設定される 0 WS の MSI（2 MHz）です。32 ビットアクセスが有効化され、プリフェッチが無効化されます。

CPU 周波数でのフラッシュメモリアクセスに必要なウェイトステート数の調整に次のソフトウェアシーケンスを使うことを ST 社は強く推奨します。

CPU 周波数の増加（同じ電圧範囲）。

- FLASH_ACR で ACC64 ビットをセットして、64 ビットアクセスを設定します。
- FLASH_ACR を読み出して、64 ビットアクセスであることを確認します。
- FLASH_ACR の LATENCY ビットに 1 WS を設定します。
- FLASH_ACR を読み出して、新しい WS 数が設定されていることを確認します。
- RCC_CFGR レジスタの SW ビットに書き込みを行い、CPU クロックソースを変更します。
- 必要に応じて、RCC_CFGR の HPRE ビットに書き込みを行い、CPU クロックプリスケアラを変更します。
- RCC_CFGR レジスタでクロックソースステータス（SWS ビット）または／および AHB プリスケアラ値（HPRE ビット）を読み出して、それぞれ新しい CPU クロックソースまたは／および新しい CPU クロックプリスケアラ値が設定されていることを確認します。

CPU 周波数の減少（同じ電圧範囲）。

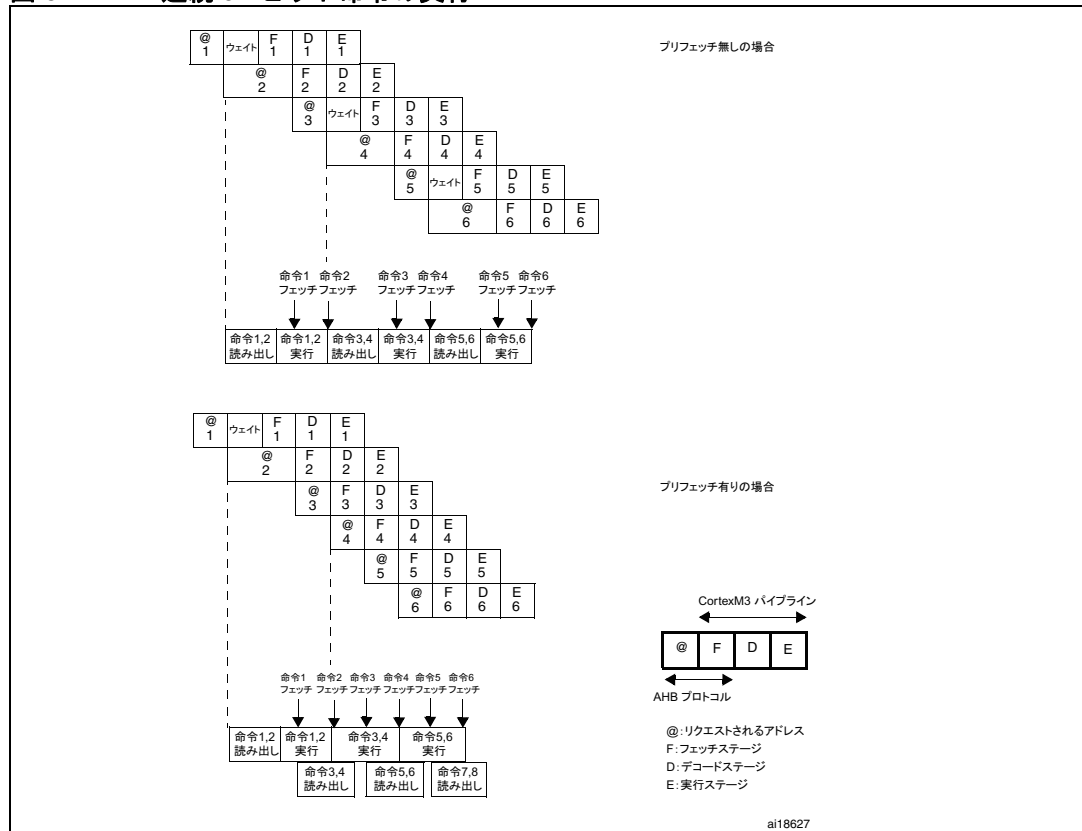
- RCC_CFGR レジスタの SW ビットに書き込みを行い、CPU クロックソースを変更します。
- 必要に応じて、RCC_CFGR の HPRE ビットに書き込みを行い、CPU クロックプリスケアラを変更します。
- RCC_CFGR レジスタでクロックソースステータス（SWS ビット）または／および AHB プリスケアラ値（HPRE ビット）を読み出して、それぞれ新しい CPU クロックソースまたは／および新しい CPU クロックプリスケアラ値が設定されていることを確認します。
- FLASH_ACR の LATENCY ビットに新しい WS 数を設定します。
- FLASH_ACR を読み出して、新しい WS 数が設定されていることを確認します。
- FLASH_ACR の ACC64 をクリアして、32 ビットアクセスを設定します。
- FLASH_ACR を読み出して、32 ビットアクセスであることを確認します。

3.3.2 フラッシュアクセスが 64 ビットのときの命令プリフェッチ

各フラッシュメモリ読み出し命令では、起動されたプログラムに応じて 32 ビットの 2 命令または 16 ビットの 4 命令からの 64 ビットが提供されます。したがって、シーケンシャルコードの場合、前の読み出し命令行の実行には、少なくとも 2 CPU サイクルが必要です。CPU から現在の命令行が要求されている間に、I-bus 上のプリフェッチを使って、フラッシュメモリから次のシーケンシャル命令行を読み出すことができます。FLASH_ACR レジスタの PRFTEN ビットをセットすると、プリフェッチが有効化されます。フラッシュメモリのアクセスに少なくとも 1 ウェイトステートが必要な場合に、この機能が役立ちます。

図 5 に、フラッシュメモリアクセスに 1 WS が必要な場合に、プリフェッチありとなしのシーケンシャル 32 ビット命令の実行を示します。

図 5. 連続 32 ビット命令の実行



注: コードがシーケンシャルでない（ブランチ）場合には、使用された現在の命令行にも、プリフェッチされた命令行にも、命令が存在しないことがあります。この場合、サイクル数で表したペナルティは、少なくともウェイトステート数に一致します。

表 13に、サポートしている ACC64、LATENCY、PRFTEN の設定を示します。

表 13. FLASH_ACR に対する許容設定

LATENCY	ACC64 = 0		ACC64 = 1	
	PRFTEN = 0	PRFTEN = 1	PRFTEN = 0	PRFTEN = 1
0	あり	なし	あり	あり
1	なし	なし	あり	あり

3.3.3 データ管理

CPU パイプラインの実行ステージに D-bus を介してフラッシュメモリからリテラルプールがフェッチされます。このため、要求されたリテラルプールが提供されるまで CPU パイプラインが停止してしまいます。リテラルプールのために失われる時間を制限するため、DCode AHB データバスを介するアクセスが ICode AHB 命令バスを介するアクセスより優先されます。

3.4 メモリ命令

3.4.1 メモリのロック/アンロック

次のブロックは、個別にロック/アンロックすることができます。

- PECCR レジスタ付きのデータ EEPROM
- プログラムメモリ
- オプションバイト

各命令に要求されるステップを次のセクションに示します。

データ EEPROM ブロックと FLASH_PECR レジスタのアンロック

リセット後、データ EEPROM ブロックと書き込み／消去制御レジスタ (FLASH_PECR) は書き込みモードでアクセスできなくなり、FLASH_PECR の PELOCK ビットはセットされます。同じアンロックシーケンスにより、同時に両保護機能が解除されます。

次のシーケンスを使って、データ EEPROM ブロックと FLASH_PECR レジスタをアンロックします。

- 書き込み／消去キーレジスタ (FLASH_PEKEYR) へ PEKEY1= 0x89ABCDEF を書込みます。
- 書き込み／消去キーレジスタ (FLASH_PEKEYR) へ PEKEY2= 0x02030405 を書込みます。

不正なキーシーケンスにより、データ EEPROM ブロックと FLASH_PECR レジスタは次のリセットまでロックされ、バスエラー (Cortex[®]-M3 ハードフォルトまたはバスフォルト) が返されます。このバスエラーは次の 3 ケースで返されます。

- 入力された PEKEY1 値が誤りの場合、最初の書き込みアクセスの後
- PEKEY1 が正常に入力され、かつ PEKEY2 値が一致しない場合、2 番目書き込みアクセスで
- PEKEYR へ 3 番目の値を書き込もうとした場合

正常に実行されると、アンロックシーケンスにより、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 を再度ロックするときは、ソフトウェアから FLASH_PECR の PELOCK ビットをセットするだけで済みます。

プログラムメモリのアンロック

プログラムメモリ（書き込み保護（WRP）されていないページ）への書き込みに対してその他の保護機能も内蔵されています。

リセット後、書き込みモードではプログラムメモリをアクセスできなくなります。すなわち、FLASH_PECR の PRGLOCK ビットがセットされます。PRGLOCK ビットをクリアすると、プログラムメモリへの書き込みアクセスが再度許可されます。

次のシーケンスを使って、プログラムメモリをアンロックします。

- FLASH_PECR レジスタをアンロックします
- プログラムメモリキーレジスタ（FLASH_PRGKEYR）へ PRGKEY1= 0x8C9DAEBF を書き込みます
- プログラムメモリキーレジスタ（FLASH_PRGKEYR）へ PRGKEY2= 0x13141516 を書き込みます

不正なキーシーケンスにより、次のリセットまで FLASH_PECR の PRGLOCK がロックされ、バスエラー（Cortex[®]-M3 ハードフォルトまたはバスフォルト）が返されます。このバスエラーは次の 3 つのケースで返されます。

- 入力された PRGKEY1 値が誤りの場合、最初の書き込みアクセスの後
- PRGKEY1 が正常に入力され、かつ PRGKEY2 値が一致しない場合、2 番目の書き込みアクセスで
- PRGKEYR へ 3 番目の値を書き込もうとした場合

正常に実行された場合、アンロックシーケンスにより PRGLOCK ビットがクリアされ、プログラムメモリへの書き込みアクセスが可能になります。

プログラムメモリを再度ロックするときは、ソフトウェアから FLASH_PECR の PRGLOCK ビットをセットするだけで済みます。

オプションバイトブロックのアンロック

オプションバイトブロックに対してその他の書き込み保護機能も内蔵されています。

リセット後、書き込みモードではオプションバイトをアクセスできなくなります。すなわち、FLASH_PECR の OPTLOCK ビットがセットされます。OPTLOCK ビットをクリアすると、オプションバイトへの書き込みアクセスが再度許可されます。

次のシーケンスを使って、オプションバイトをアンロックします。

- FLASH_PECR レジスタをアンロックします
- オプションキーレジスタ（FLASH_OPTKEYR）へ OPTKEY1= 0xFBEAD9C8 を書き込みます
- オプションキーレジスタ（FLASH_OPTKEYR）へ OPTKEY2= 0x24252627 を書き込みます

不正なキーシーケンスにより、次のリセットまで FLASH_PECR の OPTLOCK がロックされ、バスエラー（Cortex[®]-M3 ハードフォルトまたはバスフォルト）が返されます。このバスエラーは次の 3 つのケースで返されます。

- 入力された OPTKEY1 値が誤りの場合、最初の書き込みアクセスの後
- OPTKEY1 が正常に入力され、かつ OPTKEY2 値が一致しない場合、2 番目の書き込みアクセスで
- OPTKEYR へ 3 番目の値を書き込もうとした場合

正常に実行された場合、アンロックシーケンスにより OPTLOCK ビットがクリアされ、オプションバイトへの書き込みアクセスが可能になります。

オプションバイトを再度ロックするときは、ソフトウェアから FLASH_PECR の OPTLOCK ビットをセットするだけで済みます。

3.4.2 メモリの消去

プログラムメモリとデータ EEPROM に対して異なる消去命令があります。これは、消去単位が異なるためです。次の命令があります。

- **ワード消去とダブルワード消去** : データ EEPROM
- **ページ消去** (さらに Cat.4 と Cat.5 デバイスに対する **パラレルページ消去**) : プログラムメモリ
- **マス消去** : この命令は、プログラムメモリ、データ EEPROM、オプションバイト (Cat.4 と Cat.5 デバイスの両バンク) を消去します

データ EEPROM ワード 消去

この命令を使って、データ EEPROM 内のワードを消去します。そのためには、

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- データ EEPROM 内の有効なアドレスへ値 0x0000 0000 のワードを書き込みます
- これにより消去フェーズが開始されます

注 : この機能は、両バンク、RAM または外部メモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

データ EEPROM のダブルワード消去

この命令を使って、データ EEPROM 内のダブルワードを消去します。

そのためには、

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FLASH_PECR レジスタの ERASE ビットをセットします
- FLASH_PECR レジスタの DATA ビットをセットしてデータダブルワードを消去します
- BSY ビットがクリアされるのを待ちます
- 消去する 2 つの各データワードへ 0x0000 0000 を書き込みます

注 : この機能は、1 バンクごとに、または RAM から実行することができます。この機能を実行中の同じバンクで読み出しアクセスを行うと、バスエラーが発生します。

警告 : データ EEPROM のダブルワード消去は、SRAM から、外部メモリから、またはバンク 1 からバンク 2 へ、およびバンク 2 からバンク 1 へのみ実行することができます。

プログラムメモリのページ消去

この命令を使って、プログラムメモリ（64 ワード）内のページを消去します。そのためには、

- FLASH_PECR レジスタをアンロックします
- プログラムメモリをアンロックします
- FLASH_PECR レジスタの ERASE ビットをセットします
- FLASH_PECR レジスタの **PROG** ビットをセットしてプログラムページを選択します。
- BSY ビットがクリアされるのを待ちます
- 消去するプログラムページの先頭ワードへ 0x0000 0000 を書き込みます。

注： この機能は、両バンク、RAM または外部メモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

プログラムメモリのパラレルページ消去

この命令は、Cat.4とCat.5 デバイスに対してのみ使用可能です。プログラムメモリ内の 2 つのパラレルページ（各バンクで 64 ワード）の消去に使います。そのためには、

- FLASH_PECR レジスタをアンロックします
- プログラムメモリをアンロックします
- FLASH_PECR レジスタの PARALLBANK ビットをセットします
- FLASH_PECR レジスタの ERASE ビットをセットします
- FLASH_PECR レジスタの **PROG** ビットをセットしてプログラムページを選択します。
- BSY ビットがクリアされるのを待ちます
- 最初のバンク内の消去するプログラムページの先頭ワードへ 0x0000 0000 を、2 番目のバンク内の消去するプログラムページの先頭ワードへ 0x0000 0000 を、それぞれ書き込みます

注： この機能は、RAM または外部メモリから実行することができます。

3.4.3 メモリへの書き込み

プログラムメモリへ的高速ワード書き込み

この命令を使って、プログラムメモリへワードを書き込みます。ただしワードは消去済みとします。そのためには、

- FLASH_PECR レジスタをアンロックします
- プログラムメモリをアンロックします
- プログラムメモリ内の有効なアドレスへワードを書き込みますこれにより書き込みフェーズが開始されます

注： この機能は、両バンク、RAM または外部メモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

プログラムメモリのハーフページ書き込み

この命令を使って、プログラムメモリに対するハーフページ (32 ワード) 書き込みを行います。そのためには、

- FLASH_PECR レジスタをアンロックします
- プログラムメモリをアンロックします
- FLASH_PECR レジスタの FPRG ビットをセットします (これにより、データバッファのローディングシーケンスを実行するように FLASH_PECR が設定されます)
- FLASH_PECR レジスタの PROG ビットをセットして、必要とされるプログラムメモリページにアクセスします
- BSY ビットがクリアされるのを待ちます
- プログラムメモリアドレス空間のハーフページに対して 32 種類のワードを直接書き込みします。これらのワードは、ワード 0 から始まりワード 31 で終わるようにシーケンシャルに書き込む必要があります

警告： ハーフページ書き込みは、SRAM から、外部メモリから、またはバンク 1 からバンク 2 へ、およびバンク 2 からバンク 1 へのみ実行することができます。

- 注：**
- 1 **書き込むワード数が 32 ワードを超える場合、32 ワードの後にもう 1 回ハーフページ書き込み命令が開始され、フラッシュメモリに対する別のアクセスが発生する前にこの命令を完了する必要があります。**
 - 2 **この機能を実行中の同じバンクで読み出しアクセスを行うと、バスエラーが発生します。**

プログラムメモリのパラレルハーフページ書き込み

この命令を使ってプログラムメモリの 2 つのハーフページへ書き込みを行います (各バンクで 32 ワード)。この命令は、Cat.4 と Cat.5 デバイスに対してのみ使用可能です。

- FLASH_PECR レジスタをアンロックします
- プログラムメモリをアンロックします
- FLASH_PECR レジスタの PARALL_BANK ビットをセットして、32 ワードの 2 シーケンスによるパラレルローディングを各バンク 1 回実行するように、FLASH_PECR を設定します
- FLASH_PECR レジスタの FPRG ビットをセットします (これにより、データバッファのローディングシーケンスを実行するように FLASH_PECR が設定されます)
- FLASH_PECR レジスタの PROG ビットをセットして、必要とされるプログラムメモリページにアクセスします
- BSY ビットがクリアされるのを待ちます
- プログラムメモリアドレス空間内の 2 ハーフページへ 64 種類のワードを直接書き込みます (各バンクで 32 ワード) これらの 32 ワードは、バンク 1 のワード 0 から始まりワード 31 で終わるようにシーケンシャルに書き込む必要があります。次に、32 ワードはバンク 2 のワード 32 から始まりワード 63 で終わるようにシーケンシャルに書き込む必要があります。

- 注：** **この機能は、RAM または外部メモリから実行することができます。この機能実行中の EEprom に対する読み出しアクセスは禁止されています。**



データ EEPROM のダブルワード書き込み

この命令を使って、データ EEPROM に対するダブルワード書き込みを行います。そのためには、

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FLASH_PECR レジスタの FPRG ビットをセットします (これにより、データバッファのローディングシーケンスを実行するように FLASH_PECR が設定されます)
- FLASH_PECR レジスタの DATA ビットをセットして、必要とされるデータ EEPROM ページをアクセスします
- BSY ビットがクリアされるのを待ちます
- データ EEPROM アドレス空間に対して 2 ワードを書き込んで、ダブルワードを直接書き込みます。これらのワードは、ワード 0 から始まりワード 1 で終わるようにシーケンシャルに書き込む必要があります

警告： データ EEPROM のダブルワード書き込みは、SRAM から、外部メモリから、またはバンク 1 からバンク 2 へ、およびバンク 2 からバンク 1 へのみ実行することができます。

- 注：**
- 1 ロードする最初のアドレスがダブルワードの 開始アドレス (ダブルワードの倍数) である場合にのみ、データ EEPROM ダブルワードがデータ EEPROM へ書き込まれます。
 - 2 この機能を実行中の同じバンクで読み出しアクセスを行うと、バスエラーが発生します。

データ EEPROM の高速ワード書き込み

この命令を使って、データ EEPROM へワードを書き込みます。ただしワードは消去済みとします。この命令に要する時間は 1 tprog です (詳細については、表 19 (77 ページ) を参照してください)。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) をクリアします。ただし、ワードは消去済みとします (0x00000000)。
- データ EEPROM 内の有効なアドレスへワードを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいワードに対して ECC が計算されます
 - 書き込み命令が直ちに実行されます (インタフェースにより 読み出されるワードは 0x00000000 であり、FTDW ビットはクリアされている必要があります)

- 注：** この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 tprog 間停止します。

データ EEPROM のワード書き込み

この命令を使って、書き込むワードの前の値によらずに、データ EEPROM に対するワード書き込みを行います。この命令に要する時間は、FTDW ビットに応じて、1 または 2 tprog です (詳細につい

では、表 19 (77 ページ) を参照してください。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) を設定 (セット/クリア) して、書き込むワードの前の値によらずにワード書き込みを実行します。
- データ EEPROM 内の有効なアドレスヘワードを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいワードに対して新しい ECC が計算されます
 - ケース 1 : FTDW ビット = 0:
インタフェースにより読み出されるワードが 0x00000000 でない場合、消去命令が自動的に実行され、書き込み命令が続きます。この動作に要する時間は、2 tprog です。
インタフェースにより読み出されるワードが 0x00000000 の場合、書き込み命令が直ちに実行されます (高速ワード書き込みと同じ時間、1 tprog を要します)。
 - ケース 2 : FTDW ビット = 1:
FTDW ビットがセットされている場合、消去命令が常に自動的に実行され、書き込み命令が続きます。この動作に要する時間は、2 tprog です。

注 : この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

データ EEPROM の高速ハーフワード書き込み

この命令を使って、データ EEPROM へ NON NULL^(a) ハーフワードを書き込みます。ただしワード全体は消去済みとします。この命令に要する時間は 1 tprog です (詳細については、表 19 (77 ページ) を参照してください)。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) をクリアします。ただし、ワードは消去済みとします (0x00000000)。
- データ EEPROM 内の有効なアドレスへハーフワードを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいハーフワードに対して ECC が計算されます
 - 書き込み命令が直ちに実行されます (インタフェースにより読み出されるワードは 0x00000000 であり、FTDW ビットはクリアされている必要があります)

注 : この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

データ EEPROM のハーフワード書き込み

この命令を使って、書き込むワードの前の値によらずに、データ EEPROM に対する NON NULL^(a) ハーフワード書き込みを行います。この命令に要する時間は、FTDW ビットに応じて、1 または 2

a. この制約は Cat.1 デバイスに対してのみ適用されます。

tprog です (詳細については、表 19 (77 ページ) を参照してください)。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) を設定 (セット/クリア) して、書き込むハーフワードの前の値によらずにハーフワード書き込みを実行します。
- データ EEPROM 内の有効なアドレスへハーフワードを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいハーフワードに対して新しい ECC が計算されます
 - ケース 1 : FTDW ビット = 0:
インタフェースにより読み出されるワードが 0x00000000 でない場合、消去命令が自動的に実行され、書き込み命令が続きます。この動作に要する時間は、2 tprog です。
インタフェースにより読み出されるワードが 0x00000000 の場合、書き込み命令が直ちに実行されます (高速ハーフワード書き込みと同じ時間、1 tprog を要します)。
 - ケース 2 : FTDW ビット = 1:
消去命令が常に自動的に実行され、書き込み命令が続きます。この動作に要する時間は、2 tprog です。

注 : この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 tprog 間停止します。

データ EEPROM の高速バイト書き込み

この命令を使って、データ EEPROM へ NON NULL^(a) バイトを書き込みます。ただしワード全体は消去済みとします。この命令に要する時間は 1 tprog です (詳細については、表 19 (77 ページ) を参照してください)。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) をクリアします。ただし、ワードは消去済みとします (0x00000000)。
- データ EEPROM 内の有効なアドレスへバイトを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいバイトに対して新しい ECC が計算されます
 - 書き込み命令が直ちに実行されます (インタフェースにより読み出されるワードは 0x00000000 であり、FTDW ビットはクリアされている必要があります)

注 : この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 tprog 間停止します。

データ EEPROM バイト書き込み

この命令を使って、書き込むワードの前の値によらずに、データ EEPROM に対する NON NULL⁽¹⁾ バイトを書き込みを行います。この命令に要する時間は、FTDW ビットに応じて、1 または 2 tprog

a. この制約は Cat.1 デバイスに対してのみ適用されます。

です (詳細については、表 19 (77 ページ) を参照してください)。

- データ EEPROM と FLASH_PECR レジスタをアンロックします
- FTDW ビット (FLASH_PECR[8]) を設定 (セット/クリア) して、書き込むワードの前の値によらずにバイト書き込みを実行します。
- データ EEPROM 内の有効なアドレスへ NON NULL バイトを書き込みます。
- 次に、フラッシュメモリインタフェースにより、次の動作が自動的に実行されます。
 - 書き込むワードをフラッシュメモリインタフェースがアドレス指定して読み出します
 - メモリへ書き込む新しいバイトに対して新しい ECC が計算されます
 - ケース 1 : FTDW ビット = 0:
 インタフェースにより読み出されるワードが 0x00000000 でない場合、消去命令が自動的に実行され、書き込み命令が続きます。この命令に要する時間は、2 tprog です。
 インタフェースにより読み出されるワードが 0x00000000 の場合、書き込み命令が直ちに実行されます (高速バイト書き込みと同じ時間、1 tprog を要します)。
 - ケース 2 : FTDW ビット = 1:
 消去命令が常に自動的に実行され、書き込み命令が続きます。この動作に要する時間は、2 tprog です。

注： この機能は、すべてのメモリから実行することができます。同じバンクから実行すると、書き込みは 1 Tprog 間停止します。

表 14. データ EEPROM 書き込み時間

-	FTDW ビット	ワード消去ステート	書き込み時間	コメント
データ EEPROM 高速 ワード/ ハーフワード/バイト 書き込み	0	消去済みワード	1 Tprog	ユーザソフトウェアがデータ EEPROM ダブルワード/ワード消去を使って選択 したワードを消去済み
データ EEPROM ワード/ ハーフワード/バイト 書き込み	0	消去済みワード	1 Tprog	インタフェースから読み出されるワード が 0x0 に一致 ==> 消去不要、このケース は高速ワード/ハーフワード/バイト書 き込みと同じ
	0	未消去ワード	2 Tprog	インタフェースから読み出されるワード が 0x0 に一致しない ==> 消去を自動的 に実行
	1	消去済みワード または未消去ワード	2 Tprog	インタフェース (ECC モジュール) から 読み出されるワードによらず、消去を 自動的に実行

- 注：
- 1 プログラムメモリからデータワード、データハーフワードまたはデータバイトを書き込むとき、DCode と ICode は 1~3 tprog の間ロックされます。書き込みが終わると、コードの実行が再開されます。この動作を回避するときは、書き込み命令を SRAM、他のバンク、または外部メモリから実行する必要があります。
 - 2 非整列アドレスでデータワードまたはデータハーフワードに書き込みを行うときは、書き込み命令は 1 tprog より多くの時間を要します。
 - 3 プログラムメモリハーフページ書き込み、データ EEPROM ダブルワード消去、データ EEPROM ダブルワード書き込みの間、書き込まれるバンクに対するすべての読み出し命令が禁止されます（これには、DMA 読み出し命令と、ブレークポイントや周期的更新などのデバッグ読み出し命令が含まれます）。
 - 4 プログラムメモリハーフページ書き込み時、またはデータ EEPROM ダブルワード書き込み時に PGAERR がセットされると、その書き込み命令全体が停止されます。ソフトウェアから FPRG ビットと PROG/DATA ビットをリセットし、書き込み命令を初めからやり直す必要があります。

3.4.4 書き込み中読み出し (RWW)

Cat.4とCat.5 デバイスでは、フラッシュモジュールは 2 バンクで構成されます。これらの 2 バンクは同じです。各々には次が含まれます。

- プログラムメモリ (最大 256 KB)
- データ EEPROM メモリ (最大 8 KB)
- 情報ブロック (システムメモリ (4 KB)、およびオプションバイト)

このアーキテクチャにより次が可能になります。

- バンク 1 の読み出しとバンク 2 の書き込み (プログラムまたはデータマトリクス)
- バンク 2 の読み出しとバンク 1 の書き込み (プログラムまたはデータマトリクス)
- バンク 1 の読み出しとバンク 2 の読み出し
- 制約付きのバンク 1 の書き込みとバンク 2 の書き込み
 - － マス消去時
 - － パラレルハーフページ書き込み時
 - － パラレルページ消去時

他のすべてのライトホワイルライト機能は正確に同時に実行することはできないため、メモリインタフェースで管理されます。

- 1 つのバンクの書き込み中に、他のバンクへの書き込みアクセスが発生すると、最初の書き込みが終わり、2 番目の書き込みが実行できるようになるまで、バスが停止します。
- あるバンクが複数の書き込みアクセスを実行中に、他のバンクで書き込みアクセスが発生すると、WRPERR がセットされて、新しい書き込みリクエストは中止されます。

同じ方法で、読み出しアクセスと書き込みアクセスは正確に同時に実行することはできませんが、これらはメモリインタフェースで次のように管理されます。

- あるバンクへの書き込み中に、このバンクへの読み出しアクセスが発生すると、書き込みが終わり、読み出しが実行できるようになるまで、バスが停止します。
- 複数書き込みアクセスであるバンクへの書き込み中に、このバンクへの読み出しアクセスが発生すると、BUS ERROR が返されます。

注意： 1 つのバンク内での複数書き込みの実行は禁止されています (1 つのバンクから同じバンクへハーフページ、ダブルワード)。

表 15. リードホワイルライトの要約

実行場所	動作	デスティネーション	コメント
PROG 1 (バンク 1)	ワード消去 高速バイト書き込み 高速ハーフワード書き込み バイト書き込み ハーフワード書き込み	DATA2 (RWW) DATA1 (Tprog ペナルティあり)	-
	高速ワード書き込み ワード書き込み	DATA2 (RWW) PROG 2 (RWW) DATA1 (Tprog ペナルティあり) PROG 1 (Tprog ペナルティあり)	
	ページ消去	PROG 2 (RWW) PROG 1 (Tprog ペナルティあり)	
	ダブルワード消去 ダブルワード書き込み	DATA2 (RWW) DATA 1 (禁止)	DATA1 または PROG 1 に対する新しい書き込みアクセス => WRPERR
	ハーフページ書き込み	PROG2 (RWW) PROG1 (禁止)	バンク 2 に対する読み出しアクセス => HARD フォルト (バスエラー)
PROG 2 (バンク 2)	ワード消去 高速バイト書き込み 高速ハーフワード書き込み バイト書き込み ハーフワード書き込み	DATA1 (RWW) DATA2 (Tprog ペナルティあり)	-
	高速ワード書き込み ワード書き込み	DATA1 (RWW) PROG 1 (RWW) DATA 2 (Tprog ペナルティあり) PROG 2 (Tprog ペナルティあり)	
	ページ消去	PROG 1 (RWW) PROG 2 (Tprog ペナルティあり)	
	ダブルワード消去 ダブルワード書き込み	DATA 1 (RWW) DATA 2 (禁止)	DATA 2 または PROG 2 に対する新しい書き込みアクセス => WRPERR
	ハーフページ書き込み	PROG 1 (RWW) DATA 2 (禁止)	バンク 1 に対する読み出しアクセス => HARD フォルト (バスエラー)
RAM および 外部メモリ	次を含む全動作: ParallPageErase ParallHalfPageWrite MassErase	PROG 1 DATA 1 PROG 2 DATA 2	-

アライメントエラーフラグ

フラッシュメモリインタフェースは、次の 3 種類のアライメントをチェックします。

- ロードする先頭アドレスがハーフページの開始アドレス（128 バイトの倍数）であり、ロードする残りの 31 ワードが同じハーフページ内にある場合にのみ、ハーフページがプログラムメモリへ書き込まれます。
- ロードする最初のアドレスがダブルワードの開始アドレス（8 バイトの倍数）である場合にのみ、ダブルワードがデータ EEPROM へ書き込まれます。
- ハーフページ書き込み中は、ページの変更はできません。

アライメントチェックが正しくない場合、PGAERR フラグ（FLASH_SR[8]）がセットされ、割り込みを生成することができます。書き込み動作は停止します。

サイズエラーフラグ

書き込みおよび消去プロシージャ中に、フラッシュメモリインタフェースはデータサイズをチェックして書き込むデータサイズと許容命令との間のコヒーレンスを確認します。

表 16. 禁止命令

メモリブロック	データサイズ
プログラムメモリ	バイト／ハーフワード書き込みを禁止 バイト／ハーフワード／ワード消去を禁止
オプションバイトブロック	バイト／ハーフワードを禁止

チェックが正しくない場合、フラグ **SIZERR**（FLASH_SR[9]）がセットされ、割り込みを生成することができます。

バスエラー（Cortex[®]-M3ハードフォルトまたはバスフォルト）

バスエラー（Cortex[®]-M3 ハードフォルトまたはバスフォルト）が次の 3 つのケースで返されます。

- メモリが読み出し保護されているとき、読み出しアクセスが D バスまたは I バスを介して実行された、さらにデバッグ機能が接続されている間またはブートを SRAM から実行中
- DATA EEPROM/FLASH_PECR レジスタ／プログラムメモリバイトアンロックシーケンスが正しくない。詳細については、[セクション 3.4.1: メモリのロック/アンロック](#)を参照してください。
- Cat.4デバイスとCat.5 デバイスの場合、同じバンクでダブルワード消去、ダブルワード書き込み、またはハーフページ書き込みの実行中に、バンク 1 またはバンク 2 に対する読み出しアクセスが実行された場合

3.5 オプションバイトの説明

フラッシュメモリモジュールの一部が、オプションバイトのセットを格納するために予約されます。これらのオプションバイトは、製品の設定情報を格納するため、エンドアプリケーションの要求に従って設定されます。設定例として、ハードウェアまたはソフトウェアモードでウォッチドッグを選択することができます。

オプションバイトブロックでは、32 ビットワードが次表のようにマップされます。

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
相補オプションバイト 1								相補オプションバイト 0								オプションバイト 1								オプションバイト 0							

オプションブロック内部のバイト構成を表 17 に示します。

表 17. オプションバイト構成

アドレス	[31:24]	[23:16]	[15:8]	[7:0]
0x1FF80000	0xFF	nRDP	0x00	RDP
0x1FF80004	0xFF	nUSER	0x00	USER
0x1FF80008	nWRP1[15:8]	nWRP1[7:0]	WRP1[15:8]	WRP1[7:0]
0x1FF8000C	nWRP1[31:24]	nWRP1[23:16]	WRP1[31:24]	WRP1[23:16]
0x1FF80010	nWRP2[15:8]	nWRP2[7:0]	WRP2[15:8]	WRP2[7:0]
0x1FF80014	nWRP2[31:24]	nWRP2[23:16]	WRP2[31:24]	WRP2[23:16]
0x1FF80018	nWRP3[15:8]	nWRP3[7:0]	WRP3[15:8]	WRP3[7:0]
0x1FF8001C	nWRP3[31:24]	nWRP3[23:16]	WRP3[31:24]	WRP3[23:16]
0x1FF80080	nWRP4[15:8]	nWRP4[7:0]	WRP4[15:8]	WRP4[7:0]
0x1FF80084	nWRP4[31:24]	nWRP4[23:16]	WRP4[31:24]	WRP4[23:16]

オプションバイトのローディングは次の 2 つのケースで行われます。

- OBL_LAUNCH がセットされた場合（このケースでは、リセットが発生）
- V 18 ドメインの各パワーアップ後（すなわち POR またはスタンバイ後）

オプションバイトローダ（OBL）が情報ブロックを読み出して、データをオプションバイトレジスタ（FLASH_OBR）へ格納します。

オプションバイトローディングプロセスで、オプションバイトとその補数をチェックして、ローディング動作が成功したことを確認することができます。

確認に失敗した場合、OPTVERR ステータスビットがセットされ、ERRIE がセットされている場合割り込みが発生します。

読み出しモードで、オプションバイトレジスタは CPU からアクセスすることができます。

表 18. オプションバイトの説明

メモリアドレス	オプションバイト
0x1FF8 0000	ビット [23:16] : nRDP ビット [7:0] : RDP : 読み出し保護機能オプションバイト (FLASH_OBR[22:16] に格納) 読み出し保護機能を使ってフラッシュメモリに格納されているソフトウェアコードを保護します。 0xAA : レベル 0、保護機能なし 0xCC : レベル 2、チップ保護機能 (SRAM 機能でのデバッグとブートを無効化) その他 : レベル 1、メモリの読み出し保護機能 (デバッグ機能を制限)
0x1FF8 0004	ビット [23:16] nUSER ビット 8 : SPRMOD : セクタ保護機能モード選択 0 : WRPx[i]=1 ビットはセクタ書き込み保護機能を指定 1 : WRPx[i]=0 ビットはセクタ書き込みと読み出し (PCROP) 保護機能を指定 詳細については、セクション 3.7.4 : PCROPを参照してください。 ビット [7:0] USER : ユーザオプションバイト (FLASH_OBR[23:16] に格納) このバイトを使って次の機能を設定します。 – ブラウンアウトリセット閾値レベルの選択 – ウォッチドッグイベントの選択 : ハードウェアまたはソフトウェア – CPU が Stop モードになったときイベントをリセット – CPU がスタンバイモードになったときイベントをリセット ビット 3:0 : BOR_LEV[3:0] : ブラウンアウトリセット 閾値レベル ビット 4 : IWDG_SW 0 : ハードウェア非依存ウォッチドッグ 1 : ソフトウェア非依存ウォッチドッグ ビット 5 : nRST_STOP 0 : CPU が Stop モードになったときリセットを発生 1 : リセットを発生しない ビット 6 : nRST_STDBY 0 : CPU がスタンバイモードになったときリセットを発生 1 : リセットを発生させない ビット 7 : BFB2 このビットは、Cat.4とCat.5 デバイスでのみ使用可能。 このビットを使ってバンク 2 からブートします。この値に応じてシステムメモリまたはバンク 1 から CPU がブート可能で、その後ブートローダは値 @08000000 または @08030000 に応じて、バンク 1 またはバンク 2 へのジャンプを実行します。 0 : システムメモリからのブート 1 : バンク 1 からのブート
0x1FF8 0008	WRP1 : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP1[15:0] ビット [15:0] : WRP1[15:0] は FLASH_WRPR1[15:0] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ

表 18. オプションバイトの説明 (続き)

メモリアドレス	オプションバイト
0x1FF8 000C	WRP1 : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP1[31:16] ビット [15:0] : WRP1[31:16] は FLASH_WRP1[31:16] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 0010	WRP2⁽¹⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP2[15:0] ビット [15:0] : WRP2[15:0] は FLASH_WRP2[15:0] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 0014	WRP2⁽¹⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP2[31:16] ビット [15:0] : WRP2[31:16] は FLASH_WRP2[31:16] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 0018	WRP3⁽²⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP3[15:0] ビット [15:0] : WRP3[15:0] は FLASH_WRP3[15:0] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 001C	WRP3⁽²⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP3[31:16] ビット [15:0] : WRP3[31:16] は FLASH_WRP3[31:16] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 0080	WRP4⁽³⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP4[15:0] ビット [15:0] : WRP4[15:0] は FLASH_WRP4[15:0] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ
0x1FF8 0084	WRP4⁽³⁾ : メモリ書き込み保護機能オプションバイト ビット [31:16] : nWRP4[31:16] ビット [15:0] : WRP4[31:16] は FLASH_WRP4[31:16] に格納) 0 : 選択したセクタで書き込み保護機能が非アクティブ 1 : 選択したセクタで書き込み保護機能がアクティブ

- WRP2 は Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能。
- WRP3 は Cat.4 と Cat.5 デバイスでのみ使用可能。
- WRP4 は、Cat.5 デバイスでのみ使用可能。

3.5.1 オプションバイトブロックの書き込み

オプションバイトブロックでは、高速ワード書き込み、ワード書き込み、ワード消去のみが可能です。

オプションバイトは、プログラム／データ EEPROM アドレスと同じ方法で書き込まれません。

次の 2 つのアンロックシーケンスが必要です。

- FLASH_PECR レジスタのアンロック
- オプションバイトブロックのアンロック

オプションバイトを変更するときは、次のステップに従う必要があります。

- 与えられたワードの 2 オプションバイトは、同時に書き込む必要があります。
- 与えられたワードの 2 つの相補オプションバイトを計算して、同時に書き込む必要があります (32 ビットワードでのオプションバイトのマッピング詳細については、[セクション 3.5 : オプションバイトの説明 \(73 ページ\)](#) を参照してください)。
- ユーザは要求に応じた設定を行うために、オプションバイトへ書き込むことができます。
- オプションバイトのローディングにより、オプションバイトレジスタ内でこれらを自動的に更新するときは、FLASH_PECR レジスタの OBL_LAUNCH をセットして、システムリセットを発生させます。
- 新しいオプションバイトを書き込めるようにするためには、オプションバイトエラーフラグをクリアする必要があります。

次の表に、書き込み機能と消去機能をまとめます。

注： オプションバイトは対応する相補バイトと一緒に正しく書き込み済みの場合にのみ、ロードされません。

3.6 書き込み／消去機能のクイックリファレンス

表 19. 書き込み／消去機能 (Cat.1、Cat.2、Cat.3デバイス)

	動作	ブロック	ビット／プロシージャ	時間
消去動作	ワード消去 ⁽¹⁾	データ EEPROM オプションバイト	値 0x0000 0000 をアドレス へ直接書き込みます	1 tprog
	ページ消去 ⁽²⁾	プログラムメモリ	ERASE = 1 PROG = 1	1 tprog
	ダブルワード消去 ^{(3) (4)}	データ EEPROM	FPRG = 1 DATA = 1	1 tprog
	マス消去	プログラムメモリ + データ EEPROM + オプションバイト + バックアップレジスタ (RTC)	RDP : レベル 1 -> レベル 0	消去用の 2 tprog + 書き込み用の 1 tprog
書き込み動作	高速ワード書き込み	プログラムメモリ データ EEPROM オプションバイト	FTDW = 0	1 tprog
	ワード書き込み ⁽⁵⁾	データ EEPROM オプションバイト	FTDW = 1 または 0	1 または 2 tprog
	ハーフページ書き込み ^{(3) (6)}	プログラムメモリ	FPRG = 1 PROG = 1	1 tprog
	ダブルワード書き込み ⁽³⁾⁽⁴⁾	データ EEPROM	FPRG = 1 DATA = 1	1 tprog
	高速バイト書き込み ^{(7) 高速ハーフワード書き込み ⁽⁷⁾⁽⁵⁾}	データ EEPROM	FTDW = 0	1 tprog
	バイト書き込み ^{(7) ハーフワード書き込み ⁽⁷⁾⁽⁵⁾}	データ EEPROM	FTDW = 1 または 0	1 または 2 tprog

- ロードするアドレスがワードの開始アドレス（ワードの倍数）である場合にのみ、データ EEPROM 内でデータ EEPROM ワードが消去されます。
- ロードするアドレスがページの開始アドレス（256 バイトの倍数）の場合にのみ、プログラムメモリ内でページが消去されます。
- SRAM、代替バンクまたは外部メモリからのみ、ハーフページ書き込み、ダブルワード消去、ダブルワード書き込みが可能です。
- ロードする最初のアドレスがダブルワードの開始アドレス（ダブルワードの倍数）である場合にのみ、データ EEPROM でデータ EEPROM ダブルワードが書き込まれ、または消去されます。
- 非整列アドレスでデータワードまたはデータハーフワードに書き込みを行うときは、書き込み命令は 1 tprog より多くの時間を要します。
- ロードする先頭アドレスがハーフページの開始アドレス（128 バイトの倍数）の場合にのみ、ハーフページがプログラムメモリへ書き込まれます。
- 高速バイト書き込み、高速ハーフワード書き込み、バイト書き込み、ハーフワード書き込みは、NON NULL バイト／ハーフワードを書き込むときにのみ使用することができます。

表 20. 書き込み／消去機能（Cat.4とCat.5 デバイス）

	動作	ブロック	ビット／プロシージャ	時間
消去動作	ワード消去 ⁽¹⁾	データ EEPROM オプションバイト	値 0x0000 0000 をアドレス へ直接書き込みます	1 tprog
	ページ消去 ⁽²⁾	プログラムメモリ	ERASE = 1 PROG = 1	1 tprog
	ParallPageErase	プログラムメモリ	ERASE = 1 PROG = 1 PARALLBANK = 1	1 tprog
	ダブルワード消去 ^{(3) (4)}	データ EEPROM	FPRG = 1 DATA = 1	1 tprog
	マス消去	プログラムメモリ + データ EEPROM + オプションバイト + バックアップレジスタ (RTC)	RDP : レベル 1 -> レベル 0	消去用の 2 tprog + 書き込み用の 1 tprog
書き込み動作	高速ワード書き込み	プログラムメモリ データ EEPROM オプションバイト	FTDW = 0	1 tprog
	ワード書き込み ⁽⁵⁾	データ EEPROM オプションバイト	FTDW = 1 または 0	1 または 2 tprog
	ハーフページ書き込み ^{(3) (6)}	プログラムメモリ	FPRG = 1 PROG = 1	1 tprog
	ParallHalfPageWrite	プログラムメモリ	FPRG = 1 PARALLBANK = 1 PROG = 1	1 tprog
	ダブルワード書き込み ⁽³⁾⁽⁴⁾	データ EEPROM	FPRG = 1 DATA = 1	1 tprog
	高速バイト書き込み ⁽⁷⁾ 高速ハーフワード書き込み ⁽⁷⁾⁽⁵⁾	データ EEPROM	FTDW = 0	1 tprog
	バイト書き込み ⁽⁷⁾ ハーフワード書き込み ⁽⁷⁾⁽⁵⁾	データ EEPROM	FTDW = 1 または 0	1 または 2 tprog

- ロードするアドレスがワードの開始アドレス（ワードの倍数）である場合にのみ、データ EEPROM 内でデータ EEPROM ワードが消去されます。
- ロードするアドレスがページの開始アドレス（256 バイトの倍数）の場合にのみ、プログラムメモリ内でページが消去されます。
- SRAM、または外部メモリからのみ、ハーフページ書き込み、ダブルワード消去、ダブルワード書き込みが可能です。
- ロードする最初のアドレスがダブルワードの開始アドレス（ダブルワードの倍数）である場合にのみ、データ EEPROM でデータ EEPROM ダブルワードが書き込まれ、または消去されます。
- 非整列アドレスでデータワードまたはデータハーフワードに書き込みを行うときは、書き込み命令は 1 tprog より多くの時間を要します。
- ロードする先頭アドレスがハーフページの開始アドレス（128 バイトの倍数）の場合にのみ、ハーフページがプログラムメモリへ書き込まれます。
- Cat.1 デバイスとは対照的に、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでは、高速バイト書き込み、高速ハーフワード書き込み、バイト書き込み、ハーフワード書き込みは、NULL バイト／ハーフワードの書き込みに使用することができます。

3.7 メモリ保護

フラッシュメモリモジュールは、読み出しアクセスから保護することができます。

また、書き込みカウンタコンテキストの喪失から発生する不要な書き込みアクセスからメモリセクタを個別に保護することもできます。

3.7.1 プログラム EEPROM およびデータ EEPROM の読み出し保護機能 (RDP)

フラッシュメモリモジュール (データとプログラム) のユーザ領域は、読み出し動作から保護することができます。次の 3 つの読み出し保護機能レベルが定義されています。

- レベル 0 : 読み出し保護機能なし

読み出し保護機能オプションバイト RDP へ 0xAA を書き込んで、読み出し保護機能レベルをレベル 0 に設定すると、フラッシュメモリモジュールまたはバックアップ SRAM に対するすべての読み出し/書き込み動作 (書き込み保護機能なしを設定した場合) がすべてのブート設定 (デバッグ、選択した RAM またはシステムメモリからのブート) で可能になります。

レベル 1 : メモリ読み出し保護機能を有効化

これは、オプションバイト消去後のデフォルト読み出し保護機能レベルになります。任意の値 (レベル 0 とレベル 2 の設定に使うそれぞれ 0xAA と 0xCC を除きます) を RDP オプションバイトに書き込むと、読み出し保護機能レベル 1 が開始されます。読み出し保護機能レベル 1 を設定した場合:

- デバッグ機能が接続された場合または選択した RAM またはシステムメモリからのブートの場合に、フラッシュメモリモジュールアクセス (読み出し、消去、書き込み) はありません。フラッシュメモリ読み出しリクエストの場合、バスエラー (Cortex®-M3 ハードフォルトまたはバスフォルト) が発生します。フラッシュユーザブートを使用する場合、すべての動作が可能です。
- 保護機能オプションバイトを低い保護機能に設定すると、フラッシュメモリモジュールとバックアップレジスタ (RTC) がマス消去されます。すなわち、ユーザコード値がクリアされた後に読み出し保護機能が解除されます。

注 : レベル 1 がアクティブのときに、レベル 0 が要求された場合、次のステップが実行されます。

- マス消去が生成されます (RDP バイトが消去 (0x0) され、レベル 1 はアクティブのまま、コードは実行されなくなります)。
- OBL ランチがセットされた場合またはシステムリセットが生成された場合、新しい RDP バイトがロードされ (0xAA)、レベル 0 がアクティブになります。

注 : レベル 1 がアクティブのときにレベル 0 が要求されると、マス消去が実行されます。保護機能レベルを上げても (0->1、1->2、0->2) マス消去はありません。

- CPU デバッグ機能 (JTAG またはシングルワイヤ) が接続された場合、または RAM またはシステムメモリからのブートが選択された場合にも、フラッシュメモリモジュールは書き込み保護されます。

レベル 2 : メモリ読み出し保護機能の有効化と全デバッグ機能ディセーブル

注 : メモリ読み出し保護機能レベル 2 は、不可逆動作です。この場合の保護機能レベルは、レベル 0 またはレベル 1 へ下げることはできません。

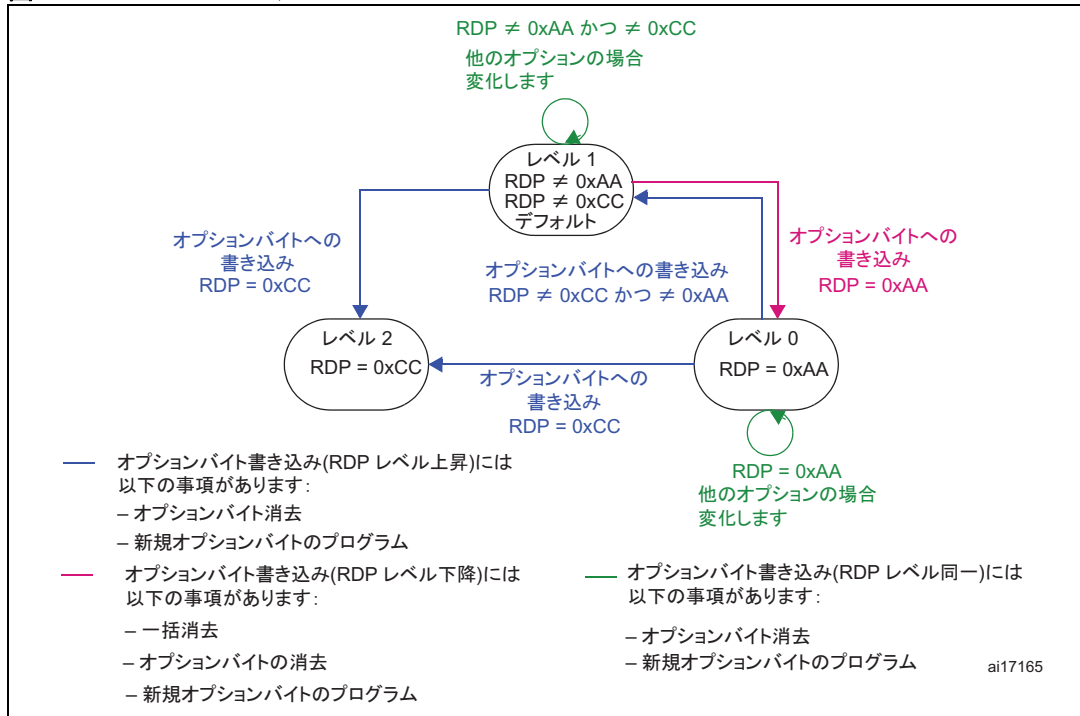
WRDP オプションバイトへ 0xCC を書き込んで、読み出し保護機能レベル 2 をアクティブにすると、レベル 1 で提供されるすべての保護機能がアクティブになり、SRAM またはシステムメモリからブートしたとき、システムメモリと全デバッグ機能 (CPU JTAG とシングルワイヤ) が無効化され、ユーザオプションを変更できなくなります。



注： レベル 2 では JTAG ポートが無効に固定されます（JTAG ヒューズとして機能）。このため、バウンダリスキャンが実行できなくなります。STMicroelectronics 社は、レベル 2 保護機能が設定された故障デバイスの解析ができません。

図 6：RDP レベルに、RDP レベルから別レベルへの移行方法を示します。

図 6. RDP レベル



RDP オプションバイトとその相補に次の一対の値が格納されている場合、フラッシュメモリモジュールは保護されます。

表 21. RDP とその相補に従うフラッシュメモリモジュール保護機能

RDP バイト値	RDP の相補値	読み出し保護機能ステータス
0xAA	0x55	レベル 0
0xCC	0x33	レベル 2
0xAA または 0xCC 以外の任意の値	RDP バイトの相補値	レベル 1
任意の値	RDP の非相補値	レベル 1

3.7.2 プログラムメモリの書き込み保護機能 (WRP)

書き込み保護機能単位はセクタ (16 ページ) です。これは、128 KBのプログラムメモリを保護するためには 32 オプションビットが必要なことを意味します。

WRPx オプションバイトを設定したとき、保護機能がアクティブになります。この WRPx オプションバイトはパワーアップ時にロードされるか、または FLASH_PECR レジスタの OBL_LAUNCH ビットをセットして再ロードされます。これらのオプションビットのコピーは FLASH_WRPx レジスタに格納されます。

注： *メモリ読み出し保護機能レベルを選択すると (RDP レベル = 1)、CPU デバッグ機能 (JTAG またはシングルワイヤー) が接続されている場合、または nWRPx = 0 のときでも RAM またはシステムメモリからのブートが選択された場合、プログラム EEPROM とデータ EEPROM の書き込みまたは消去ができなくなります。*

データ EEPROM は、WRP ビットによって保護されません。

3.7.3 書き込み保護機能エラーフラグ

書き込み保護されたメモリページに対して消去/書き込み命令が実行されると、FLASH_SR レジスタの書き込み保護機能エラーフラグ (WRPERR) がセットされます。このフラグは、保護されたすべてのアドレスに対するソフトウェアからの書き込みごとにセットされます。

したがって、ソフトウェアが次に対して書き込みを行うと WRPERR フラグがセットされます。

- 書き込み保護されたページ
- システムメモリページ
- PEKEY、PRGKEY または OPTKEY によりアンロックされていない場合の、プログラムメモリ、データ EEPROM またはオプションバイトブロック
- RDP オプションバイトがセットされ、かつデバイスがデバッグモードにある場合、または SRAM からのブートの場合のデータ EEPROM とプログラムメモリ
- ダブルワード消去、ダブルワード書き込みまたはハーフページ書き込みが他のバンクに対して実行されている場合の1つのバンク (Cat.4とCat.5 デバイスの場合)

3.7.4 PCROP

このセクションはCat.2とCat.3 デバイスにのみ適用されます。

フラッシュメモリのユーザ領域は、D バス読み出しアクセスから保護することができます。PCROP 機能の有無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。ここでは、PcROP オプションを使ってネイティブコードをコンパイルしたものと想定しています。

保護されたセクタはコード実行でアクセス可能ですが (I-bus バスのみ)、D バスでの読み出し/書き込みはできません (CPU データ読み出しまたはデバッグアクセス)。

このモードは、SPRMOD オプションバイトと WRPx オプションバイトにより、アクティブにされます。

- SPRMOD = 0 : WRPx はそれぞれのユーザセクタの書き込み保護機能を制御します。
- SPRMOD = 1 : WRPx はそれぞれのユーザセクタの読み出し保護機能 (PcROP) を制御します。

対称セクタが保護済みの場合、D バス経由のすべての読み出しアクセスにより、RDERR フラグエラーが発生します。

すべての読み出し保護されたセクタは書き込み保護もされているため、これらのセクタに対して書き込みアクセスを行うと、WRPERR が発生します (WRP 保護機能と同様)。



SPRMOD の非アクティブ化および／またはユーザセクタの非保護は、RDP がレベル-1 からレベル-0 へ変化する際にのみ発生します。SPRMOD がアクティブのとき、ユーザオプション変更により SPRMOD をリセットまたは WRPx を非アクティブ化しようとしても、書き込みは開始しますが、SPRMOD または保護されたセクタは変更されません。

注： **PCROP モードがアクティブの場合（SPRMOD =1）、WRPx ビットのアクティブ値が反転します。WRPx = "0" の場合、ユーザセクタ i は読み出し保護されます（PcROP）。**

3.8 割り込み

FLASH_PECR レジスタの書き込み終了割り込みイネーブルビット（EOPIE）をセットすると、消去または書き込み動作が正常終了したとき、割り込みの生成が可能になります。この場合、FLASH_SR レジスタの書き込み終了（EOP）ビットがセットされます。

FLASH_PECR レジスタのエラー割り込みイネーブルビット（ERRIE）をセットすると、書き込み動作または消去動作中、あるいはオプションバイトのローディング中にエラーが発生した場合に割り込みの生成が可能になります。この場合、FLASH_SR レジスタの次のエラーフラグの 1 つがセットされます。

- 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（書き込みアライメントエラーフラグ）
- OPTVERR（オプション有効エラーフラグ）
- OPTVERRUSR（ユーザオプション有効エラーフラグ）
- SIZERR（サイズエラーフラグ）

表 22. 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
書き込み終了	EOP	EOPIE
エラー	RDERR WRPERR PGAERR OPTVERR OPTVERRUSR SIZERR	ERRIE

3.9 レジスタの説明

3.9.1 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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
予約済み																												RUN_PD	SLEEP_PD	ACC64	PRFTEN	LATENCY
																												rw	rw	rw	rw	rw

ビット 31:5 予約済み、クリア状態を保つ必要があります。

ビット 4 **RUN_PD**: Run 時の省電力モード
 FLASH_PDKEYR への書き込みによりこのビットがアンロックされた場合にのみ、このビットは書き込み可能です。
 このビットは、STM32L1xxxxが Run モードのとき、フラッシュメモリモジュールがパワーダウンモードまたはアイドルモードのいずれにあるかを指定します。
 コードが SRAM から実行された場合にのみフラッシュメモリモジュールをパワーダウンモードにすることができます。
 0：フラッシュモジュールをアイドルモードにします。
 1：フラッシュモジュールをパワーダウンモードにします。

ビット 3 **SLEEP_PD**: Sleep 時の省電力モード
 STM32L1xxxxが Sleep モードの場合、このビットを使って、フラッシュメモリモジュールをパワーダウンモードまたはアイドルモードにします。
 0：フラッシュモジュールをアイドルモードにします。
 1：フラッシュモジュールをアイドルモードにします。

ビット 2 **ACC64** : 64 ビットアクセス
 このビットを使って、1 回に 64 ビットまたは 32 ビットのデータをメモリから読み出します。32 ビットアクセスはメモリ使用を減らすために使用されます。これに対して、64 ビットアクセスは性能を向上させるために使用されます。この場合、プリフェッチの有効化が役立ちます。
 0：32 ビットアクセス
 1：64 ビットアクセス
注： 32 ビットアクセスは低消費電力モードです。低周波数でのみ使用されます。すなわち、遅延が 0 ウェイトステートで、プリフェッチがオフで使用されます。
注： このビットは、LATENCY ビットおよびPRFTEN ビットと同時に書き込むことはできません。

ビット 1 **PRFTEN** : プリフェッチイネーブル
 0：プリフェッチを無効化
 1：プリフェッチを有効化
注： ACC64 がセットされている場合にのみ、プリフェッチを有効化することができます。
 ACC64 がセットされている場合にのみ、このビットをセットまたはクリアすることができます。

ビット 0 **LATENCY** : 遅延
 このビットは、メモリアクセス時間に対する CPU クロック周期の比を表します。
 0：ゼロウェイトステート
 1：1 ウェイトステート
注： ACC64 がセットされている場合にのみ、LATENCY をセットすることができます。
 ACC64 がセットされている場合にのみ、このビットをセットまたはクリアすることができます。

3.9.2 書き込み／消去制御レジスタ（FLASH_PECR）

このレジスタを使って、すべての消去動作と書き込み動作を実行します。FLASH_PECR で正しい書き込みシーケンスが実行された後にのみ、書き込みアクセス可能になります。

アドレスオフセット：0x04

リセット値：0x0000 0007

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	
予約済み													OBL_LAUNCH	ERRIE	EOPIE	PARALLEL BANK	予約済み					FPRG	ERASE	FTDW				データ	PROG	OPTLOCK	PRGLOCK	PELOCK
													rw1	rw	rw	rw						rw	rw				rw	rw	rs	rs	rs	

ビット 31:19 予約済み、クリア状態を保つ必要があります。

ビット 18 **OBL_LAUNCH**：オプションバイトローディングを開始します。

ソフトウェアからこのビットをセットして、オプションバイトローディングを開始します。オプションバイトローディングが完了した後にのみ、このビットはクリアされます。OPTLOCK がセットされている場合、このビットへの書き込みはできません。

このビットがセットされると、リセットが発生します。

0：オプションバイトローディングが完了

1：オプションバイトのローディングが必要です

ビット 17 **ERRIE**：エラー割り込みイネーブル

0：割り込みを無効化

1：割り込みを有効化

ビット 16 **EOPIE**：書き込み終了割り込みイネーブル

0：割り込みを無効化

1：割り込みを有効化

ビット 15 **PARALLBANK**：パラレルバンクモード。

このビットを使って、進行中の書き込みまたは消去処理がない場合、ソフトウェアからセットまたはクリアすることができます。

このビットがセットされると、並行して 2 ハーフページを書き込むか、または 2 ページを消去することができます（最初のバンクの最初のハーフページと 2 番目バンクの 2 番目のハーフページ）。

0：パラレルバンクモードを無効化

1：パラレルバンクモードを有効化

注： このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 14:11 予約済み、クリア状態を保つ必要があります。

ビット 10 **FPRG**：ハーフページ／ダブルワード書き込みモード

進行中の書き込みまたは消去処理がない場合、ソフトウェアからこのビットを書き込むことができます。

このビットを使って、プログラムメモリに対するハーフページ書き込み、またはデータ EEPROM に対するダブルワード書き込みを有効化/無効化します。

プログラムメモリに対するハーフページ書き込みには 32 回のローディングが必要です。

データ EEPROM に対するダブルワード書き込みには 2 回のローディングが必要です。

PELOCK がセットされているときにこのビットはクリアされます。

0：ハーフページ／ダブルワード書き込みを無効化

1：ハーフページ／ダブルワード書き込みを有効化

ビット 9 ERASE : ページまたはダブルワード消去モード

進行中の書き込みまたは消去処理がない場合、ソフトウェアからこのビットを書き込むことができます。

このビットを使って、プログラムメモリのページ消去、またはデータ EEPROM とオプションバイトブロックのダブルワード消去を有効化/無効化します。

PELOCK がセットされているときにこのビットはクリアされます。

0 : ページまたはダブルワード消去を無効化

1 : ページまたはダブルワード消去を有効化

ビット 8 FTDW : バイト、ハーフワード、ワード書き込みに対する固定時間データ書き込み

進行中の書き込みまたは消去処理がない場合、ソフトウェアからこのビットを書き込むことができます。

このビットは、データ EEPROM に対してのみ使用することができます。

PELOCK がセットされているときにこのビットはクリアされます。

0 : 事前の消去動作なしにバイト、ハーフワードまたはワードの書き込みを行います。これは、ワードの 0x0000 0000 への書き込み中に可能です。

1 : バイト、ハーフワード、ワードの書き込み前に、消去フェーズが自動的に実行されます。このため、書き込み時間は固定で、 $2 t_{\text{prog}}$ になります。

ビット 7:5 予約済み、クリアされた状態に保つ必要があります。**ビット 4 DATA** : データ EEPROM の選択

進行中の書き込みまたは消去処理がない場合、ソフトウェアからこのビットを書き込むことができます。

データ EEPROM のダブルワード消去/書き込みの前に、このビットをセットする必要があります。

PELOCK がセットされているときにこのビットはクリアされます。

0 : データ EEPROM を非選択

1 : データ EEPROM を選択

ビット 3 PROG : プログラムメモリの選択

進行中の書き込みまたは消去処理がない場合、ソフトウェアからこのビットを書き込むことができます。

プログラムメモリに対する書き込みアクセスを得るためにはこのビットをセットする必要があります。ただし、ワード書き込みの場合は除きます。

PELOCK がセットされているときにこのビットはクリアされます。

0 : プログラムメモリを非選択

1 : プログラムメモリを選択

ビット 2 OPTLOCK : オプションバイトブロックのロック

このビットには 1 のみを書き込むことができます。このビットがセットされると、オプションバイトブロックがロックされたことを表示します。

このビットは、アンロックシーケンスが検出された後にハードウェアからクリアされます。アンロック動作が不成功のとき、または OPTKEYR に対する 3 回目のアクセスのとき、バスエラー（Cortex®-M3 ハードフォルトまたはバスフォルト）が発生して、このビットは、次のリセットまで、セットされたままになります。

PELOCK がセットされているときにこのビットはセットされます。

0 : オプションをアンロック

1 : オプションをロック

ビット 1 PRGLOCK : プログラムメモリのロック

このビットには 1 のみを書き込むことができます。このビットがセットされると、プログラムメモリへの書き込みができないことを示します。このビットは、アンロックシーケンスが検出された後にハードウェアからクリアされます。アンロック動作が不成功のとき、または PRGKEYR に対する 3 回目のアクセスのとき、バスエラー（Cortex®-M3 ハードフォルトまたはバスフォルト）が発生して、このビットは、次のリセットまで、セットされたままになります。

PELOCK がセットされているときにこのビットはセットされます。

0 : プログラムメモリをアンロック

1 : プログラムメモリをロック

ビット 0 PELOCK : FLASH_PECR とデータ EEPROM のロック

このビットには 1 のみを書き込むことができます。このビットがセットされると、FLASH_PECR レジスタとデータ EEPROM がロックされたことを表示します。このビットは、アンロックシーケンスが検出された後にハードウェアからクリアされます。アンロック動作が不成功のとき、または PEKEYR に対する 3 回目のアクセスのとき、バスエラー（Cortex®-M3 ハードフォルトまたはバスフォルト）が発生して、このビットは、次のリセットまで、セットされたままになります。

このビットがクリアされると、データ EEPROM への書き込みアクセスが許可されます。

0 : FLASH_PECR をアンロック

1 : FLASH_PECR をロック

3.9.3 パワーダウンキーレジスタ（FLASH_PDKEYR）

パワーダウンキーレジスタを使って、FLASH_ACR の RUN_PD ビットをアンロックします。

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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PDKEYR[31:0]																															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 PDKEYR[31:0] : FLASH_ACR キー内の RUN_PD

これらのビットは、FLASH_ACR レジスタの RUN_PD ビットをセットするときに使うキーを表します。

PDKEY1 : 0x04152637

PDKEY2 : 0xFAFBFCFD

3.9.4 書き込み／消去キーレジスタ（FLASH_PEKEYR）

書き込み／消去キーレジスタを使って、FLASH_PECR に対するアクセスを許可します。したがって、データ EEPROM に対する書き込み動作と消去動作も許可されます。

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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PEKEYR[31:0]																															
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

ビット 31:0 **PEKEYR[31:0]**：FLASH_PEC とデータ EEPROM キー

これらのビットは、FLASH_PECR レジスタとデータ EEPROM に対する書き込みアクセスをアンロックするキーを表します。

PEKEY1：0x89ABCDEF

PEKEY2：0x02030405

3.9.5 プログラムメモリキーレジスタ（FLASH_PRGKEYR）

プログラムメモリキーレジスタを使って、プログラムメモリに対する書き込み動作と消去動作を許可します。FLASH_PEKEYR で正しい書き込みシーケンスが実行された後にのみ書き込みアクセスが可能になります。

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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRGKEYR[31:0]																															
W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

ビット 31:0 **PRGKEYR[31:0]**：プログラムメモリキー

これらのビットは、プログラムメモリをアンロックするキーを表します。

PRGKEY1：0x8C9DAEBF

PRGKEY2：0x13141516

3.9.6 オプションバイトキーレジスタ（FLASH_OPTKEYR）

オプションキーレジスタを使って、オプションバイトブロックに対する書き込み動作と消去動作を許可します。FLASH_PEKEYR で正しい書き込みシーケンスが実行された後にのみ書き込みアクセスが可能になります。

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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTKEYR[31:0]																															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31:0 **OPTKEYR**：オプションバイトキー

これらのビットは、オプションバイトブロックに対する書き込みアクセスをアンロックするキーを表します。

OPTKEY1:0xFBEAD9C8

OPTKEY2:0x24252627

3.9.7 ステータスレジスタ（FLASH_SR）

アドレスオフセット：0x18

システムリセット値：0b0000 0000 0000 0000 000x x000 0000 0100

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
予約済み																		RDERR	OPTVERRUSR	OPTVERR	SIZERR	PGAERR	WRPERR	予約済み				READY	ENDHV	EOP	BSY
																		rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1					r	r	r	r

ビット 31:14 予約済み、クリア状態を保つ必要があります。

ビット 13 **RDERR**：読み出し保護エラー

D バス経由で読み出されるアドレスが、メモリの読み出し保護（PCROP）部分に属している場合に、ハードウェアからセットされます。

1 を書き込むとクリアされます。

ビット 12 **OPTVERRUSR**：オプション UserValidity エラー

ユーザオプション読み出しがユーザが設定したものでない場合にハードウェアからセットされます。

1 を書き込むとクリアされます。

オプションが正常にロードされていない場合、システムリセットごとに、OPTVERRUSR が再度セットされます。したがって、ERRIE がセットされると直ちに割り込みが発生します。

注： このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 11 **OPTVERR**：オプション有効エラー

オプション読み出しがソフトウェアから設定したものでない場合にハードウェアからセットされます。1 を書き込むとクリアされます。

オプションが正常にロードされていない場合、システムリセットごとに、OPTVERR はロジックレベル 1 になります。したがって、ERRIE がセットされるごとに割り込みが発生します。

ビット 10 SIZERR : サイズエラー

書き込むデータサイズが禁止されている場合に、ハードウェアからセットされます。
1 を書き込むとクリアされます。

ビット 9 PGAERR : 書き込みアライメントエラー

書き込むデータが所定のハーフページまたはダブルワードに収まらない場合に、ハードウェアからセットされます。
1 を書き込むとクリアされます。

ビット 8 WRPERR : 書き込み保護エラー

消去／書き込み対象アドレスが、メモリの書き込み保護部分に属している場合に、ハードウェアからセットされます。
1 を書き込むとクリアされます。

ビット 7:4 予約済み、クリア状態を保つ必要があります。**ビット 3 READY : 低消費電力モード後のフラッシュメモリモジュールレディ**

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。

0 : フラッシュメモリモジュールが非レディ

1 : フラッシュメモリモジュールがレディ

ビット 2 ENDHV : 高電圧終了

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。

0 : 書き込み／消去動作中高電圧を供給

1 : 高電圧を終了

ビット 1 EOP : 動作終了

高電圧の供給が停止し、書き込みが中断されない場合に、このビットはハードウェアからセットされます。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 : EOP イベントは発生していません。

1 : EOP イベントが発生 EOPIE がセットされている場合、割り込みが発生します。

ビット 0 BSY : 書き込み／消去動作の進行中

0 : 書き込み／消去動作が非進行中

1 : 書き込み／消去動作が進行中

3.9.8 オプションバイトレジスタ（FLASH_OBR）

アドレスオフセット：0x1C

リセット値：RDP および USER オプションバイトに依存、バージョン製品の初期値は 0x00F8 0000。

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	
予約済み								BFB2	nRST_STDBY	nRST_STOP	IWDG_SW	BOR_LEV[3:0]				予約済み								SPRMOD	RDPRT							
								r	r	r	r	r	r	r	r									r								

ビット 31:24 予約済み、クリア状態を保つ必要があります。

ビット 23 **BFB2**：バンク 2 からのブート。このビットには、OBL によりロードされるユーザオプションバイトが含まれます。

バンク 2 からブートするときこのビットを使います。実際には、このビットはシステムメモリまたはバンク 1 からのブートがオプション設定で選択されていることを表示します。次に、@08000000 と @08030000 の値に応じて、ソフトウェアによりバンク 1 またはバンク 2 への分岐が実行されます。

0：システムメモリからのブート

1：バンク 1 からのブート

このビットは読み出し専用です。

注： このビットは、Cat.4 と Cat.5 デバイスでのみ使用可能です。

ビット 22:16 ユーザオプションバイト

これらのビットには、OBL によりロードされるユーザオプションバイトが含まれます。

ビット 22：nRST_STDBY

ビット 21：nRST_STOP

ビット 20：IWDG_SW

ビット 19:16: BOR_LEV[3:0]：ブラウンアウトリセット閾値レベル

0xxx：BOR OFF：1.45 V～1.55 V 電圧範囲に対するリセット閾値レベル（パワーダウンの場合）

この特別なケースでは、BOR を無効化するためにデバイス OBL シーケンスを開始させる BOR LEVEL 1 を V_{DD33} が超える必要があります。次に、PDR によりパワーダウンをモニタします。

注： BOR を無効化した場合、1.65 V と VPDR 閾値との間に「グレーゾーン」が存在します（これは、 V_{PDR} 閾値を通過するまでリセットを発生することなく、 V_{DD33} が最小動作電圧（1.65 V）を下回っていることを意味します）。

1000：BOR LEVEL 1：1.69 V～1.8 V 電圧範囲（パワーオン）に対するリセット閾値レベル

1001：BOR LEVEL 2：1.94 V～2.1 V 電圧範囲（パワーオン）に対するリセット閾値レベル

1010：BOR LEVEL 3：2.3 V～2.49 V 電圧範囲（パワーオン）に対するリセット閾値レベル

1011：BOR LEVEL 4：2.54 V～2.74 V 電圧範囲（パワーオン）に対するリセット閾値レベル

1100：BOR LEVEL 5：2.77 V～3.0 V 電圧範囲（パワーオン）に対するリセット閾値レベル

これらのビットは読み出し専用です。

ビット 15:9 予約済み、クリア状態を保つ必要があります。

ビット 8 **SPRMOD** : セクタ保護機能モードの選択

0 : FLASH_WRPx レジスタで WRP[i]=1 ビットに設定すると、セクタ書き込み保護機能が指定されます。

1 : FLASH_WRPx レジスタで WRP[i]=0 ビットに設定すると、セクタ書き込みおよび読み出し (PCROP) 保護機能が指定されます。

詳細については、[セクション 3.7.4 : PCROP](#)を参照してください。

ビット 7:0 **RDPRT[7:0]** : 読み出し保護機能

これらのビットには、OBL によりロードされる読み出し保護オプションレベルが含まれます。

0xAA : レベル 0、読み出し保護機能が非アクティブ

0xCC : レベル 2、読み出し保護機能がアクティブ

その他 : レベル 1、メモリの読み出し保護機能がアクティブ。オプションのバイト消去後のデフォルト設定

3.9.9 書き込み保護機能レジスタ (FLASH_WRPx)

アドレスオフセット : 0x20、0x80、0x84、0x88

リセット値 : オプションバイト WRPx の値に依存、バージン製品の初期値は 0x0000 0000。

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
WRPx[31:0]																															
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	x	x	x	

ビット 31:0 **WRPx[n]** : 書き込み保護機能、n は関係するメモリセクタ番号

これらのビットには、OBL によりロードされる書き込み保護オプションが含まれます。

0 : セクタ n は非書き込み保護

1 : セクタ n は書き込み保護

注 : *FLASH_WRP2 はCat.3、Cat.4、Cat.5 デバイスでのみ使用可能。*

FLASH_WRP3 はCat.4 とCat.5 デバイスでのみ使用可能。

FLASH_WRP4 はCat.5 デバイスのみ使用可能。

3.9.10 レジスタマップ

次の表に、レジスタマップとリセット値の一覧を示します。

表 23. 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	予約済み																												RUN_PD	SLEEP_PD	Acc64	PRFTEN	LATENCY0	
	リセット値： 0x0000 0000																													0	0	0	0	0	
0x04	FLASH_PECR	予約済み														OBL_launch	ERRIE	EOPIE	PARALLELBANK	予約済み				FPRG	ERASE	FTDW	予約済み				データ	PRG	OPTLOCK	PRGLOCK	PELOCK
	リセット値： 0x0000 0007															0	0	0	0					0	0	0					0	0	1	1	1
0x08	FLASH_PDKEYR	PDKEYR[31:0]																																	
	リセット値： 0x0000 0000	0	0	0	0	0	0	0	0	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	FLASH_PEKEYR	PEKEYR[31:0]																																	
	リセット値： 0x0000 0000	0	0	0	0	0	0	0	0	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	FLASH_PRGKEYR	PRGKEYR[31:0]																																	
	リセット値： 0x0000 0000	0	0	0	0	0	0	0	0	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	FLASH_OPTKEYR	OPTKEYR[31:0]																																	
	リセット値： 0x0000 0000	0	0	0	0	0	0	0	0	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	FLASH_SR	予約済み																			RDERR	OPTVERRUSR	OPTVERR	SIZVERR	PGAERR	WRPERR	予約済み				READY	ENDHV	EOP	BSY	
	リセット値： 0x0000 0004																				0	0	0	0	0	0					0	1	0	0	
0x1C	FLASH_OBR	予約済み										BFB2	nRST_STDBY	nRST_STOP	IWDG_SW	BOR_LEV [3:0]			予約済み							RDPRT[7:0]									
	リセット値： 0x xxxx xxxx											X	x	x	x	x	x	x	x								x	x	x	x	x	x	x	x	x
0x20	FLASH_WRP1	WRP1[31:0]																																	
	リセット値： 0x xxxx xxxx	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	x	x	x	x		
0x80	FLASH_WRP2	WRP2[31:0]																																	
	リセット値： 0x xxxx xxxx	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	x	x	x	x		
0x84	FLASH_WRP3	WRP3[31:0]																																	
	リセット値： 0x xxxx xxxx	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	x	x	x	x		

表 23. 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
0x88	FLASH_WRP4	WRP4[31:0]																															
	リセット値: 0x xxxx xxxx	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	x	x	x	

4 CRC 計算ユニット

このセクションは、特に指定がない限り、STM32L1xxxファミリ全体に適用されます。

4.1 CRC の概要

CRC (Cyclic Redundancy Check) 計算ユニットは、32 ビットデータワードと決められた生成多項式から CRC コードを発生するときに使用されます。

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。EN/IEC 60335-1 規格の範囲内では、フラッシュメモリの整合性を確認する手段となっています。リンク時に生成され、特定のメモリ位置に保存されたりファレンスシグネチャと比較するために、CRC 計算ユニットは、ランタイム時にソフトウェアのシグネチャの計算を支援します。

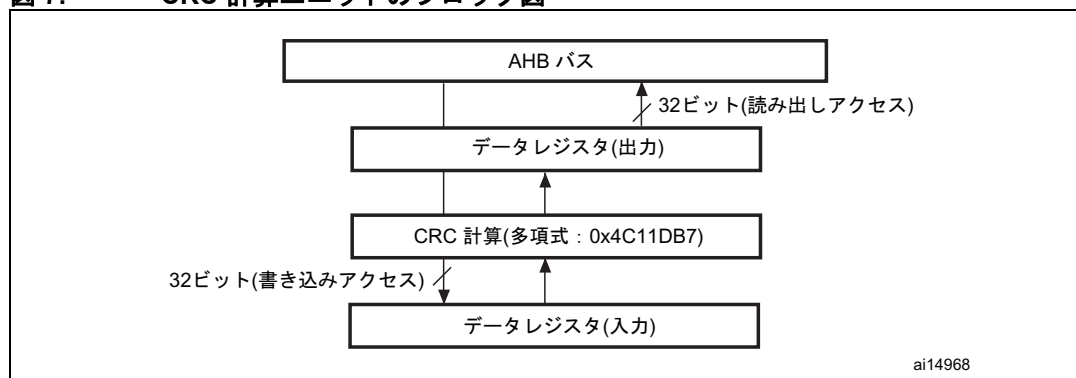
4.2 CRC の主な特徴

- CRC-32 (イーサネット) 多項式: $0x4C11DB7$ を使用

$$X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$
- シングル入力/出力 32 ビットデータレジスタ
- CRC 計算は 4 AHB クロックサイクル (HCLK) で完了
- 汎用 8 ビットレジスタ (一時ストレージとして使用可能)

図 7 にブロック図を示します。

図 7. CRC 計算ユニットのブロック図



4.3 CRC の機能説明

- CRC 計算ユニットは、主に次のような 1 つの 32 ビット・データ・レジスタから構成されています。
- CRC 計算機に新しいデータを入力する入力レジスタとして使用されます（レジスタ書き込み時）。
 - 前の CRC 計算結果を保持します（レジスタを読み出し時）。

データレジスタへの書き込み動作のたびに、前の CRC 値と新しい値の組み合わせが作成されます（CRC 計算は 32 ビットデータワード単位で行われ、バイト単位ではありません）。

書き込み動作は、CRC 計算の終了まで一時停止されるので、連続書き込みアクセスや連続読み書きアクセスが可能です。

CRC 計算機は、CRC_CR レジスタの RESET 制御ビットで、0xFFFF FFFF にリセットできます。この操作は、CRC_IDR レジスタの内容に影響しません。

4.4 CRC レジスタ

CRC 計算ユニットは、2 つのデータレジスタと制御レジスタから構成されています。CRC 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。

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]															
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
DR [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 31:0 データレジスタビット

- 新しいデータを CRC 計算機に書き込むときに入力レジスタとして使用されます。
- 読み出し時には、前の CRC 計算結果を保持します。



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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								IDR[7:0]							
								r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/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 汎用 8 ビットデータレジスタビット

1 バイトの一時記憶場所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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み														リセット	
														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 RESET ビット

CRC 計算ユニットをリセットして、データレジスタを 0xFFFF FFFF にセットします。

このビットはセットのみができ、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 レジスタマップ

次の表に、CRC レジスタマップとリセット値を示します。

表 24. CRC 計算ユニットのレジスタ・マップとリセット値

オフ セット	レジスタ	31-24	23-16	15-8	7	6	5	4	3	2	1	0
0x00	CRC_DR	データレジスタ										
	リセット値	0xFFFF FFFF										
0x04	CRC_IDR	予約済み			独立型データレジスタ							
	リセット値				0x00							
0x08	CRC_CR	予約済み										リセット
	リセット値											0



5 電源コントローラ (PWR)

5.1 電源

BOR の使用時、このデバイスは $1.8 \sim 3.6 \text{ V } V_{DD}$ の動作電源電圧を必要とします (パワーダウン時は最小 1.65 V)。BOR の非使用時、このデバイスは $1.65 \sim 3.6 \text{ V } V_{DD}$ の動作電源電圧を必要とします。

内蔵リニア電圧レギュレータを使用して、 $1.2 \sim 1.8 \text{ V}$ の内部デジタル電源を供給します。

- BOR の使用時、 $V_{DD} = 1.8 \text{ V}$ (パワーオン時) または 1.65 V (パワーダウン時) $\sim 3.6 \text{ V}$ 。BOR の非使用時、 $V_{DD} = 1.65 \text{ V} \sim 3.6 \text{ V}$ 。

V_{DD} は、I/O用と内蔵レギュレータ用の外部電源です。 V_{DD} ピンを使って外部から供給します。

- $V_{CORE} = 1.2 \sim 1.8 \text{ V}$

V_{CORE} は、デジタルペリフェラル、SRAM、フラッシュメモリ用の電源です。内蔵電圧レギュレータで生成します。 V_{DD} に応じて、3種類の V_{CORE} 範囲をソフトウェアから選択することができます (図 9 参照)。

- BOR 使用時は、 V_{SSA} 、 $V_{DDA} = 1.8 \text{ V}$ (パワーオン時) または 1.65 V (パワーダウン時) $\sim 3.6 \text{ V}$ で、BOR 非使用時は、 V_{SSA} 、 $V_{DDA} = 1.65 \sim 3.6 \text{ V}$ です。

V_{DDA} は、ADC、DAC、リセットブロック、RC オシレータ、PLL 用の外部アナログ電源です。ADC 使用時の V_{DDA} の最小電圧は 1.8 V です。

- V_{REF-} 、 V_{REF+}

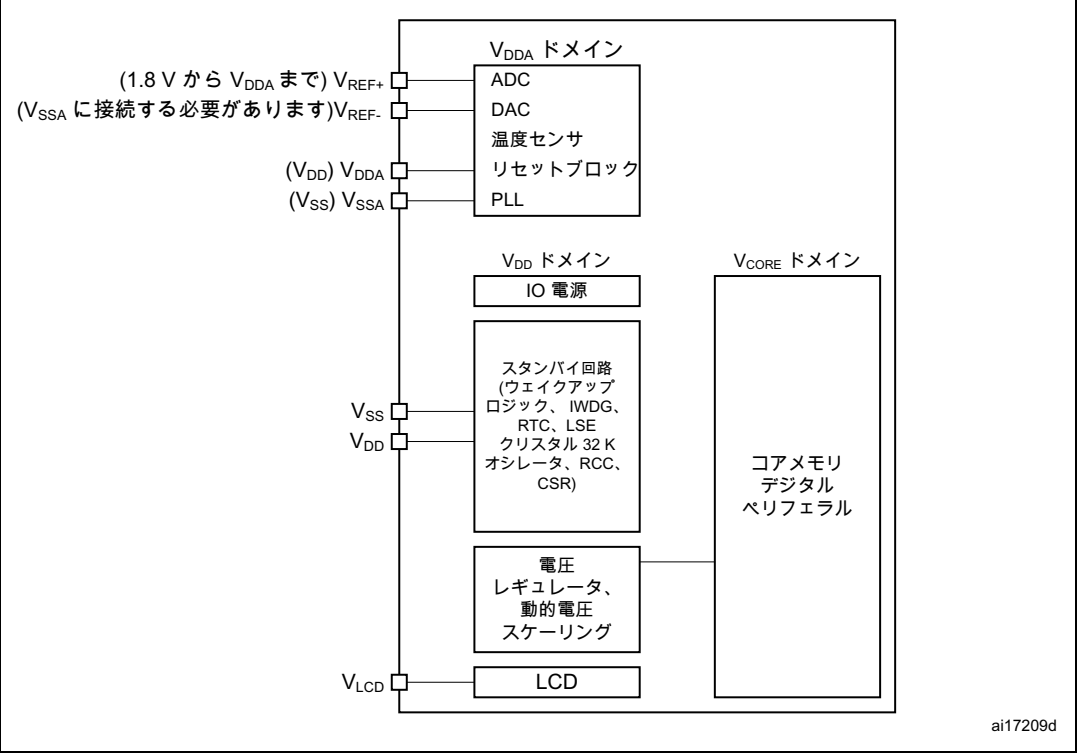
V_{REF+} は入力基準電圧です。

V_{REF-} と V_{REF+} は外部ピンとして LQFP144、UFBGA132、LQFP100、UFBGA100、TFBGA64 の各パッケージにのみ存在します。その他の場合は、それぞれ V_{SSA} と V_{DDA} に接続されます。

- $V_{LCD} = 2.5 \sim 3.6 \text{ V}$

LCD コントローラの電源は、 V_{LCD} ピンを使って外部から、または内蔵ステップアップコンバータで生成される内部電圧から供給することができます。

図 8. 電源の概要



1. V_{DDA} および V_{SSA} はそれぞれ V_{DD} および V_{SS} に接続する必要があります。
2. パッケージに応じて V_{REF-} が使用可能な場合は、このピンを V_{SSA} へ接続する必要があります。
3. 使用する動作電源範囲に応じて、一部のペリフェラルは限定された機能と性能で使用できます。詳細については、STM32L1xxxx データシートの「一般動作条件」のセクションを参照してください。

5.1.1 A/D と DAC コンバータの独立した電源と基準電圧

AD 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。この電源回路は別々にフィルタやシールドを内蔵しているため、PCB 上のノイズを防ぐことができます。

- ADC 用の電源は、別々の V_{DDA} ピンから入力されます。
- 分離された電源グラウンドは、 V_{SSA} ピンを使用しています。

BGA 64 ピンおよび 100 ピン以上の全パッケージの場合

低電圧入力および出力での精度を確保するため、 V_{DD} より低い別々の外部参照電圧を V_{REF+} に接続することができます。 V_{REF+} は回路の最大電圧であり、アナログ入力 (ADC) または出力 (DAC) 信号のフルスケール値に相当します。

- ADC の場合
 - 全速 (ADCCLK = 16 MHz、1 Msps) では、 $2.4\text{ V} \leq V_{REF+} = V_{DDA}$
 - 中速 (ADCCLK = 8 MHz、500 Ksps) では、 $1.8\text{ V} \leq V_{REF+} = V_{DDA}$
 - 中速 (ADCCLK = 8 MHz、500 Ksps) では、 $2.4\text{ V} \leq V_{REF+} \neq V_{DDA}$
 - 低速 (ADCCLK = 4 MHz、250 Ksps) では、 $1.8\text{ V} \leq V_{REF+} < V_{DDA}$
 - 製品電圧範囲 3 を選択すると ($V_{Core} = 1.2\text{ V}$)、ADC は低速 (ADCCLK = 4 MHz、250 Ksps) になります。
- DAC の場合
 - $1.8\text{ V} \leq V_{REF+} < V_{DDA}$
- V_{DDA} が 2.4 V より高い場合、 V_{REF+} の電圧は $2.4\text{ V} \sim V_{DDA}$ の範囲が可能です。
- V_{DDA} が 2.4 V より低い場合は、 V_{REF+} は V_{DDA} に一致する必要があります。

64 ピン以下のパッケージの場合 (ただし BGA パッケージを除く)

V_{REF+} ピンと V_{REF-} ピンはありません。これらは内部で ADC 電源 (V_{DDA}) とグラウンド (V_{SSA}) に接続されています。

5.1.2 独立した LCD 電源

V_{LCD} ピンは、ガラス LCD のコントラストを制御するために設けてあります。このピンは次の 2 つの方法で使用することができます。

- このピンは、ガラス LCD のセグメントラインとコラムラインに対してマイクロコントローラから供給される所要最大電圧を外部回路から入力することができます。
- また、マイクロコントローラの電圧ステップアップコンバータが使用する外部コンデンサをこのピンを使って接続することもできます。このステップアップコンバータはソフトウェアから制御されて、ガラス LCD のセグメントラインとコラムラインに対して所要電圧を供給します。

このセグメントラインとコラムラインに対する電圧は、ガラス LCD ピクセルのコントラストを決定します。フレーム間のデッドタイムを設定すると、このコントラストを小さくすることができます。

- 外部電源を V_{LCD} ピンに入力する場合、 $2.5\text{ V} \sim 3.6\text{ V}$ の範囲である必要があります。 V_{DD} には依存しません。
- LCD に内蔵ステップアップコンバータを使用する場合、 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してください)。

5.1.3 RTC レジスタと 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、定期的な割り込み機能を備えたプログラマブルウェイクアップフラグを搭載しています。RTC は、Cat.1、Cat.2 デバイスでは 20 個のバックアップデータレジスタ (80 バイト) を、Cat.3、Cat.4、Cat.5 デバイスでは 32 個のバックアップデータレジスタ (128 バイト) を、バリュールラインデバイスでは 5 個のバックアップデータレジスタ (20 バイト) を、それぞれ内蔵しています。タンパ検出イベントが発生すると、これらのバックアップレジスタはリセットされます。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 レジスタのアクセス

リセット後、RTC レジスタ (RTC レジスタと 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 レジスタアクセスの有効化は次のように行います。

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化します。
2. PWR_CR レジスタの DBP ビットをセットします ([セクション 5.4.1](#)参照)。
3. RCC_CSR レジスタの RTCSEL[1:0] ビットを使って RTC クロックソースを選択します。
4. RCC_CSR レジスタの RTCEN ビットを設定して RTC クロックを有効化します。

5.1.4 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、1.2~1.8 V (typ) の 3 種類の範囲をソフトウェアから設定することができます ([セクション 5.1.5](#)参照)。

電圧レギュレータは、リセット後は常にイネーブルです。アプリケーションモードに応じて、メイン (MR)、低消費電力 (LPR)、パワーダウンの 3 つの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) モードになり、 V_{CORE} ドメイン (CORE、メモリ、デジタルペリフェラル) にフル電力を供給します。
- 低消費電力の Run モードでは、レギュレータは低消費電力 (LPR) モードになり、 V_{CORE} ドメインに低電力を供給して、レジスタと内部 SRAM の値を保持します。
- Sleep モードでは、レギュレータはメイン (MR) モードになり、 V_{CORE} ドメインにフル電力を供給して、レジスタと内部 SRAM の値を保持します。
- 低消費電力の sleep モードでは、レギュレータは低消費電力 (LPR) モードになり、 V_{CORE} ドメインに低電力を供給して、レジスタと内部 SRAM の値を保持します。
- Stop モードでは、レギュレータは V_{CORE} ドメインに低電力を供給して、レジスタと内部 SRAM の値を保持します。
- STANDBY モードでは、レギュレータはパワーオフ状態になります。STANDBY 回路を除き、レジスタと SRAM の内容は失われます。

5.1.5 ダイナミック電圧スケーリング管理

ダイナミック電圧スケーリングは電源管理技術であり、状況に応じてデジタルペリフェラルで使用される電圧 (V_{CORE}) を増減させます。

V_{CORE} を大きくするダイナミック電圧スケーリングはオーバーボルテングと呼ばれ、デバイス性能を向上させることができます。STM32L1xxxx 動作条件対性能の説明については [図 9](#) を参照してください。

V_{CORE} を小さくするダイナミック電圧スケーリングはアンダーボルテングと呼ばれ、省電力のために、特にバッテリー駆動のラップトップやその他のモバイルデバイスで実行されます。

範囲 1

範囲 1 は「高性能」範囲です。

電圧レギュレータは、V_{DD} 入力電圧が 1.71 V~3.6 V ^(a) の範囲内にあるかぎり 1.8 V 電圧 (typ) を出力します。フラッシュの書き込み動作と消去動作はこの範囲内で実行することができます。

範囲 2 と範囲 3

V_{DD} (1.65~3.6 V) に対する制約なしで、安定化 1.5 V (typ、範囲 2) または 1.2 V (typ、範囲 3) を出力するようにレギュレータを設定することもできます。

- 1.5 V でもフラッシュメモリは機能できますが、中速の読み出しアクセス時間になります。これは「中性能」範囲です。この場合も、フラッシュメモリの書き込み動作と消去動作は可能です。
- 1.2 V でもフラッシュメモリは機能できますが、低速の読み出しアクセス時間になります。これは「低性能」範囲です。この場合は、フラッシュメモリの書き込み動作と消去動作は不可能です。

各範囲での性能については [表 25](#) を参照してください。

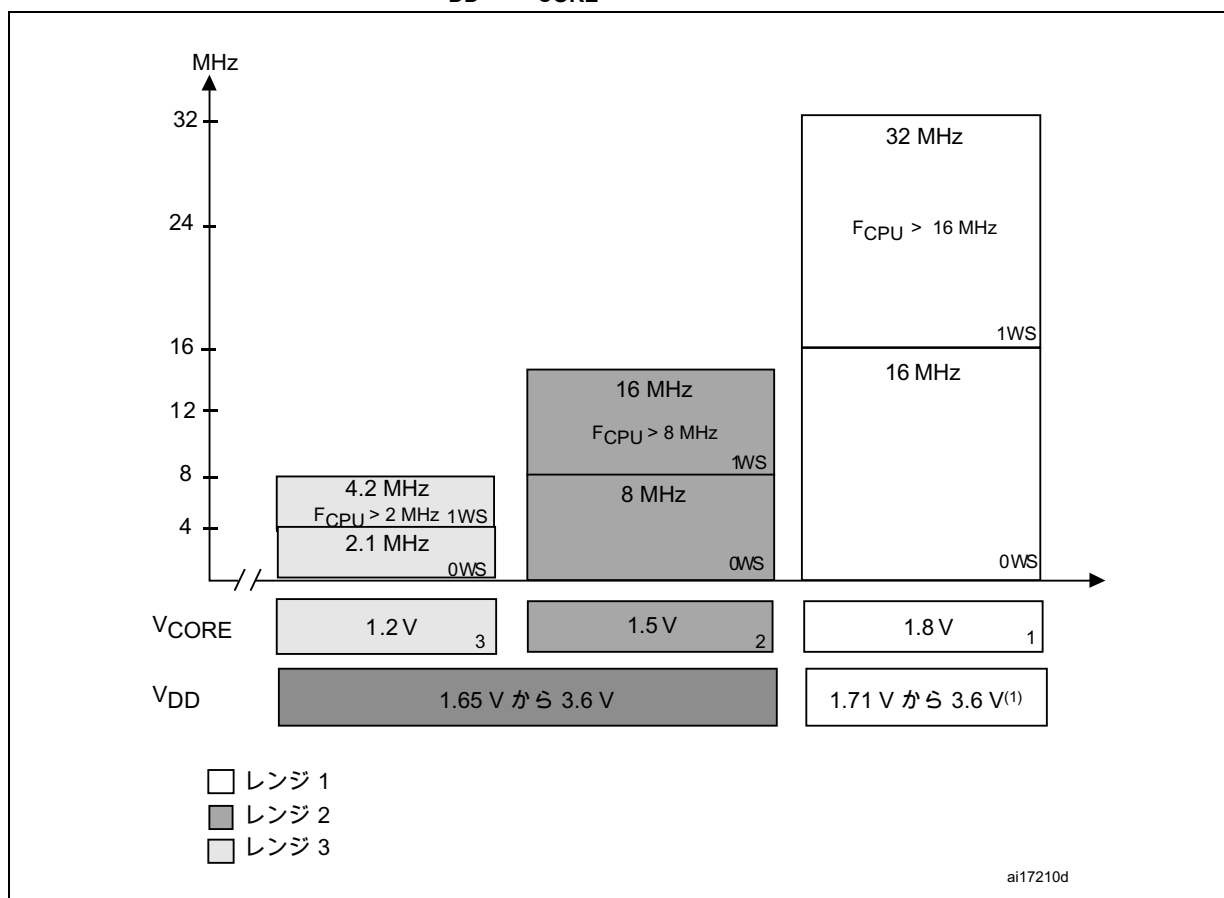
表 25. 性能対 V_{CORE} 範囲

CPU 性能	消費電力性能	V _{CORE} 範囲	Typ 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 ⁽¹⁾
中優先	中優先	2	1.5	16	8	1.65~3.6
低	高	3	1.2	4.2	2.1	

1. デバイスに応じて、V_{DD} 範囲 1 は 1.71~3.6 V または 2.0 V~3.6 V が可能です。個別の製品データシートを参照してください。

a. デバイスに応じて、V_{DD} 範囲 1 は 1.71~3.6 V または 2.0 V~3.6 V が可能です。特定の製品データシートを参照してください。



図 9. STM32L1xxxx 性能対 V_{DD} と V_{CORE} の範囲

1. デバイスに応じて、 V_{DD} 範囲 1 は 1.71~3.6 V または 2.0 V~3.6 V が可能です。個別の製品データシートを参照してください。

5.1.6 ダイナミック電圧スケーリング設定

電圧レギュレータ範囲を設定するときは次の手順に従う必要があります。

1. V_{DD} をチェックして、許容範囲を調べます (図 9 : STM32L1xxxx 性能対 V_{DD} と V_{CORE} の範囲参照)。
2. PWR_CSR の VOSF ビットをポーリングします。このビットが 0 にリセットされるまで待ちます。
3. PWR_CR レジスタの VOS[12:11] ビットをセットして、電圧スケーリング範囲を設定します。
4. PWR_CSR レジスタの VOSF ビットをポーリングします。このビットが 0 にリセットされるまで待ちます。

注 : 電圧スケーリング設定時、レギュレータが安定するまで ($VOSF=0$)、システムクロックが停止します。割り込みに対する応答時間が重要な場合、または使用するペリフェラル (タイマ、通信など) に応じて、アプリケーション開発時に、これを考慮する必要があります。

5.1.7 2.0 V 以下に V_{DD} 低下した場合の電圧レギュレータとクロックの管理

V_{CORE} 範囲 1 が選択され、かつ V_{DD} が 2.0 V 以下に低下した場合、アプリケーションからシステムを再設定する必要があります。

システムの再設定には次の 3 ステップの手順が必要です。

1. V_{DD} が 2.0 V を下回ったことを検出します。
PVD を使って V_{DD} 電圧をモニタし、電圧が選択したレベルを下回ったときに割り込みを発生させます。2.0 V の電圧制限値を検出するときは、ソフトウェア PVD 閾値 2 (2.26 V typ) を使って、アプリケーションから選択することができます。PVD の詳細については、[セクション 5.2.3](#) 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範囲に合わせます。
2.0 V より下では、システムクロック周波数は、範囲 2 では 16 MHz に、範囲 3 では 4.2 MHz に、それぞれ制限されます。
3. 所望の電圧範囲を選択します。
 V_{DD} が 1.71 V より低い場合は、範囲 2 または範囲 3 のみ選択可能であることに注意してください。

注： V_{CORE} 範囲 2 または範囲 3 が選択され、かつ V_{DD} が 2.0 V より低い場合は、システム再設定は不要です。

5.1.8 V_{CORE} 範囲を変更する際の電圧レギュレータとクロックの管理

V_{DD} が 2.0 V より高い場合は、3 つの電圧範囲のいずれも選択できます。

- 電圧範囲が対象電圧範囲を上回る場合 (例えば、範囲 1 から範囲 2 へ)
 - a) クロック周波数を次のステップで選択する低い電圧範囲に合わせます。
 - b) 所望の電圧範囲を選択します。
- 電圧範囲が対象電圧範囲を下回る場合 (例えば、範囲 3 から範囲 1 へ)
 - a) 所望の電圧範囲を選択します。
 - b) 必要に応じてクロック周波数を調整します。

V_{DD} が 2.0 V より低い場合は、範囲 2 と範囲 3 のみが選択可能です。

- 範囲 2 から範囲 3 へ
 - a) クロック周波数を電圧範囲 3 に合わせます。
 - b) 電圧範囲 3 を選択します。
- 範囲 3 から範囲 2 へ
 - a) 電圧範囲 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。

5.2 電源供給スーパバイザ

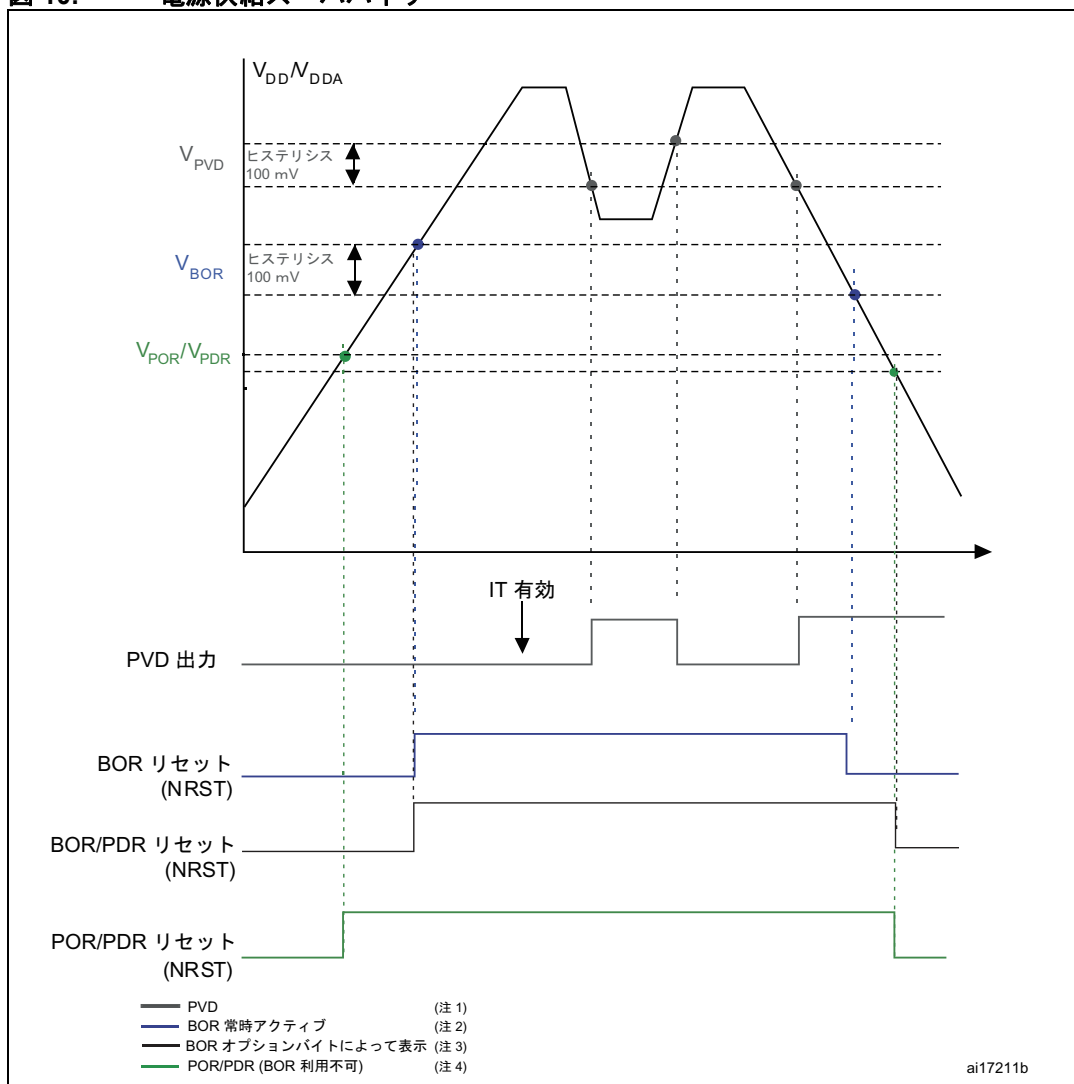
デバイスは、ゼロパワーパワーオンリセット (POR) / パワーダウリセット (PDR)、さらにブラウンアウトリセット (BOR) 回路を内蔵しています。1.8~3.6 V で動作するデバイスでは、パワーオン時に BOR が常にアクティブであるため、1.8 V から正常動作が保証されます。1.8 V の BOR 閾値に到達すると、オプションバイトのローディング処理が開始され、デフォルト閾値の使用または変更が行われるか、または BOR が永久に無効化されます (この場合、パワーダウン時の V_{DD} 最小値は 1.65 V になります)。1.65 V~3.6 V で動作するデバイスでは、BOR は固定的に無効化されています。したがって、パワーオン時のスタートアップ時間は最小 1 ms typ まで検出することができます。

1.65~3 V の 5 つの BOR 閾値をオプションバイトを使って設定することができます。Stop モードで消費電力を削減するために、内部電圧基準 V_{REFINT} を自動的に切り離すことができます。 V_{DD} が規定の閾値 V_{POR} 、 V_{PDR} または V_{BOR} を下回る場合、外部リセット回路なしで、デバイスはリセットモードを維持します。

このデバイスは、 V_{DD}/V_{DDA} 電源をモニタし、 V_{PVD} 閾値と比較するプログラマブルな電圧検出器 (PVD) を内蔵しています。200 mV ステップの 1.85~3.05 V の 7 種類の PVD レベルをソフトウェアからを選択することができます。 V_{DD}/V_{DDA} が V_{PVD} 閾値を下回る場合および/または V_{DD}/V_{DDA} が V_{PVD} 閾値を上回る場合、割り込みを発生させることができます。その後、割り込みサービスルーチンが警告メッセージを生成するか、または MCU を安全な状態にします。PVD はソフトウェアから有効化されます。

様々な電源供給スーパバイザ (POR、PDR、BOR、PVD) を [図 10](#) に示します。

図 10. 電源供給スーパバイザ



1. PVD はすべての STM32L デバイスに存在し、ソフトウェアから有効化/無効化されます。
2. BOR は 1.8~3.6 V で動作するデバイスにのみあり、オプションバイトで無効化されないかぎり、POR/PDR 閾値をマスクします。
3. オプションバイトを使って BOR を無効化した場合、 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. 1.65~3.6 V で動作するデバイスでは、BOR がいないため、 V_{DD} が POR レベルを上回るとリセットが解除され、 V_{DD} が PDR レベルを下回ると、アサートされます。

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

このデバイスは、最小 1.5 V までの動作を可能にする POR/PDR 回路を内蔵しています。

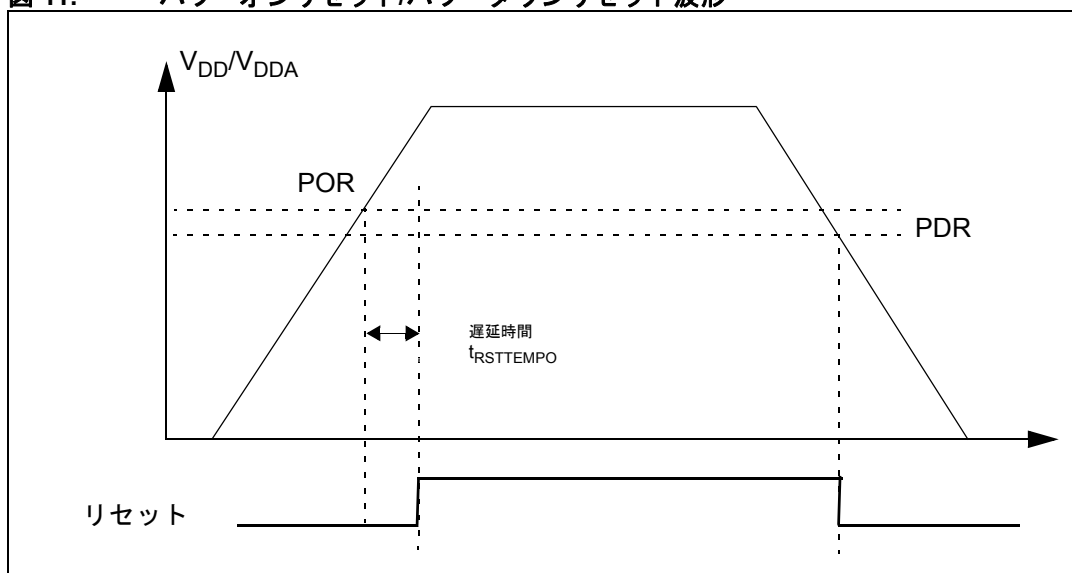
V_{DD}/V_{DDA} が規定の閾値 V_{POR} を下回っている場合、このデバイスは、外部のリセット回路を必要とせずリセットモードに留まることができます。POR 機能は常に有効化されており、POR 閾値は 1.5 V です。

パワーダウン時、電源電圧 (V_{DD}) が V_{PDR} 閾値を下回ったとき、PDR はデバイスのリセットを維持します。PDR 機能は常に有効化されており、PDR 閾値は 1.5 V です。

POR と PDR は、BOR が無効化された場合にのみ使用されます ([セクション 5.2.2: ブラウンアウトリセット参照](#))。最小動作電圧 (1.65 V) を保証するため、BOR を BOR レベル 0 に設定する必要があります。BOR が無効化された場合、最小動作電圧 (1.65 V) と V_{POR}/V_{PDR} 閾値との間に「グレーゾーン」が存在します。これは、 V_{PDR} 閾値に到達するまでデバイスリセットなしで V_{DD} が 1.65 V を下回ることができることを意味します。

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

図 11. パワーオンリセット/パワーダウンリセット波形



5.2.2 ブラウンアウトリセット

パワーオン時、電源電圧が規定の閾値 V_{BOR} 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維持されます。

1.65~3.6 V で動作するデバイスでは、BOR オプションが存在しないため、電源は POR/PDR によりモニタされます。POR/PDR 閾値が 1.5 V の場合と同様に、 V_{POR}/V_{PDR} 閾値と最小製品動作電圧 1.65 V との間に「グレーゾーン」が存在します。

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

次に、システムリセットが解除されたとき、BOR レベルを再設定するか、またはオプションバイトのローディングにより無効化することができます。

BOR レベルを最小レベルに維持する場合 (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 により完全に管理され、製品動作電圧は安全な範囲になります。

さらに、オプションバイトにより BOR オプションが無効化された場合、パワーダウンリセットは PDR から制御され、1.65 V と V_{PDR} との間に「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。デフォルトでは、レベル 4 閾値が有効になっています。5種類の V_{BOR} 閾値を選択することができます。

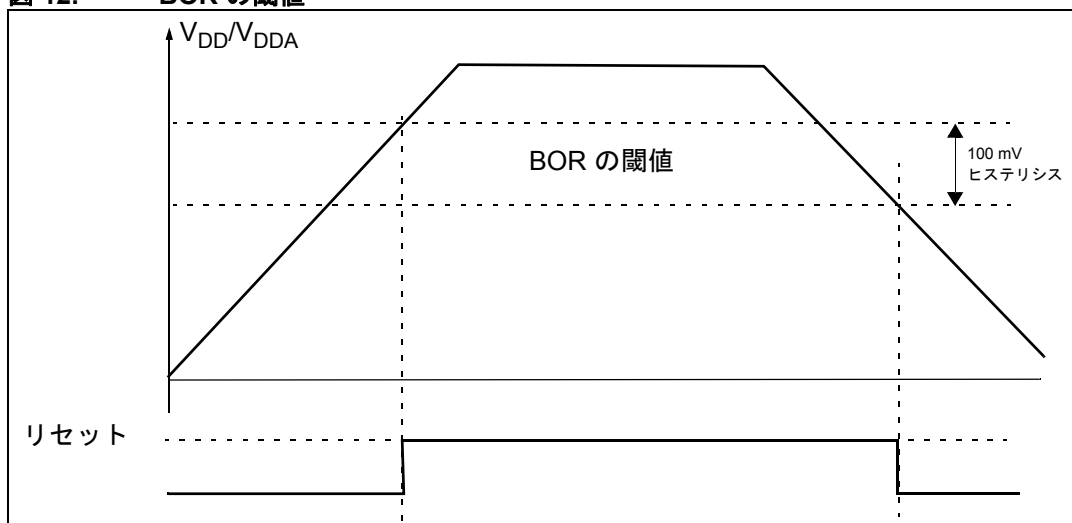
- BOR レベル 0 (V_{BOR0}) : 1.69~1.80 V の電圧範囲に対するリセット閾値レベル
- BOR レベル 1 (V_{BOR1}) : 1.94~2.1 V の電圧範囲に対するリセット閾値レベル
- BOR レベル 2 (V_{BOR2}) : 2.3~2.49 V の電圧範囲に対するリセット閾値レベル
- BOR レベル 3 (V_{BOR3}) : 2.54~2.74 V の電圧範囲に対するリセット閾値レベル
- BOR レベル 4 (V_{BOR4}) : 2.77~3.0 V の電圧範囲に対するリセット閾値レベル

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} 上限を上回ると、デバイスリセットが解除されて、システムが動作を開始できます。

デバイスオプションバイトを設定して、BOR を無効化することができます。BOR 機能を無効化するときは、 V_{DD} を V_{BOR0} より高くして、デバイスオプションバイトの設定シーケンスを開始させる必要があります。次に、POR と PDR を使ってパワーオンとパワーダウンをモニタします ([セクション 5.2.1 : パワーオンリセット \(POR\) / パワーダウンリセット \(PDR\)](#) 参照)。

BOR 閾値のヒステリシスは約 100 mV です (電源電圧の立上りエッジと立下りエッジの間)。

図 12. BOR の閾値



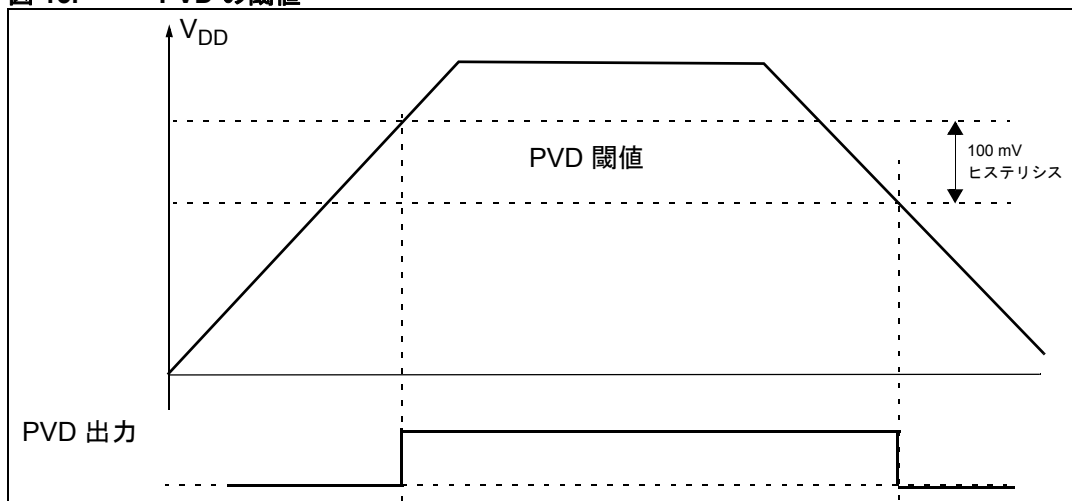
5.2.3 プログラマブル電圧検出器 (PVD)

PWR_CR の PLS[2:0] ビットによって選択された閾値と比較することによって、PVD を使用して V_{DD} 電源を監視することができます (セクション 5.4.1 参照)。

PVD は外部入力アナログ電圧 (PVD_IN) を使用することができます。このピンは内部で VREFINT に接続されています。PVD_IN (PB7) は、PLS[2:0] = 111 のとき、アナログモードに設定する必要があります。PVDE ビットをセットすると、PVD が有効化されます。

V_{DD} が PVD 閾値より高いか低いかに示す目的で、PWR_CSR (セクション 5.4.2 参照) の PVDO フラグが使用できます。このイベントは EXTI のライン 16 に内部で接続されており、EXTI レジスタを介して許可状態となっていれば、割り込みを発生させることができます。EXTI ライン 16 の立ち上り/立ち下りエッジの設定に応じて、 V_{DD} が PVD 閾値を下回ったとき、または上回ったとき、あるいはその両方で、PVD 出力割り込みを発生させることができます。たとえば、この割り込みサービスルーチンで、緊急停止処理などを実行することが可能です。

図 13. PVD の閾値



5.2.4 内部電圧基準 (V_{REFINT})

内部基準 (V_{REFINT}) は、アナログペリフェラルに対して安定な電圧を提供します。内部電圧基準 (V_{REFINT}) を使って管理される機能は、BOR、PVD、ADC、LCD、コンパレータです。内部電圧基準 (V_{REFINT}) は、BOR、PVD、ADC、LCD またはコンパレータがこれを使用する場合にアクティブになります。

内部電圧基準の消費電力は、特に Stop モードと Standby モードでは無視できません。消費電力を小さくするため、PWR_CR レジスタの ULP ビット (超低消費電力) をセットして、内部電圧基準を無効化することができますが、この場合、Stop/Standby モードから抜け出すとき、内部電圧基準を使って管理される機能は、内部電圧基準スタートアップ時間の間 (最大 3 ms) 不安定になります。

ウェイクアップ時間を短くするため、デバイスは内部電圧基準スタートアップ時間を待たないで Stop/Standby モードから抜け出すことができます。これは、PWR_CR レジスタの FWU ビット (高速ウェイクアップ) をセットした後に Stop/Standby モードに入ることにより実行されます。

ULP ビットがセットされると、Stop/Standby モードに入る前に有効化された機能はこれらのモードの間無効化され、FWU 値によらず内部電圧基準スタートアップ時間の経過後にのみ再度有効化できます。PWR_CSR レジスタの VREFINTRDYF フラグは、内部電圧基準がレディであることを示します。

5.3 低消費電力モード

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電源リセット後は 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、いくつかの低消費電力モードが用意されています。消費電力の節減、スタートアップ時間の短縮、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 つの低消費電力モードを備えています。

- 低消費電力 run モード: レギュレータは低消費電力モード、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®-M3CORE は停止、ペリフェラルは動作
- 低消費電力 sleep モード: Cortex®-M3CORE 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、レギュレータは低消費電力モード、RAM はパワーダウン、フラッシュは停止
- Stop モード (全クロックが停止、レギュレータは動作、レギュレータは低消費電力モード)
- Standby モード: V_{CORE} ドメインの電源がオフ

さらに、次のような手段で 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
- 使用しない場合の APBx および AHBx ペリフェラルへのクロックの停止

表 26. 低消費電力モードの一覧

モード名	エントリ	ウェイクアップ	V _{CORE} ドメイン 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
低消費電力動作	LPSDSR ビットと LPRUN ビット + クロック設定	レギュレータがメインレギュ レータ (1.8 V) になります	なし	なし	低消費電力モード
スリープ (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	ON
	WFE	ウェイクアップイベント			
低消費電力ス リープ (Sleep now または Sleep-on-exit)	LPSDSR ビット + WFI	割り込み	CPU クロックオフ その他のクロック またはアナログ クロックソースに 影響なし フラッシュ CLK は オフ	なし	低消費電力モード
	LPSDSR ビット + WFE	ウェイクアップイベント			
STOP	PDDS、LPSDSR 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)	全 V _{CORE} ドメイン クロックがオフ	HSI、 HSE、MSI オシレータ がオフ	ON、低消費電力 モード (PWR_CR に依存)
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

5.3.1 低消費電力モードでのクロック動作

APB ペリフェラルと DMA のクロックは、ソフトウェアから無効化することができます。

Sleep モードと低消費電力 sleep モード

Sleep モードと低消費電力 sleep モードでは CPU クロックが停止します。メモリインタフェースクロック (FLITF インタフェースと RAM インタフェース) と全ペリフェラルのクロックは、Sleep 時にソフトウェアから停止することができます。低消費電力 sleep モードでは、メモリインタフェース (FLITF) クロックが停止し、RAM がパワーダウします。AHB から APB へのブリッジクロックは、接続されているペリフェラルのすべてのクロックが無効化されたとき、Sleep/低消費電力 sleep モード時にハードウェアにより無効化されます。

Stop モードと Standby モード

システムクロックとすべての高速クロックは、Stop モードと Standby モードで停止します。

- PLL は無効化されます。
- 内部 RC 16 MHz (HSI) 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 (Stop モード) またはリセット (Standby モード) によりこのモードから抜け出すとき、内部 MSI オシレータがシステムクロックとして選択されます。デバイスが Stop モードを抜け出すとき、前の MSI 設定 (範囲とトリミング値) は維持されます。Standby モードを抜け出すとき、範囲とトリミング値は 2 MHz デフォルト値にリセットされます。

フラッシュ書き込み動作中または APB ドメインへのアクセス中の場合、Stop/Standby モードの開始は、フラッシュメモリまたは APB のアクセスが完了するまで遅らされます。

5.3.2 システムクロックの低下

RUN モードでは、プリスケアラレジスタをプログラムすることによってシステムクロック (SYSCLK、HCLK、PCLK1、PCLK2) の速度を下げるすることができます。これらのプリスケアラを使用して、スリープモードに入る前にペリフェラルの速度を低下させておくこともできます。

詳細については、[セクション 6.3.3: クロック設定レジスタ \(RCC_CFGR\)](#) を参照してください。

5.3.3 ペリフェラルクロックゲーティング

RUN モードでは、個々のペリフェラルとメモリの HCLK および PCLKx をいつでも停止して、消費電力を節減できます。

SLEEP モードで消費電力をさらに節減するには、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しておきます。

ペリフェラルのクロックは AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBENR)、APB2 ペリフェラルクロックイネーブルレジスタ (RCC_APB2ENR) および APB1 ペリフェラルクロックイネーブルレジスタ (RCC_APB1ENR) で制御されます ([セクション 6.3.8: AHB ペリフェラルクロックイネーブルレジスタ \(RCC_AHBENR\)](#)、[セクション 6.3.10: APB1 ペリフェラルクロックイネーブルレジスタ \(RCC_APB1ENR\)](#)、[セクション 6.3.9: APB2 ペリフェラルクロックイネーブルレジスタ \(RCC_APB2ENR\)](#) 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。



5.3.4 低消費電力 run モード (LP run)

システムの Run モード中の消費電力をさらに削減するときは、レギュレータを低消費電力モードに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} 範囲 1 を超えることはできません。

電圧レギュレータとペリフェラルの動作条件については、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 ($7 \times RTCLK$) より小さい場合に RTC カレンダーレジスタを読み出せるようにするときは、ソフトウェアからカレンダー時間と日付レジスタを 2 回読み出す必要があります。

RTC_TR の 2 回目の読み出しで初回の読み出しと一致した場合、そのデータが正しいことが保証されます。その他の場合、3 回目の読み出しアクセスを行う必要があります。

低消費電力 run モードは、 V_{CORE} が範囲 2 のときにのみ開始することができます。さらに、低消費電力 run モードを選択した場合、ダイナミック電圧スケーリングを使用する必要があります。低消費電力 run モードを選択した場合、レギュレータを低消費電力モードに設定した Stop モードと Sleep モードのみが可能です。

注： 低消費電力 run モードでは、すべての I/O ピンが Run モードと同じ状態を維持します。

低消費電力 run モードの開始

低消費電力 run モードに入るときは、次の手順に従います。

- RCC_APBxENR レジスタと RCC_AHBENR レジスタを使って、各デジタル IP クロックを有効化するか、無効化する必要があります。
- システムクロック周波数は、 f_{MSI} 範囲 1 の周波数を超えないように低く抑える必要があります。
- レギュレータは、ソフトウェアから低消費電力モードに設定されます (LPRUN ビットと LPSDSR ビットをセット)。

低消費電力 run モードの終了

低消費電力 run モードを終了するときは、次の手順に従います。

- ソフトウェアからレギュレータをメインレギュレータモードに設定します。
- 必要に応じて、フラッシュメモリを切り離します。
- クロックシステム周波数を上げることができます。

5.3.5 SLEEP モード

Sleep モードの開始

WFI (Wait For Interrupt) 命令または WFE (イベント待ち) 命令を実行することにより、SLEEP モードに入ります。SLEEP モード遷移メカニズムの選択には、Cortex[®]-M3 システム制御レジスタの SLEEPONEXIT ビットに応じて 2 つのオプションが使用できます。

- 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 モードに入ります。
- 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 を終了した時点でスリープモードに入ります。

注： Sleep 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ちます。

SLEEP モードへの入り方の詳細については、表 27 : Sleep-now および 表 28 : Sleep-on-exit を参照してください。

SLEEP モードの終了

WFI 命令によって SLEEP モードに入った場合、ペリフェラルの割り込みがネスト型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 モードに入った場合、イベントが発生すると同時に MCU は 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®-M3 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 から処理を再開する場合、(NVIC 割り込みクリアペンディングレジスタの) ペリフェラル割り込みペンディングビットとペリフェラル 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。
- または、外部 EXTI ラインまたは内部 EXTI ラインをイベントモードに設定します。CPU が WFE から再開するときには、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 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
モードの開始	次の条件下での WFI (割り込み待ち) または WFE (イベント待ち) – SLEEPDEEP = 0 および – SLEEPONEXIT = 0 Cortex®-M3 システム制御レジスタの説明を参照
モードの終了	WFI 命令を使用して開始した場合 : 割り込み : を参照。 表 48 : ベクタテーブル (Cat.1 およびCat.2 デバイス) WFE 命令で SLEEP モードに入った場合、 ウェイクアップイベント : を参照してください。 セクション 10.2.3 : ウェイクアップイベント管理
ウェイクアップ待ち時間	なし

表 28. Sleep-on-exit

Sleep-on-exit	説明
モードの開始	次の条件下での WFI (割り込み待ち) – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 Cortex®-M3 システム制御レジスタの説明を参照してください。
モードの終了	割り込み 表 48 : ベクタテーブル (Cat.1 およびCat.2 デバイス) 、 表 49 : ベクタテーブル (Cat.3 デバイス) 、 表 50 : ベクタテーブル (Cat.4 およびCat.5 デバイス) を参照してください。
ウェイクアップ待ち時間	なし

5.3.6 低消費電力 sleep モード (LP スリープ)

低消費電力 sleep モードの開始

電圧レギュレータを低消費電力モードに設定して、WFI (割り込み待ち) または WFE (イベント待ち) 命令を実行すると、低消費電力 sleep モードが開始されます。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 を超えることはできません。

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

低消費電力 sleep モードは、 V_{CORE} が範囲 2 のときにのみ開始することができます。

注 : **APB1 クロック周波数が RTC クロック周波数の 7 倍 ($7 \times \text{RTCCLK}$) より小さい場合に RTC カレンダーレジスタを読み出せるようにするときは、ソフトウェアからカレンダー時間と日付レジスタを 2 回読み出す必要があります。**

RTC_TR の 2 回目の読み出しで初回の読み出しと一致した場合、そのデータが正しいことが保証されます。その他の場合、3 回目の読み出しアクセスを行う必要があります。

SLEEP 低消費電力モード遷移メカニズムの選択には、Cortex[®]-M3 システム制御レジスタの SLEEPONEXIT ビットに応じて 2 つのオプションが使用できます。

- 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 モードに入ります。
- 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 を終了した時点でスリープモードに入ります。

低消費電力 sleep モードに入るときは、次の手順に従います。

- フラッシュメモリは、制御ビット (FLASH_ACR レジスタの SLEEP_PD) を使って切り離すことができます (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 \(フラッシュ\)](#) を参照してください)。これにより消費電力が削減されますが、ウェイクアップ時間が長くなります。
- RCC_APBxENR レジスタと RCC_AHBENR レジスタを使って、各デジタル IP クロックを有効化するか、無効化する必要があります。低消費電力 sleep モードで動作を続けるためには、RCC_APBxLPENR と RCC_AHBLPENR を使ってデジタル IP クロックを有効化する必要があります。
- システムクロック周波数は下げる必要があります。
- レギュレータは、ソフトウェアから低消費電力モードに設定されます (LPSSDR ビットをセット)。
- Sleep モードを開始するときは、WFI/WFE 命令を実行する必要があります。

注 : **低消費電力 sleep モードでは、すべての I/O ピンが Run モードと同じ状態を維持します。**

低消費電力 sleep モードの開始方法については、[表 29: Sleep-now](#) と [表 30: Sleep-on-exit](#) を参照してください。

低消費電力 sleep モードの終了

WFI 命令によって低消費電力 SLEEP モードに入った場合、ペリフェラルの割り込みがネスト型ベクタ割り込みコントローラ (NVIC) によって認識されると、デバイスは低消費電力 SLEEP モードを終了することができます。

WFE 命令によって低消費電力 SLEEP モードに入った場合、MCU はイベントの発生直後に低消費電力 SLEEP モードを終了します。ウェイクアップイベントを次のようにして発生させることができます。

- 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®-M3 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 から処理を再開する場合、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割り込みペンディングビットとペリフェラル 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。
- または、外部 EXTI ラインまたは内部 EXTI ラインをイベントモードに設定します。CPU が WFE から再開するときには、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り低消費電力 sleep モードを終了させる場合、レギュレータがメインレギュレータモードに設定され、必要に応じてフラッシュメモリが接続され、システムクロックを高くすることができます。

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

Sleep 低消費電力モードの終了方法については、[表 29: Sleep-now](#)と [表 30: Sleep-on-exit](#)を参照してください。

表 29. Sleep-now

Sleep-now モード	説明
モードの開始	電圧レギュレータは低消費電力モードで、フラッシュメモリは切り離されます。 次の場合 WFI（割り込み待ち）または WFE（イベント待ち） – SLEEPDEEP = 0 および – SLEEPONEXIT = 0 Cortex®-M3 システム制御レジスタの説明を参照してください。
モードの終了	電圧レギュレータは低消費電力モードで、フラッシュメモリは接続されます。 WFI 命令を使用して開始した場合 割り込み：を参照。 表 48: ベクタテーブル (Cat.1 および Cat.2 デバイス) WFE 命令を使用して開始した場合 ウェイクアップイベント：を参照してください。 セクション 10.2.3: ウェイクアップイベント管理
ウェイクアップ待ち時間	低消費電力モードからのレギュレータウェイクアップ時間

表 30. Sleep-on-exit

Sleep-on-exit	説明
モードの開始	電圧レギュレータは低消費電力モードで、フラッシュメモリは切り離されます。 次の間WFI（割り込み待ち） – SLEEPDEEP = 0 および – SLEEPONEXIT = 1 Cortex®-M3 システム制御レジスタの説明を参照してください。
モードの終了	割り込み 表 48: ベクタテーブル (Cat.1 および Cat.2 デバイス) 、 表 49: ベクタテーブル (Cat.3 デバイス) 、 表 50: ベクタテーブル (Cat.4 および Cat.5 デバイス) を参照してください。
ウェイクアップ待ち時間	低消費電力モードからのレギュレータウェイクアップ時間

5.3.7 Stop モード

STOP モードは、Cortex®-M3 のディープスリープ(deepsleep) モードとペリフェラルクロックスイッチ機能を組み合わせたものです。電圧レギュレータは、通常モードまたは低消費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、HSI、および HSE RC オシレータは無効になります。内蔵 SRAM とレジスタの内容は保持されます。

最小消費電力の Stop モードでは、内部フラッシュメモリも低消費電力モードになります。フラッシュメモリがパワーダウンモードにある場合、STOP モードからのウェイクアップの際に余分なスタートアップ時間が必要になります。

Stop モードで消費電力を小さくするためには、V_{REFINT}、BOR、PVD、温度センサを切り離した後に、Stop モードを開始します。Stop モードが終了した後、PWR_CR レジスタの ULP ビットを使ってソフトウェアからこれらを再度接続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ちます。

Stop モードの開始

STOP モードへの入り方については、表 31 を参照してください。

注： STOP モードに入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 HSI に切り替える必要があります。

そうしないと、STOP モードに入る前に外部クロック (外部オシレータ) が切り離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 モードに入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) 機能が作動してしまいます。

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消費電力モードにします。これは、PWR_CR レジスタの LPSSDR ビットで設定されます (セクション 5.4.1 参照)。

フラッシュ書き込み動作中または APB ドメインへのアクセス中の場合、Stop モードの開始は、メモリまたは APB のアクセスが完了するまで遅らされます。

STOP モードでは、個別の制御ビットをプログラムすることによって、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 21 : 独立型ウォッチドッグ (IWDG) のセクション 21.3 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 6.3.14 参照)。
- 内部 RC オシレータ (LSI RC) : これは、RCC_CSR レジスタの LSION ビットによって設定されます。
- 外部 32.768 kHz オシレータ (LSE OSC) : これは、RCC_CSR レジスタの LSEON ビットによって設定されます。

ADC、DAC、LCD も、STOP モードに入る前に停止させない限り、STOP モードで電力を消費します。これらを停止させるときは、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 31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、システムクロックとして MSI RC オシレータが選択されます。

電圧レギュレータが低消費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、更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 31. Stop モード

STOP モード	説明
モードの開始	次の条件下での WFI（割り込み待ち）または WFE（イベント待ち） – Cortex®-M3 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 – PWR_CR の LPSSDR ビットの設定による電圧レギュレータのモードを選択 注： STOP モードに入るには、すべての EXTI ラインペンディングビット (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、STOP モード開始手続きが無視され、プログラムの実行が継続します。
モードの終了	WFI 命令を使用して開始した場合： 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 48：ベクタテーブル (Cat.1 およびCat.2 デバイス)、表 49：ベクタテーブル (Cat.3 デバイス)、表 50：ベクタテーブル (Cat.4およびCat.5 デバイス) を参照してください。 WFE 命令を使用して開始した場合： イベントモードに設定されている任意の EXTI ライン。を参照してください。セクション 10.2.3：ウェイクアップイベント管理 (245 ページ)
ウェイクアップ待ち時間	MSI RC ウェイクアップ時間 + レギュレータの低消費電力モードからのウェイクアップ時間 + FLASH ウェイクアップ時間

5.3.8 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®-M3 のディープスリープ (deepsleep) モードに基づきます。したがって、V_{CORE} ドメインはパワーオフします。PLL、MSI、HSI オシレータ、および HSE オシレータもオフになります。SRAM とレジスタの内容は、RTC レジスタ、RTC バックアップレジスタ、STANDBY 回路を除いて失われます (図 8を参照してください)。

Standby モードの開始

STANDBY モードへの入り方の詳細については、表 32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ムすることによって、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 21.3 : IWDG の機能詳細 (567 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 6.3.14参照)。
- 内部 RC オシレータ (LSI RC) : これは、RCC_CSR レジスタの LSION ビットによって設定されます。
- 外部 32.768 kHz オシレータ (LSE OSC) : これは、RCC_CSR レジスタの LSEON ビットによって設定されます。

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 (WUKP1、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 からウェイクアップ後に、PWR 電源制御/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 (ブートピン信号のサンプリング、リセットベクタのフェッチなど) と同様に再開します。PWR_CSR レジスタ (セクション 5.4.2参照) の SBF ステータスフラグは、MCU が STANDBY モードに在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 32を参照してください。

表 32. Standby モード

STANDBY モード	説明
モードの開始	次の条件下での WFI (割り込み待ち) または WFE (イベント待ち) – Cortex®-M3 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 またはアラーム 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
ウェイクアップ待ち時間	リセットフェーズ



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。

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 (STANDBY モードでも有効)
- ウェイクアップピン 2 (WKUP2) に設定された場合の RTC_AF1 ピン (PC13)、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。
- 有効化された場合の WKUP ピン 1 (PA0) と WKUP ピン 3 (PE6)。

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、デバッグ接続が切れてしまいます。これは Cortex[®]-M3 コアにクロックが供給されなくなることによるものです。

ただし、DBGMCU_CR レジスタのいくつかの設定ビットをセットすることによって、低消費電力モードを広範囲にわたって使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 30.16.1：低電力モードのデバッグサポート](#)を参照してください。

5.3.9 RTC とコンパレータを使用した、Stop モードおよび Standby モードからの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) に無関係に、RTC アラームイベント、RTC ウェイクアップイベント、タンパイベント、タイムスタンプイベント、またはコンパレータイイベントにより低消費電力モードからウェイクアップすることができます。

これらの RTC 代替機能は、Stop および Standby 低消費電力モードからシステムをウェイクアップさせることができますが、コンパレータイイベントは Stop モードのみからシステムを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 (オート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消費電力モードからウェイクアップすることもできます。

RTC は、一定の時間間隔で STOP または STANDBY モードからウェイクアップするためのプログラム可能なタイムベースとなります。この目的のため、RCC_CSR レジスタの RTCSEL[1:0] ビットを設定することによって、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 6.3.14](#)参照)。

- 低消費電力 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 (標準的な条件で追加消費電力 1 μ A 未満) で高精度のタイムベースを提供します。
- 低消費電力内部 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。内部 RC オシレータは、消費電力が最小とな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 アラームを発生させ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 レジスタの RTC ウェイクアップ割り込みを有効にします。
 - c) RTC が RTC ウェイクアップイベントを発生させるように設定します。

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 アラームを発生させ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 レジスタの RTC ウェイクアップ割り込みを有効にします。
 - b) RTC が RTC ウェイクアップイベントを発生させるように設定します。

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 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) を選択したエッジ (立ち下がり、立ち上がりまたは立ち下がり立ち上がり) を検出するように設定します。
 - b) イベントが発生するようにコンパレータを設定します。

5.4 電源制御レジスタ

ペリフェラルレジスタは、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。

5.4.1 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るリセット)

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	LPRUN	予約済み	VOS[1:0]		FWU	ULP	DBP	PLS[2:0]			PVDE	CSBF	CWUF	PDDS	LPSDSR
	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、読み出すと常に 0 が返されます。

ビット 14 **LPRUN**: 低消費電力 run モード

LPRUN ビットと LPSDSR ビットがセットされると、レギュレータはメインモードから低消費電力モードへ切り替わります。その他の場合は、メインモードを維持します。LPRUN がリセットされると、レギュレータはメインモード動作に戻ります。

MCU が低消費電力 run モードの場合、LPSDSR のリセットは禁止されています。LPSDSR は低消費電力モード開始の準備として使用され、低消費電力モードを開始する際に選択するレギュレータ設定をシステムに示します。LPSDSR ビットは、LPRUN ビットをセットする前にセットする必要があります。LPRUN ビット = 0 のときにのみ、LPSDSR をリセットすることができます。

- 0 : 低消費電力 run モードで電圧レギュレータはメインモード
- 1 : 低消費電力 run モードで電圧レギュレータは低消費電力モード

ビット 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 12:11 **VOS[1:0]**: 電圧スケーリング範囲選択

これらのビットを使って、内部 レギュレータ電圧範囲を選択します。
電源インタフェースのリセット前に、RCC_APB1RSTR レジスタの PWRRST ビットをリセットして、これらのビットを「10」に設定し、それにしたがってシステム周波数を設定する必要があります。

- 00 : 禁止 (ビットは不変で前の値を維持、電圧スケーリング範囲は不変)
- 01 : 1.8 V (範囲 1)
- 10 : 1.5 V (範囲 2)
- 11 : 1.2 V (範囲 3)

ビット 10 **FWU**: 高速ウェイクアップ

このビットは、ULP ビットとの組み合わせで機能します。

ULP = 0 の場合、FWU を無視

ULP = 1 かつ FWU = 1 の場合、低消費電力モードを終了する際に V_{REFINT} スタートアップ時間を無視 PWR_CSR レジスタの VREFINTRDYF フラグは、 V_{REFINT} が再度レディになるタイミングを示します。

ULP=1 かつ FWU = 0 の場合、 V_{REFINT} がレディのときにのみ (スタートアップ時間の後)、低消費電力モードが終了します。RCC_APB1RSTR レジスタの PWRRST ビットをリセットしても、このビットはリセットされません。

- 0 : V_{REFINT} がレディのときにのみ低消費電力モードが終了します。
- 1 : 低消費電力モードが終了すると、 V_{REFINT} スタートアップ時間は無視されます。



ビット 9 **ULP** : 超低消費電力モード

セットすると、低消費電力モードで V_{REFINT} が切り離されます。RCC_APB1RSTR レジスタの PWRRST ビットをリセットしても、このビットはリセットされません。

0 : 低消費電力モードで V_{REFINT} がオン

1 : 低消費電力モードで V_{REFINT} がオフ

ビット 8 **DBP** : バックアップ書き込み保護の無効化

リセット状態では、RTC、RTC レジスタ、RCC_CSR レジスタは、擬似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、RCC_CSR レジスタへのアクセスを無効化

1 : RTC、RTC バックアップ、RCC_CSR レジスタへのアクセスを有効化

注 : *HSE の 2、4、8、16 分周が RTC クロックとして使用される場合、このビットは 1 にセットしたままに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、ソフトウェアから書き込みます。

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 (内部で V_{REFINT} と比較)

PVD_IN 入力 (PB7) は、PLS[2:0] = 111 のとき、アナログ入力に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/クリアします。

0 : PVD は無効

1 : PVD を有効

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 : 影響無し

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 : 影響無し

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

ビット 1 **PDDS** : パワーダウンドィープスリープ

このビットは、ソフトウェアによってセット/クリアします。

0 : CPU がディープスリープに入ると、STOP モードに入ります。レギュレータは低消費電力モードになります。

1 : CPU がディープスリープに入ると、STANDBY モードに入ります。

ビット 0 **LPSDSR** : Low-power deepsleep/sleep/低消費電力 run

– DeepSleep/Sleep モード

このビットをセットすると、CPU が sleep モードまたは deepsleep モードになるときレギュレータは低消費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き、レギュレータはメインモードに戻ります。

– 低消費電力 run モード

このビットをセットすると、LPRUN がセットされるときレギュレータは低消費電力モードに切り替わります。LPRUN がリセットされると、レギュレータはメインモード動作に戻ります。

このビットは、ソフトウェアによってセット/クリアします。

0 : deepsleep/Sleep/低消費電力 run モードで電圧レギュレータはオンになります。

1 : deepsleep/Sleep/低消費電力 run モードで電圧レギュレータは低消費電力モードになります。

5.4.2 PWR 電源制御/ステータスレジスタ (PWR_CSR)

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るリセットではありません。)

このレジスタを読み出すには、通常の APB 読出しに対して、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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み					EWUP3	EWUP2	EWUP1	予約済み	REG LPF	VOSF	VREFINT RDYF	PVDO	SBF	WUF	
					rw	rw	rw		r	r	r	r	r	r	

ビット 31:11 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 **EWUP3**: イネーブル WKUP ピン 3

このビットは、ソフトウェアによってセット/クリアします。

0: WKUP ピン 3 は汎用 I/O として使用されます。この WKUP ピン 3 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 3 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 3 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 9 **EWUP2**: イネーブル WKUP ピン 2

このビットは、ソフトウェアによってセット/クリアします。

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**: イネーブル WKUP ピン 1

このビットは、ソフトウェアによってセット/クリアします。

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、リセット値に保持する必要があります。

ビット 5 **REGLPF**: レギュレータ LP フラグ

MCU が低消費電力 run モードのとき、このビットはハードウェアからセットされます。

MCU が低消費電力 run モードから抜け出るとき、レギュレータがメインモードでレディになるまで、このビットは 1 を維持します。レギュレータメインモードを待つときは、このビットのポーリングが推奨されます。レギュレータがレディのとき、このビットはハードウェアからリセットされます。

0: メインモードでレギュレータはレディ

1: レギュレータ 電圧は低消費電力モード

ビット 4 VOSF : 電圧スケーリング選択フラグ

電圧範囲を変更した後、内部レギュレータがレディになるまで遅延が必要です。VOSF ビットは、レギュレータが PWR_CR レジスタの VOS ビットで指定された電圧レベルに到達したことを示します。

PWR_CR レジスタの VOS[1:0] が変更されたとき、このビットはリセットされます。

レギュレータがレディになるとセットされます。

- 0 : 選択された電圧範囲でレギュレータがレディ
- 1 : レギュレータ 電圧出力が要求された VOS レベルへ変化中

ビット 3 VREFINTRDYF : 内部電圧基準 (V_{REFINT}) レディフラグ

このビットは、内部電圧基準 V_{REFINT} の状態を示します。

- 0 : V_{REFINT} はオフ
- 1 : V_{REFINT} はレディ

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。
- 1 : V_{DD} は PLS[2:0] ビットで選択された PVD 閾値より低い。

注 : PVD は STANDBY モードによって停止します。このため、このビットは STANDBY またはリセット後は、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でセットされ、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 ビットに 1 を書き込むことでクリアされます。

- 0 : デバイスは STANDBY モードに入っていない
- 1 : デバイスは STANDBY に入っている

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かのいずれかによってクリアされます。

- 0 : ウェイクアップイベントは発生していない
- 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 タイムスタンパイイベント、RTC ウェイクアップのいずれかからウェイクアップイベントを受信した

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によって) WKUP ピンが有効化された場合、追加のウェイクアップイベントが検出されます。

5.4.3 PWR レジスタマップ

次の表は PWR レジスタの一覧です。

表 33. 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	予約済み																		LPRUN	予約済み	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	予約済み																						EWUP3	EWUP2	EWUP1	予約済み			REGLPF		VOSF	VREFINTRDYF		PVDO	SBF	WUF
	リセット値																																				

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。



6 リセットおよびクロック制御 (RCC)

6.1 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。

6.1.1 システムリセット

システムリセットは、RTC、RTC バックアップレジスタ、制御/ステータスレジスタ、RCC_CSR を除くすべてのレジスタにリセット値を設定します。

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。

1. NRST ピンのローレベル (外部リセット)
2. 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)
3. 独立型ウォッチドッグのカウント終了条件 (IWDG リセット)
4. 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](#)参照)
5. 低電力管理リセット ([低電力管理リセット](#)参照)
6. 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](#)参照)
7. Standby モードの終了

リセットソースは、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 6.3.14](#)参照)。

ソフトウェアリセット

Cortex®-M3 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と、デバイスのソフトウェアリセットが強制的に行われます。詳細については、Cortex®-M3 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。

1. STANDBY モードに入るときに生成されるリセット :
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によって有効になります。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 モードに入る代わりにデバイスがリセットされます。
2. STOP モードに入るときのリセット :
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によって有効になります。この場合、STOP モード遷移シーケンスが正常に実行されるたびに、STOP モードに入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

FLASH_PECR レジスタの OBL_LAUNCH ビット (ビット 18) がセットされると、オプションバイトローダリセットが生成されます。このビットを使って、ソフトウェアからオプションバイトのローディングが開始されます。

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 \(フラッシュ\)](#) を参照してください。

6.1.2 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。

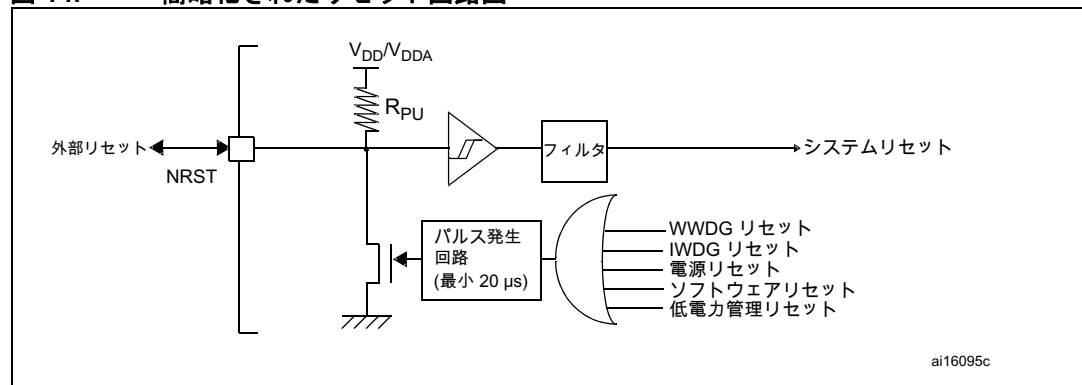
1. パワーオン/パワーダウンリセット (POR/PDR リセット)
2. BOR リセット

電源リセットは、RTC ドメインを含むすべてのレジスタにリセット値をセットします (図 14 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 48 : ベクタテーブル (Cat.1 および Cat.2 デバイス)、表 49 : ベクタテーブル (Cat.3 デバイス)、表 50 : ベクタテーブル (Cat.4 および Cat.5 デバイス) 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対して 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 14. 簡略化されたリセット回路図



6.1.3 RTC レジスタとバックアップレジスタのリセット

RTC ペリフェラルレジスタ、RTC クロックソース選択レジスタ (RCC_CSR)、バックアップレジスタは、次のイベントが発生したときにのみリセットされます。

1. RCC_CSR レジスタの RTCRST ビットをセットすると、ソフトウェアリセットが発生します (セクション 6.3.14 参照)。
2. 電源リセット (BOR/POR/PDR)

6.2 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。

- HSI (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

MSI は、リセットからのスタートアップ、Stop または Standby 低消費電力モードからのウェイクアップの後、システムクロックソースとして使われます。

デバイスは、また、次の 2 つのセカンダリクロックソースを持ちます。

- 37 kHz 低速内蔵 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) 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、電力消費を最適化できます。

複数のプリスケアラを利用して、AHB 周波数、高速 APB (APB2) および低速 APB (APB1) ドメインの周波数を設定することができます。AHB、APB1、APB2 ドメインの最大周波数は 32 MHz です。これはデバイス電圧範囲に依存します。詳細については、PWR の章のダイナミック電圧スケール管理セクションを参照してください。

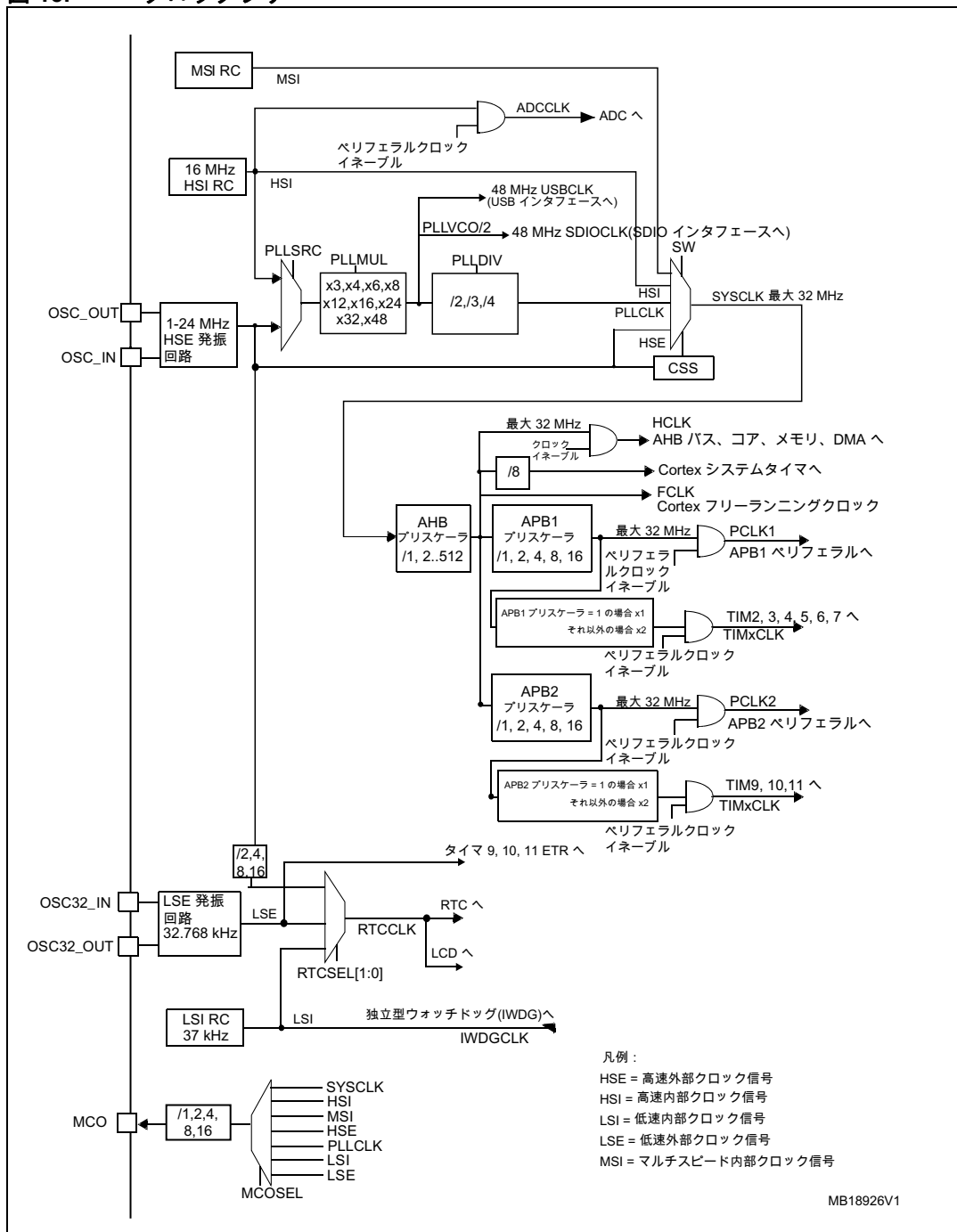
すべてのペリフェラルクロックは、システムクロック (SYSCLK) から派生されます。ただし、次のものを除きます。

- PLL VCO クロックから来る 48 MHz のクロック USB および SDIO クロック。
- 常に HSI クロックである ADC クロック。1、2 または 4 分周により、クロック周波数をデバイス動作状態に合わせることができます。詳細については、PWR の章の動作電源範囲のセクションを参照してください。
- LSE、LSI または 1 MHz HSE_RTC (HSE をプログラマブルプリスケアラで分周) から来る RTC/LCD クロック。
- 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 クロック周波数以上である必要があります。

RCC は Cortex®システムタイマ (SysTick) 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構成できます。

図 15. 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 プリスケアラが1の場合、タイマクロック周波数は接続されている 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。

FCLK は Cortex®-M3 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 ARM® Cortex®-M3 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。

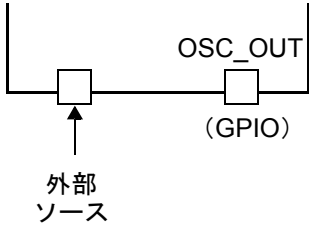
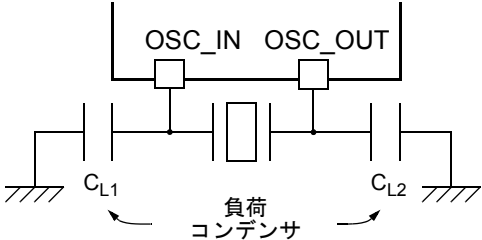
6.2.1 HSE クロック

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安定化時間を少なくするために、共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 16. 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	
クリスタル／セラミック発振子	

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 の周波数を使用することができます。このモードは **クロック制御レジスタ RCC_CR (セクション 6.3.1 参照)** の HSEBYP ビットと HSEON ビットをセットすることで指定します。50 % まで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_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 16 参照)。

外部クリスタル／セラミック発振子 (HSE クリスタル)

1~24 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利点があります。

関連するハードウェア構成を図 16 に示します。詳細については、データシートの電気特性のセクションを参照してください。

RCC_CR レジスタ (セクション 6.3.1 参照) の HSERDY フラグは、HSE オシレータの安定/不安定を示します。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 クロックはリリースされません。**RCC_CR レジスタ**で割り込みが有効化された場合、割り込みが発生します。

HSE クリスタルは、**RCC_CR レジスタ** の HSEON ビットを使ってオン/オフすることができます。

6.2.2 HSI クロック

HSI クロック信号は、内部 16 MHz RC オシレータから発生されます。これは、システムクロックまたは PLL 入力として直接使用することができます。

HSI 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、較正を実施していても、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。

較正

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、周囲温度 $T_A = 30^\circ\text{C}$ で 1 % の精度を確保するよう、製造時に ST 社においてデバイスごとの較正が行われています。

リセット直後に 製造時の較正值は、**内部クロックソース校正レジスタ (RCC_ICSCR) (セクション 6.3.2 参照)** の HSICAL[7:0] 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にさらされる場合は、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 HSITRIM[4:0] ビットを使用して、アプリケーションで HSI 周波数をトリミングできます。デフォルトの HSITRIM 値は 16 で、HSI オシレータを公称周波数 8 MHz に調整します。HSITRIM 値を大きくすると、HSI オシレータ周波数が高くなります。トリミングステップ (F_{HSITRIM}) は、2 つの連続する HSICAL ステップの間で約 40 kHz です。トリミングコードが 16 の倍数の場合、トリミングコード変化に対する周波数応答は非単調性を示します。詳細については、デバイスデータシートを参照してください。HSI 周波数変動の測定方法については、**セクション 6.2.14: TIM9/TIM10/TIM11 による内部/外部クロックの測定**を参照してください。

RCC_CR の HSIRDY フラグは、HSI 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、HSI RC 出力クロックはリリースされません。

HSI RC オシレータは、**RCC_CR レジスタ**の HSION ビットを使用してオン/オフできます。

6.2.3 MSI クロック

MSI クロック信号は、内部RC オシレータから発生されます。周波数範囲は、RCC_ICSCR レジスタの MSIRANGE[2:0] ビットを使ってソフトウェアから調整することができます ([セクション 6.3.2: 内部クロックソース校正レジスタ \(RCC_ICSCR\)](#) 参照)。65.536 kHz、131.072 kHz、262.144 kHz、524.288 kHz、1.048 MHz、2.097 MHz (デフォルト値)、4.194 MHz の 7種類の周波数範囲を使用することができます。

MSI は、リセットからの再起動、Stop または Standby 低消費電力モードからのウェイクアップの後、システムクロックとして使われます。リセット後の再起動または Standby からのウェイクアップでは、MSI 周波数はデフォルト値に設定されます。RCC_ICSCR レジスタ値は STOP モードで保持されるため、ウェイクアップ後の MSI 周波数と較正は、STOP モードに入る前と同じです。

MSI RC オシレータには、低価格で (外付け部品不要) 低消費電力クロックソースを提供する利点があります。このクロックは、消費電力とウェイクアップ時間を小さくするため低消費電力モードで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 の安定/不安定を示します。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 出力クロックはリリースされません。

MSI RC は、[RCC_CR レジスタ](#)の HSEON ビットを使ってオン/オフすることができます ([セクション 6.3.1](#)参照)。

このクロックは、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 6.2.9: クロックセキュリティシステム \(CSS\) \(138 ページ\)](#) を参照してください。

較正

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、周囲温度 $T_A = 30^\circ\text{C}$ で 1 % の精度を確保するよう、製造時に ST 社においてデバイスごとの較正が行われています。

リセット後、工場較正值が 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にさらされる場合は、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については、[セクション 6.2.14: TIM9/TIM10/TIM11 による内部/外部クロックの測定](#)を参照してください。

6.2.4 PLL

内部 PLL クロックは、HSI RC または HSE クリスタルから供給することができます。システムクロックとして、また USB ペリフェラルに対する 48 MHz クロックとして使用されます（[図 15](#)と [セクション 6.3.1：クロック制御レジスタ \(RCC_CR\)](#) を参照してください）。

PLL 入力クロック周波数は、2~24 MHz である必要があります。

必要とする周波数は、PLL の通倍比と出力分周比を使って実現することができます。

- アプリケーションで USB または SDIO インタフェースを使用する場合、PLL VCO クロック（PLL通倍比で指定）は、96 MHz の出力周波数に設定する必要があります。これは、USB または SDIO へ 48 MHz クロックを供給するために必要です（SDIOCLK または USBCLK = PLLVCO/2）。
- システムクロックは、PLL VCO を出力分周比で分周して供給されます。

注： *アプリケーションソフトウェアはPLL 通倍比を正しく設定して、製品が範囲 1 の場合 PLLVCO として 96 MHz を、製品が範囲 2 の場合 PLLVCO として 48 MHz を、製品が範囲 3 の場合 PLLVCO として 24 MHz を、それぞれ超えないようにする必要があります。*

また、出力分周比も正しく設定して、SYSCLK として 32 MHz を超えないように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 です（HSE を PLL ソースとして使用の場合）。

PLL の設定（ソースクロック、通倍比、出力分周比の選択）は、PLL を有効化する前に行う必要があります。PLL が有効になると、これらのパラメータを変更できません。

PLL 設定を変更するときは、次の手順で行います。

1. PLLON に 0 を設定して PLL を無効化します。
2. PLLRDY がクリアされるまで待ちます。PLLRDY。PLL は停止します。
3. 必要なパラメータを変更します。
4. PLLON に 1 を設定して、PLL を再度有効化します。

RCC_CIR レジスタで割り込みを有効化した場合、PLL がレディになると割り込みが発生します（[セクション 6.3.4](#)参照）。

6.2.5 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、低電力で高精度な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/オフを切り替えます（[セクション 6.3.14](#)参照）。

RCC_CSR レジスタの LSERDY 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からセットされるまで、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_CIR レジスタで割り込みが有効化された場合、割り込みが発生します（[セクション 6.3.4](#)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 の周波数を使用することができます。このモードは [RCC_CR](#)（[セクション 6.3.1](#)参照）の LSEBYP ビットと LSEON ビットをセットすることで選択されます。50 % まで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、一方、OSC_32_OUT ピンは GPIO ピンとして使用することができます（[図 16](#)参照）。

6.2.6 LSI クロック

LSI RC は、独立型ウォッチドッグ (IWDG) のために STOP モードおよび STANDBY モードで動作する低消費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_CR 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 6.3.14](#)参照)。

RCC_CSR レジスタの LSIRDY フラグは、低速内部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、クロックはリリースされません。RCC_CIR レジスタで割り込みが有効化された場合、割り込みが発生します ([セクション 6.3.4](#)参照)。

LSI の測定

LSI オシレータの周波数のばらつきを測定して、正確な RTC 基準時間および／または妥当な精度の IWDG タイムアウト (これらのペリフェラルのクロックソースとして LSI が使用されているとき) を確保できます。詳細については、データシートの電気特性のセクションを参照してください。LSI 周波数変動の測定方法については、[セクション 6.2.14: TIM9/TIM10/TIM11 による内部/外部クロックの測定](#)を参照してください。

6.2.7 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。

- HSI 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は、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、ターゲットクロックソースの準備ができていない場合のみ行われます (スタートアップ遅延後にクロックが安定しているか、PLL がロックされている状態)。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、準備ができていないクロックと、システムクロックとして使用されているクロックを示します。

6.2.8 システムクロックソース周波数対電圧範囲

次の表に、製品電圧範囲に依存する様々なクロックソース周波数を示します。

表 34. システムクロックソース周波数

製品電圧範囲	クロック周波数			
	MSI	HSI	HSE	PLL
範囲 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 (PLLVC0 max = 96 MHz)
範囲 2 (1.5 V)	4.2 MHz	16 MHz	16 MHz	16 MHz (PLLVC0 max = 48 MHz)
範囲 3 (1.2 V)	4.2 MHz	NA	8 MHz	4 MHz (PLLVC0 max = 24 MHz)



6.2.9 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ソフトウェアから起動できます。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 クロックに障害が検出された場合、このオシレータは自動的に無効化され、ソフトウェアに障害を通知する割り込み (クロックセキュリティシステム割り込み (CSSI)) が生成されるので、MCU は対応処理を行うことができます。CSSI は、Cortex[®]-M3 NMI (Non-Maskable Interrupt) 例外ベクタにリンクされています。

注： *CSS が有効になると、HSE クロック障害が発生した場合、CSS 割り込みが発生し、NMI が自動的に生成されます。NMI は、CSS 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。結果として、NMI ISR では、RCC_CIR レジスタの CSSC ビットをセットすることによって、CSS 割り込みをクリアしなければなりません。*

MSE オシレータが直接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ていて、PLL クロックがシステムクロックとして使用されていることを意味します)、障害が検出されると、システムクロックが HSI 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。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 のクロック入力の場合、障害が発生すると、PLL も無効になります。

6.2.10 LSE のクロックセキュリティシステム

Cat.2、Cat.3、Cat.4、Cat.5 デバイスでは、RCC_CSR レジスタの LSECSSON ビットに書き込みを行って、LSE のクロックセキュリティシステムをソフトウェアから起動することができます。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 での故障検出後にのみ、無効化することができます。LSECSSON は、LSE と LSI を有効化し (LSEON と LSION を有効化)、レディになり (ハードウェアから LSERDY と LSIRDY をセット)、RTCSEL を使って RTC クロックを選択した後に書き込む必要があります。

CSS 故障が外部 LSE オシレータで検出されると、LSECSSD フラグがセットされます。故障イベントは EXTI ライン 19 にリンクされます。Standby モードから MCU をウェイクアップさせるか、または run、Sleep、Stop 低消費電力モードでの TAMPER_STAMP 割り込みルーチン内で故障イベントへのサービスを可能にする割り込みを発生させます。

割り込み発生を有効化するときは、RCC_CIR 割り込みレジスタの LSECSIE ビットをセットする必要があります (セクション 6.3.4 : クロック割り込みレジスタ (RCC_CIR) (148 ページ) 参照)。

CSS 故障の場合、LSE クロックの RTC への供給を停止し、フェイルセーフモードをファームウェアから管理する必要があります (LSE オシレータを停止し、RTC クロックソースを LSI または HSE クロックへ切り替えます)。

6.2.11 RTC クロックと LCD クロック

RTC と LCD は同じクロックソースを持ち、LSE、LSI、または HSE 1 MHz クロック（HSE をプログラマブルなプリスケアラで分周）を選ぶことができます。RCC_CSR レジスタの RTCSEL[1:0] ビット（[セクション 6.3.14](#) 参照）および RCC_CR レジスタの RTCPRE[1:0] ビット（[セクション 6.3.1](#) 参照）を設定して選択します。

RTC クロックソースと LCD クロックソースをいったん選択した後、その選択を変更する唯一の方法は、RCC_CSR レジスタの RTCRST ビットをセットするか、または POR によります。

LSE または LSI を RTC クロックソースとして使用する場合、RTC は Stop 低消費電力モードと Standby 低消費電力モードで動作を続けるため、ウェイクアップソースとして使用することができます。ただし、HSE が RTC クロックソースの場合、RTC は Stop 低消費電力モードと Standby 低消費電力モードで使用できませんが、LSE または LSI を RTC クロックソースとして使用する場合には、LCD を Stop 低消費電力モードで使用することができます。

注： *APB1 クロック周波数が RTC クロック周波数の 7 倍 ($7 \times \text{RTCLK}$) より小さい場合に RTC カレンダーレジスタを読み出せるようにするときは、ソフトウェアからカレンダー時間と日付レジスタを 2 回読み出す必要があります。*

RTC_TR の 2 回目の読み出しで初回の読み出しと一致した場合、そのデータが正しいことが保証されます。その他の場合、3 回目の読み出しアクセスを行う必要があります。

6.2.12 ウォッチドッグクロック

独立型ウォッチドッグ (IDG) 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 オシレータの立ち上がりの過渡状態が終わると、このクロックが IWDG に供給されます。

6.2.13 クロック出力機能

マイクロコントローラクロック出力 (MCO) 機能を使うと、設定可能なプリスケアラ（1、2、4、8、16）を使って外部 MCO ピン（PA8）へクロックを出力することができます。対応する GPIO ポートの構成レジスタをオルタネート機能モードで設定する必要があります。次の 7 つの信号を MCU クロックとして選択することができます。

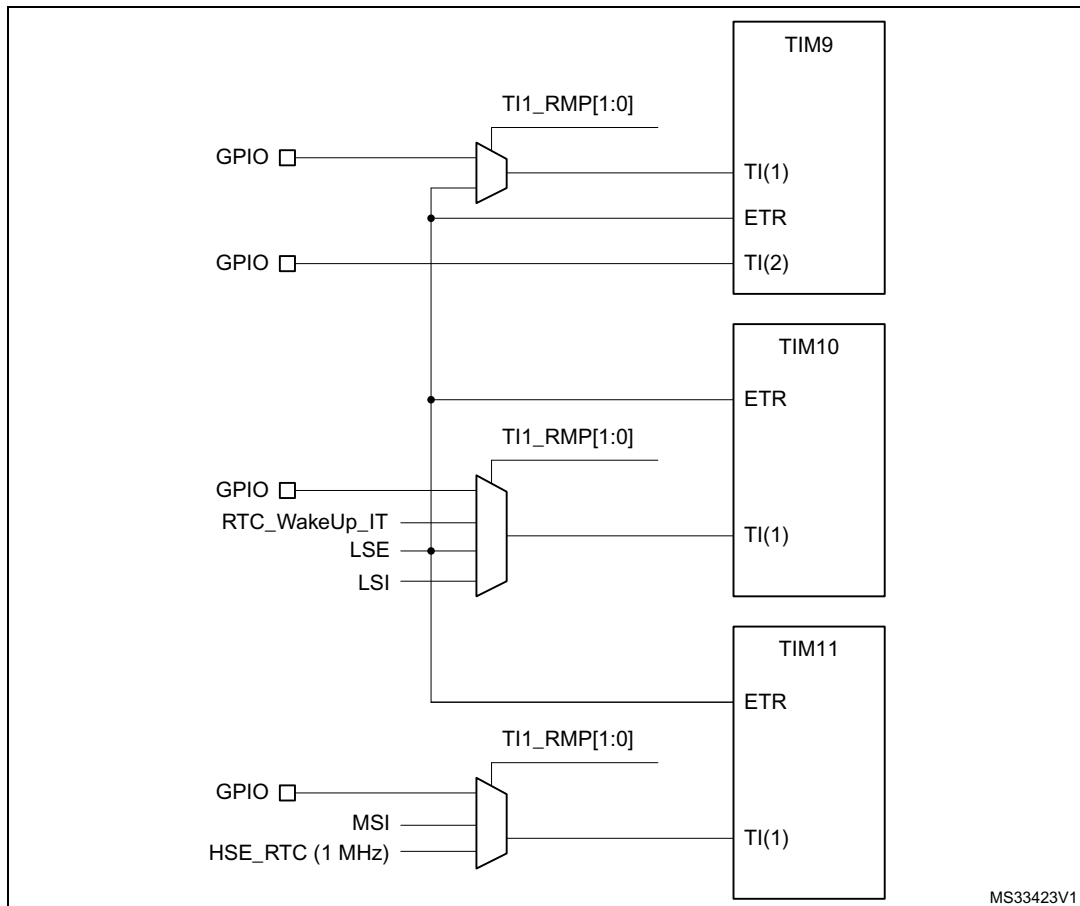
- SYCLK
- HSI
- MSI
- HSE
- PLL
- LSI
- LSE

選択は、RCC_CFGR レジスタの MCOSEL[2:0] ビットから制御されます（[セクション 6.3.3](#) 参照）。

6.2.14 TIM9/TIM10/TIM11 による内部/外部クロックの測定

図 17 に示すように、TIM9/TIM10/TIM11 チャンネル 1 入力キャプチャを使ってすべての内蔵クロックソースジェネレータの周波数を間接的に測定することができます。

図 17. TIM9/TIM10/TIM11 チャンネル 1 入力キャプチャを使用する周波数測定



各タイマには、入力キャプチャをトリガする I/O または内部クロックを選択する入力マルチプレクサがあります。この選択は、TIMx_OR レジスタの TI1_RMP [1:0] ビットを使って行います。

TIM9 と TIM10 の場合、LSE をチャンネル 1 入力キャプチャに接続する主要な目的は、HSI 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（このため、HSI または MSI をシステムクロックソースとして使用する必要があります）。LSE 信号の連続するエッジ間の、それぞれ HSI と MSI のクロック数が、内部クロック周期の測定値を提供します。LSE クリスタルの高精度（通常数十 ppm）を利用すると、同じ分解能で内部クロック周波数を求めることができるため、ソースを調整して製造/温度および/または電圧による周波数変化を補償することができます。

MSI オシレータと HSI オシレータには、ユーザからアクセス可能な、この目的に使用する専用の較正ビットがあります。

基本的には相対測定（例えば HSI/LSE 比）を行うことで、精度は 2 つのクロックソース間の比に密接に関係します。比が大きいほど、測定精度が良くなります。

ただし、MSI クロックが低い場合（1 MHz 以下）、十分な精度は得られません。この場合、次が推奨されます。

- 連続した複数のキャプチャ結果を積算します。
- タイマの入力キャプチャプリスケアラ（8 周期ごとに最大 1 キャプチャ）を使います。
- RTC ウェイクアップ割り込み信号（RTC クロックが LSE の場合）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よくなります。このため、RTC ウェイクアップ割り込みを有効化する必要があります。

TIM10 も LSI の測定に使うことができます。これはクリスタルを使わないアプリケーションで役立ちます。超低消費電力 LSI オシレータの製造プロセス変動は大きいので、HSI クロックソースの関数として測定することにより、HSI の高精度で周波数を求めることができます。

最後に、TIM11 にはその他の 2 つソースがあります。クリスタル不使用アプリケーションで TIM10 が LSI を使用したように、TIM11 は MSI を使用することができます。HSE_RTC 周波数（プログラマブルなプリスケアラで HSE を分周）は比較的高いため（1 MHz）、相対周波数測定の精度は良くないので、これの主目的は外部クリスタル周波数の概略表示です。これは、高調波またはサブ高調波周波数（-50/+100 % 偏差）を測定できることを要求した IEC 60730/IEC 61335 規格の条件を満たす場合に便利です。

6.2.15 TIM9/TIM10/TIM11 用のクロック非依存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 の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、プロセッサ状態（プロセッサの停止、または低速、中速、高速動作）を考慮しないでタスクスケジューリングが可能になります。

この目的のために、LSE クロックは内部で、追加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（[図 17 \(140 ページ\)](#) 参照）。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 つのコンペアチャンネルを持つ最大 3 つの独立なタイムベース（自動再ロード機能を使用）が得られます。例えば、TIM9 の自動再ロード割り込みは、メインティックの 250 ms 後に追加割り込みが発生する 1 秒ティック割り込み用に設定することができます。

注： この設定では、外部クロック（LSE）と APB クロックとの間の比が少なくとも 2 であることを確認する必要があります。アプリケーションで LSE クロック周波数の 2 倍（一般に LSE = 32.768 kHz であるため、 $2 \times \text{LSE} = 65.536 \text{ kHz}$ ）より低い APB クロック周波数を使う場合は、タイマの外部トリガプリスケアラ機能を使う必要があります。この機能は、ETR クロックを最大 8 分周することができます。

6.3 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。

6.3.1 クロック制御レジスタ (RCC_CR)

アドレスオフセット : 0x00

リセット値 : 0b0XX0 0000 0000 0X00 0000 0000、ここで X は不定

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
予約済み	RTCPRE[1:0]		CSS ON	予約済み	PLL RDY		PLLON	予約済み					HSE BYP	HSE RDY	HSE ON
	rw	rw	rw		r	rw							rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み						MSI RDY	MSION	予約済み					HSI RDY	HSION	
						r	rw						r	rw	

ビット 31 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 30:29 **RTCPRE[1:0]** RTC/LCD プリスケアラ

HSE から 1 MHz クロックを発生するため、これらのビットはソフトウェアからセットまたはリセットされます。このプリスケアラは、HSE が有効化されている場合 (HSEON = 1) 変更できません。これらのビットはパワーオンリセットでリセットされ、システムリセット後にその値を維持します。

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 28 **CSSON** : クロックセキュリティシステムイネーブル

このビットは、クロックセキュリティシステム (CSS) を有効化/無効化するために、ソフトウェアからセット/クリアされます。CSSON をセットすると、クロック検出器が HSE オシレータがレディのときハードウェアから有効化され、オシレータ故障が検出されたときハードウェアから無効化されます。

0 : クロックセキュリティシステム OFF (クロック検出器 OFF)

1 : クロックセキュリティシステム ON (HSE オシレータが安定している場合クロック検出器が ON、その他の場合は OFF)

ビット 27:26 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 25 **PLL RDY** : PLL クロックレディフラグ

このビットはハードウェアからセットされ、PLL がロック状態であることを示します。

0 : PLL アンロック

1 : PLL ロック

ビット 24 **PLLON** : PLL イネーブル

このビットは、PLL を有効化するために、ソフトウェアからセット/クリアされます。

STOP または STANDBY モードに入るときに、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や、システムクロックになるように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 23:19 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 18 HSEBYP : HSE クロックバイパス

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、ソフトウェアからセット/クリアされます。外部クロックを HSEON ビットを使って有効化し、デバイスで使用できるようにする必要があります。

HSE オシレータを無効化している場合にのみ、HSEBYP ビットを書き込むことができます。

このビットはパワーオンリセットでリセットされ、システムリセット後にその値を維持します。

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

このビットはハードウェアからセットされ、HSE オシレータが安定していることを示します。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

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 モードに入るときに、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、このビットをリセットできません。

0 : HSE オシレータ OFF

1 : HSE オシレータ ON

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 MSIRDY : MSI クロックレディフラグ

このビットはハードウェアからセットされ、MSI オシレータが安定していることを示します。

0 : MSI オシレータが非レディ

1 : MSI オシレータがレディ

注 : *MSION ビットがクリアされた後、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

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 モードから抜け出るとき、またはシステムクロックとして直接または間接的に使用された HSE オシレータが故障したとき、ハードウェアからセットされ MSI オシレータを ON にします。MSI がシステムクロックとして使用されている場合、このビットはクリアできません。

0 : MSI オシレータ OFF

1 : MSI オシレータ ON

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 HSIRDY : 内部高速クロックレディフラグ

このビットはハードウェアからセットされ、HSI オシレータが安定していることを示します。HSION ビットがクリアされた後、HSIRDY は HSI クロックの 6 サイクル後にローになります。

0 : HSI オシレータが非レディ

1 : HSI オシレータがレディ

ビット 0 HSION : 内部 高速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。

HSI が直接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、このビットをクリアできません。

0 : HSI オシレータ OFF

1 : HSI オシレータ ON

6.3.2 内部クロックソース校正レジスタ（RCC_ICSCR）

アドレスオフセット：0x04

リセット値：0x00XX B0XX、ここで X は不定。

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
MSI 範囲 [2:0]			HSITRIM[4:0]					HSICAL[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

このビットは、MSI を校正するために、ソフトウェアからセットされます。

これらのビットは、MSICAL[7:0] ビットに追加されるユーザプログラミング可能トリミング値を提供します。内蔵 MSI RC の周波数に影響する電圧や温度の変化を補償できるように設定することができます。

ビット 23:16 **MSICAL[7:0]**：MSI クロック校正

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。

ビット 15:13 **MSI 範囲[2:0]**：MSI クロック範囲

これらのビットはソフトウェアから設定され、MSI の周波数範囲を選択します。次の 7 種類の周波数範囲を使用することができます。

000:範囲 0、65.536 kHz 前後
 001:範囲 1、131.072 kHz 前後
 010:範囲 2、262.144 kHz 前後
 011:範囲 3、524.288 kHz 前後
 100:範囲 4、1.048 MHz 前後
 101:範囲 5、2.097 MHz（リセット値）前後
 110:範囲 6、4.194 MHz 前後
 111：設定禁止

ビット 12:8 **HSITRIM[4:0]**：高速内部クロックトリミング

これらのビットは、HSICAL[7:0] ビットに追加されるユーザプログラミング可能トリミング値を提供します。内蔵 HSI RC の周波数に影響する電圧や温度の変化を補償できるように設定することができます。

ビット 7:0 **HSICAL[7:0]** 内部高速クロック校正

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。

6.3.3 クロック設定レジスタ (RCC_CFGR)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

アクセス : 0 ≤ 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 個の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
予約済み	MCOPRE[2:0]			予約済み	MCOSEL[2:0]			PLLDIV[1:0]		PLLMUL[3:0]				予約済み	PLL SRC
	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
予約済み	PPRE2[2:0]			PPRE1[2:0]			HPRE[3:0]				SWS[1:0]		SW[1:0]		
	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r	r	r/w	r/w	

ビット 31 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 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 出力を有効化する前に、このプリスケアラを変更することが推奨されます。
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 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 26:24 **MCOSEL[2:0]** : 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。
000 : MCO 出力を無効化、MCO のクロックなし
001 : SYSCCLK クロックを選択
010 : HSI オシレータクロックを選択
011 : MSI オシレータクロックを選択
100 : HSE オシレータクロックを選択
101 : PLL クロックを選択
110 : LSI オシレータクロックを選択
111 : LSE オシレータクロックを選択

注 : このクロック出力は、スタートアップ時または MCO クロックソースの切り替え中は、サイクルが短縮されることがあります。

ビット 23:22 **PLLDIV[1:0]** : PLL 出力分周比

ソフトウェアからこれらのビットをセット/クリアして、PLL VCO クロックからの PLL 出力クロック分周比を制御します。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 通倍比

ソフトウェアからこれらのビットを書き込んで、PLL VCO クロックを発生する PLL 通倍比を指定します。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 クロックエントリ x 3
0001 : PLLVCO = PLL クロックエントリ x 4
0010 : PLLVCO = PLL クロックエントリ x 6
0011 : PLLVCO = PLL クロックエントリ x 8
0100 : PLLVCO = PLL クロックエントリ x 12
0101 : PLLVCO = PLL クロックエントリ x 16
0110 : PLLVCO = PLL クロックエントリ x 24
0111 : PLLVCO = PLL クロックエントリ x 32
1000 : PLLVCO = PLL クロックエントリ x 48
その他 : 設定禁止

注意 : PLL VCO クロック周波数は、製品が範囲 1 の場合 96 MHz を、製品が範囲 2 の場合 48 MHz を、製品が範囲 3 の場合 24 MHz を、それぞれ超えることはできません。

ビット 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 PLLSRC : PLL エントリクロックソース

このビットは、PLL クロックソースを選択するために、ソフトウェアからセット/クリアされます。PLL が無効化されている場合にのみこのビットに書き込みを行うことができます。

0 : HSI 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 最小入力クロック周波数は 2 MHz です。

ビット 15:14 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 13:11 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、ソフトウェアからセット/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 低速クロック (PCLK1) の分周比を制御するために、ソフトウェアからセット/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、ソフトウェアからセット/クリアされます。

注意： デバイス電圧範囲に応じて、ソフトウェアからこれらのビットを正しく設定して、システム周波数が最大許容周波数を超えないようにする必要があります（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管理セクションを参照してください）。これらのビットに対する書き込み動作の後および電圧範囲を狭くする前に、このレジスタを読み出して、新しい値が考慮されていることを確認して下さい。

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、システムクロックとして使用されているクロックソースを示すために、ハードウェアからセット/クリアします。

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 : HSI 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 のソースを選択するために、ソフトウェアからセット/クリアされます。STOP および STANDBY モードを終了するときや、直接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、強制的 MSI を選択するために、ハードウェアによってセットされます（クロックセキュリティシステムが有効な場合）。

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 : HSI 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。

6.3.4 クロック割り込みレジスタ (RCC_CIR)

アドレスオフセット : 0x0C

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
予約済み								CSSC	LSECSSC	MSI RDYC	PLL RDYC	HSE RDYC	HSI RDYC	LSE RDYC	LSI RDYC
								w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	LSECSSIE	MSI RDYIE	PLL RDYIE	HSE RDYIE	HSI RDYIE	LSE RDYIE	LSI RDYIE	CSSF	LSECSSF	MSI RDYF	PLL RDYF	HSE RDYF	HSI RDYF	LSE RDYF	LSI RDYF
	r/w	r/w	r/w	r/w	r/w	r/w	r/w	r	r	r	r	r	r	r	r

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 **CSSC** : 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、CSSF 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 : CSSF フラグをクリアします。

ビット 22 **LSECSSC** : LSE CSS 割り込みクリア

ソフトウェアからセットされ、LSECSSF をクリアします。クリア完了時ハードウェアからリセットされます。

0 : LSECSSF を非クリア

1 : LSECSSF をクリア

注 : このビットは、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 21 **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。

0 : 影響無し

1 : MSIRDYF をクリア

ビット 20 **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。

0 : 影響なし。

1 : PLLRDYF がクリアされます。

ビット 19 **HSE RDYC** : HSE レディ割り込みクリア

このビットは、HSE RDYF 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 : HSE RDYF がクリアされます。

ビット 18 **HSIRDYC** : HSI レディ割り込みクリア

このビットは、HSIRDYF 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 : HSIRDYF がクリアされます。

ビット 17 **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。

0 : 影響なし。

1 : LSERDYF がクリアされます。

ビット 16 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。

0 : 影響なし。

1 : LSIRDYF がクリアされます。

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 LSECSSIE : LSE CSS 割り込みイネーブル

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から生成される割り込みを許可/禁止するため、ソフトウェアからセット/リセットされます。

0 : LSE CSS 割り込みを無効化

1 : LSE CSS 割り込みを有効化

注 : このビットは、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 13 MSIRDYIE : MSI レディ割り込みイネーブル

MSI オシレータ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、ソフトウェアからセット/クリアされます。

0 : MSI レディ割り込みを無効化

1 : MSI レディ割り込みを有効化

ビット 12 PLLRDYIE : PLL レディ割り込みイネーブル

PLL ロックから発生する割り込みを有効/無効にするために、ソフトウェアからセット/クリアされます。

0 : PLL ロック割り込み無効

1 : PLL ロック割り込み有効

ビット 11 HSERDYIE : HSE レディ割り込みイネーブル

HSE オシレータ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、ソフトウェアからセット/クリアされます。

0 : HSE レディ割り込み無効

1 : HSE レディ割り込み有効

ビット 10 HSIRDYIE : HSI レディ割り込みイネーブル

HSI オシレータ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、ソフトウェアからセット/クリアされます。

0 : HSI レディ割り込み無効

1 : HSI レディ割り込み有効

ビット 9 LSERDYIE : LSE レディ割り込みイネーブル

LSE オシレータ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、ソフトウェアからセット/クリアされます。

0 : LSE レディ割り込み無効

1 : LSE レディ割り込み有効

ビット 8 LSIRDYIE : LSI レディ割り込みイネーブル

LSI オシレータ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、ソフトウェアからセット/クリアされます。

0 : LSI レディ割り込み無効

1 : LSI レディ割り込み有効

ビット 7 CSSF : クロックセキュリティシステム割り込みフラグ

HSE オシレータで障害が検出されたときに、ハードウェアからセットされます。

CSSC ビットをセットすることによって、ソフトウェアからクリアされます。

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 6 LSECSSF LSE CSS 割り込みフラグ

ソフトウェアから LSECSSC ビットに書き込みを行ってリセットします。外部 32 KHz オシレータで故障が検出されたときハードウェアからセットされ、さらに LSECSSIE ビットがセットされます。

0 : 外部 32 KHz オシレータ (LSE) で故障非検出

1 : 外部 32 KHz オシレータ (LSE) で故障を検出

注 : このビットは、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 5 MSIRDYF : MSI レディ割り込みフラグ

MSI が安定して、MSIRDYDIE がセットされていると、このビットはハードウェアからセットされます。ソフトウェアからクリアされて、MSIRDYC ビットがセットされます。

0 : MSI からクロックレディ割り込みが発生していません。

1 : MSI からクロックレディ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

PLL がロックし、PLLRDYIE ビットがセットされているときに、ハードウェアからセットされます。

PLLRDYC 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。

0 : PLL ロック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 ロック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

HSE が安定して、HSERDYDIE がセットされていると、このビットはハードウェアからセットされます。

HSERDYC 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。

0 : HSE からクロックレディ割り込みが発生していません。

1 : HSE からクロックレディ割り込みが発生しました。

ビット 2 HSIRDYF : HSI レディ割り込みフラグ

HSI が安定して、HSIRDYDIE がセットされていると、このビットはハードウェアからセットされます。

HSIRDYC 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。

0 : HSI からクロックレディ割り込みは発生していません。

1 : HSI から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

LSE が安定して、LSERDYDIE がセットされていると、このビットはハードウェアからセットされます。

LSERDYC 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。

0 : LSE からクロックレディ割り込みは発生していません。

1 : LSE から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

LSI が安定して、LSIRDYDIE がセットされていると、このビットはハードウェアからセットされます。

LSIRDYC 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。

0 : LSI からクロックレディ割り込みは発生していません。

1 : LSI からクロックレディ割り込みが発生しました。

6.3.5 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 : 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
予約済み	FSMC RST	予約済み		AES RST	予約済み	DMA2RST	DMA1RST	予約済み							
	rw			rw		rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLITFR ST	予約済み		CRCR ST	予約済み				GPIOG RST	GPIOF RST	GPIOH RST	GPIOE RST	GPIOD RST	GPIOC RST	GPIOB RST	GPIOA RST
rw			rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **FSMCRST** : FSMC 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 : FSMC をリセット

注 : このビットは、Cat.4 デバイスでのみ使用可能です。

ビット 29:28 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 **AESRST** : AES 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 : AES をリセット

注 : このビットは STM32L16x デバイスでのみ使用可能です。

ビット 26 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 25 **DMA2RST** : DMA2 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 : DMA2 をリセット

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 24 **DMA1RST** : DMA1 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 : DMA1 をリセット

ビット 23:16 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 **FLITFRST** : FLITF 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。フラッシュメモリがパワーダウンモードのときにのみ、FLITF リセットを有効化することができます。

0 : 影響無し

1 : FLITF をリセット

ビット 14:13 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 **CRCRST** : CRC 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 : CRC をリセット



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 **GPIOGRST** : IO ポート G 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 : IO ポート G をリセットします。

注 : このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 6 **GPIOFRST** : IO ポート F 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 : IO ポート F をリセットします。

注 : このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 5 **GPIOHRST** : IO 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 : リセット

ビット 4 **GPIOERST** : IO ポート E 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 : IO ポート E をリセットします。

ビット 3 **GPIODRST** : IO 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 : IO ポート D をリセットします。

ビット 2 **GPIOCRST** : IO 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 : IO ポート C をリセットします。

ビット 1 **GPIOBRST** : IO 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 : IO ポート B をリセットします。

ビット 0 **GPIOARST** : IO 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 : IO ポート A をリセットします。

6.3.6 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 : 0x14

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	USART1 RST	予約済 み	SPI1 RST	SDIO RST	予約済み	ADC1 RST	予約済み				TIM11 RST	TIM10 RST	TIM9 RST	予約済 み	SYSCFG RST
	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 **USART1RST** : 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、リセット値に保持する必要があります。

ビット 12 **SPI1RST** : 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 **SDIORST** : SDIO 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 : SDIO をリセット

注 : このビットは、Cat.4 デバイスでのみ使用可能です。

ビット 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 **ADC1RST** : ADC1 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 : ADC1 インタフェースをリセットします。

ビット 8:5 予約済みであり、リセット値に保持する必要があります。

ビット 4 **TIM11RST** : TIM11 タイマリセット

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 : TIM11 タイマをリセットします。

ビット 3 **TIM10RST** : TIM10 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 : TIM10 タイマをリセットします。

ビット 2 **TIM9RST** : TIM9 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 : TIM9 タイマをリセットします。

ビット 1 予約済みであり、リセット値に保持する必要があります。

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

6.3.7 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 : 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
COMP RST	予約済み	DAC RST	PWR RST	予約済み				USB RST	I2C2 RST	I2C1 RST	UART5 RST	UART4 RST	USART3 RST	USART2 RST	予約済み
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
SPI3 RST	SPI2 RST	予約済み		WWDG RST	予約済み	LCD RST	予約済み			TIM7 RST	TIM6 RST	TIM5 RST	TIM4 RST	TIM3 RST	TIM2 RST
rw	rw			rw		rw				rw	rw	rw	rw	rw	rw

ビット 31 **COMPRST** : COMP 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 : COMP インタフェースをリセット

ビット 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 **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 インタフェースをリセットします。

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 27:24 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 **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** : I²C 2 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 : I²C 2 をリセット

ビット 21 I2C1RST : I²C 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 : I²C 1 をリセット

ビット 20 UART5RST : UART 5 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 : UART 5 をリセットします。

注： このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 19 UART4RST : UART 4 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 : UART 4 をリセットします。

注： このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 18 USART3RST : USART 3 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 : USART 3 をリセットします。

ビット 17 USART2RST : USART 2 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 : USART 2 をリセットします。

ビット 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 SPI3RST : SPI 3 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 3 をリセットします。

注： このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 14 SPI2RST : SPI 2 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 2 をリセットします。

ビット 13:12 予約済みであり、リセット値に保持する必要があります。

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、リセット値に保持する必要があります。

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

ビット 8:6 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 TIM7RST : タイマ 7 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 : タイマ 7 をリセットします。



ビット 4 **TIM6RST** : タイマ 6 リセット

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 : タイマ 6 をリセットします。

ビット 3 **TIM5RST** : タイマ 5 リセット

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 : タイマ 5 をリセットします。

注： このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 2 **TIM4RST** : タイマ 4 リセット

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 : タイマ 4 をリセットします。

ビット 1 **TIM3RST** : タイマ 3 リセット

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 : タイマ 3 をリセットします。

ビット 0 **TIM2RST** : タイマ 2 リセット

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 : タイマ 2 をリセットします。

6.3.8 AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHBENR)

アドレスオフセット : 0x1C

リセット値 : 0x0000 8000

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

注 : ペリフェラルクロックが有効でないときには、ペリフェラルレジスタの値をソフトウェアによって読み出すことはできず、返される値は常に 0x0 です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み	FSMCE N	予約済み		AES EN	予約済み	DMA2E N	DMA1EN	予約済み							
	rw			rw		rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLITF EN	予約済み		CRCEN	予約済み				GPIOG EN	GPIOFE N	GPIOHE N	GPIOEE N	GPIODE N	GPIOCE N	GPIOBE N	GPIOAE N
rw			rw					rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **FSMCEN** : FSMC 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。

0 : FSMC クロック無効

1 : FSMC クロック有効

注 : このビットは、Cat.4 デバイスでのみ使用可能です。

ビット 29:28 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 **AESEN** : AES 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。

0 : AES クロックを無効化

1 : AES クロックを有効化

注 : このビットは STM32L16x デバイスでのみ使用可能です。

ビット 26 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 25 **DMA2EN** : DMA2 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。

0 : DMA2 クロック無効

1 : DMA2 クロック有効

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 24 **DMA1EN** : DMA1 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。

0 : DMA1 クロック無効

1 : DMA1 クロック有効

ビット 23:16 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 **FLITFEN** : FLITF クロックイネーブル

フラッシュメモリがパワーダウンモードのときにのみ、このビットを書き込むことができます。

0 : FLITF クロックを無効化

1 : FLITF クロックを有効化

ビット 14:13 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 **CRCCEN** : CRC 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。

0 : CRC クロック無効

1 : CRC クロック有効

ビット 11:6 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 7 **GPIOGEN** : IO ポート G 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。

0 : IO ポート G クロック無効

1 : IO ポート G クロック有効

注 : このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 6 **GPIOFEN** : IO ポート F 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。

0 : IO ポート F クロック無効

1 : IO ポート F クロック有効

注 : このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 5 **GPIOHEN** : IO ポート H 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。

0 : IO ポート H クロックを無効化

1 : IO ポート H クロックを有効化

ビット 4 **GPIOEEN** : IO ポート E 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。

0 : IO ポート E クロック無効

1 : IO ポート E クロック有効

ビット 3 **GPIODEN** : IO ポート D クロックイネーブル

ソフトウェアによってセット/クリアされます。

0 : IO ポート D クロック無効

1 : IO ポート D クロック有効

ビット 2 **GPIOCEN** : IO ポート C 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。

0 : IO ポート C クロック無効

1 : IO ポート C クロック有効

ビット 1 **GPIOBEN** : IO ポート B 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。

0 : IO ポート B クロック無効

1 : IO ポート B クロック有効

ビット 0 **GPIOAEN** : IO ポート A 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。

0 : IO ポート A クロック無効

1 : IO ポート A クロック有効

6.3.9 APB2 ペリフェラルクロックイネーブルレジスタ (RCC_APB2ENR)

アドレス : 0x20

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

ノーウェイトステート。ただし、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した場合を除きます。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ます。

注 : ペリフェラルクロックが有効でないときには、ペリフェラルレジスタの値をソフトウェアによって読み出すことはできず、返される値は常に 0x0 です。

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	USART1E N	予約済み	SPI1E N	SDIO EN	予約済み	ADC1E N	予約済み				TIM11 EN	TIM10 EN	TIM9 EN	予約済み	SYS_CFG EN
	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 **USART1EN** : USART1 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。

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、リセット値に保持する必要があります。

ビット 12 **SPI1EN** : SPI 1 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。

0 : SPI 1 クロック無効

1 : SPI 1 クロック有効

ビット 11 **SDIOEN** : SDIO 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。

0 : SDIO クロック無効

1 : SDIO クロック有効

注 : このビットは、Cat.4 デバイスでのみ使用可能です。

ビット 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 **ADC1EN** : ADC1 インタフェース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。

0 : ADC1 インタフェースを無効化

1 : ADC1 インタフェースクロックを有効化

ビット 8:5 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 **TIM11EN** : TIM11 タイマ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。

0 : TIM11 タイマクロック無効

1 : TIM11 タイマクロック有効

ビット 3 **TIM10EN** : TIM10 タイマ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。

0 : TIM10 タイマクロック無効

1 : TIM10 タイマクロック有効



ビット 2 **TIM9EN** : TIM9 タイマ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。

0 : TIM9 タイマクロック無効

1 : TIM9 タイマクロック有効

ビット 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 **SYSCFGEN** : システム設定コントローラ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。

0 : システム設定コントローラクロックを無効化

1 : システム設定コントローラクロックを有効化

6.3.10 APB1 ペリフェラルクロックイネーブルレジスタ (RCC_APB1ENR)

アドレス : 0x24

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

ノーウェイトステート。ただし、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した場合を除きます。この場合、APB1 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ます。

注 : ペリフェラルクロックが有効でないときには、ペリフェラルレジスタの値をソフトウェアによって読み出すことはできず、返される値は常に 0x0 です。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
COMPEN	予約済み	DACEN	PWREN	予約済み				USBEN	I2C2EN	I2C1EN	UART5EN	UART4EN	USART3EN	USART2EN	予約済み
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
SPI3EN	SPI2EN	予約済み		WWWDGEN	予約済み	LCDEN	予約済み			TIM7EN	TIM6EN	TIM5EN	TIM4EN	TIM3EN	TIM2EN
rw	rw			rw		rw				rw	rw	rw	rw	rw	rw

ビット 31 **COMPEN** : COMP インタフェース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。
 0 : COMP インタフェースクロックを無効化
 1 : COMP インタフェースクロックを有効化

ビット 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 **DACEN** : DAC インタフェース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。
 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
 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 27:24 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 **USBEN** : USB 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。
 0 : USB クロック無効
 1 : USB クロック有効

ビット 22 **I2C2EN** : I²C 2 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。
 0 : I²C 2 クロックを無効化
 1 : I²C 2 クロックを有効化

ビット 21 **I2C1EN** : I²C 1 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。
 0 : I²C 1 クロックを無効化
 1 : I²C 1 クロックを有効化

ビット 20 **UART5EN** : UART 5 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。

0 : UART 5 クロック無効

1 : UART 5 クロック有効

注: このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 19 **UART4EN** : UART 4 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。

注: 0 : UART 4 クロック無効

1 : UART 4 クロック有効

このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 18 **USART3EN** : USART 3 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。

0 : USART 3 クロック無効

1 : USART 3 クロック有効

ビット 17 **USART2EN** : USART 2 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。

0 : USART 2 クロック無効

1 : USART 2 クロック有効

ビット 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 **SPI3EN** : SPI 3 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。

0 : SPI 3 クロック無効

1 : SPI 3 クロック有効

注: このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 14 **SPI2EN** : SPI 2 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。

0 : SPI 2 クロック無効

1 : SPI 2 クロック有効

ビット 13:12 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 **WWDGEN** : ウィンドウ型ウォッチドッグ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。

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、リセット値に保持する必要があります。

ビット 9 **LCDEN** : LCD 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。

0 : LCD クロックを無効化

1 : LCD クロックを有効化

ビット 8:6 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 **TIM7EN** : タイマ 7 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。

0 : タイマ 7 クロック無効

1 : タイマ 7 クロック有効

ビット 4 **TIM6EN** : タイマ 6 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。

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 **TIM5EN** : タイマ 5 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。
 0 : タイマ 5 クロック無効
 1 : タイマ 5 クロック有効
注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。
- ビット 2 **TIM4EN** : タイマ 4 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。
 0 : タイマ 4 クロック無効
 1 : タイマ 4 クロック有効
- ビット 1 **TIM3EN** : タイマ 3 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。
 0 : タイマ 3 クロック無効
 1 : タイマ 3 クロック有効
- ビット 0 **TIM2EN** : タイマ 2 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。
 0 : タイマ 2 クロック無効
 1 : タイマ 2 クロック有効

6.3.11 低消費電力モードレジスタでの AHB ペリフェラルクロックイネーブル (RCC_AHBLPENR)

アドレスオフセット : 0x28

リセット値 : 0x0101 903F

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

注 : ペリフェラルクロックは、AHBENR レジスタで予め有効化されている場合にのみ、sleep モードで有効化することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み	FSMC LPEN	予約済み		AES LPEN	予約済み	DMA2 LPEN	DMA1 LPEN	予約済み							SRAM LPEN
	rw			rw		rw	rw								rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLITF LPEN	予約済み		CRC LPEN	予約済み				GPIOGL PEN	GPIOFL PEN	GPIOHL PEN	GPIOEL PEN	GPIODL PEN	GPIOCL PEN	GPIOBL PEN	GPIOA LPEN
rw			rw					rw	rw	rw	rw	rw	rw	rw	rw

- ビット 31 予約済みであり、リセット値に保持する必要があります。
- ビット 30 **FSMCLPEN** : FSMC クロックイネーブル — Sleep モード時
 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 : FSMC クロックを無効化 — Sleep モード時
 1 : FSMC クロックを有効化 — Sleep モード時
注 : このビットは、Cat.4 デバイスでのみ使用可能です。
- ビット 29:28 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 **AESLPEN** : AES クロックイネーブル — Sleep モード時
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 : AES クロックを無効化 — Sleep モード時
1 : AES クロックを有効化 — Sleep モード時

注： このビットは STM32L16x デバイスでのみ使用可能です。

ビット 26 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 25 **DMA2LPEN** : DMA2 クロックイネーブル — Sleep モード時
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 : DMA2 クロックを無効化 — Sleep モード時
1 : DMA2 クロックを有効化 — Sleep モード時

注： このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 24 **DMA1LPEN** : DMA1 クロックイネーブル — Sleep モード時
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 : DMA1 クロックを無効化 — Sleep モード時
1 : DMA1 クロックを有効化 — Sleep モード時

ビット 23:17 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 **SRAMLPEN** : SRAM クロックイネーブル — Sleep モード時
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 : SRAM クロックを無効化 — Sleep モード時
1 : SRAM クロックを有効化 — Sleep モード時

ビット 15 **FLITFLPEN** : FLITF クロックイネーブル — Sleep モード時
フラッシュメモリがパワーダウンモードのときにのみ、このビットに書き込むことができます。
0 : FLITF クロックを無効化 — Sleep モード時
1 : FLITF クロックを有効化 — Sleep モード時

ビット 14:13 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 **CRCLPEN** : CRC クロックイネーブル — Sleep モード時
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 : CRC クロックを無効化 — Sleep モード時
1 : CRC クロックを有効化 — Sleep モード時

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 **GPIOGLPEN** : IO ポート G クロックイネーブル — Sleep モード時
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 : IO ポート G クロックを無効化 — Sleep モード時
1 : IO ポート G クロックを有効化 — Sleep モード時

注： このビットは、Cat.4 と Cat.5 デバイスでのみ使用可能です。

ビット 6 **GPIOFLPEN** : IO ポート F クロックイネーブル — Sleep モード時
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 : IO ポート F クロックを無効化 — Sleep モード時
1 : IO ポート F クロックを有効化 — Sleep モード時

注： このビットは、Cat.4 と Cat.5 デバイスでのみ使用可能です。

ビット 5 **GPIOHLPEN** : IO ポート H クロックイネーブル — Sleep モード時
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 : IO ポート H クロックを無効化 — Sleep モード時
1 : IO ポート H クロックを有効化 — Sleep モード時

- ビット 4 **GPIOLPEN** : IO ポート E クロックイネーブル — Sleep モード時
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 : IO ポート E クロックを無効化 — Sleep モード時
1 : IO ポート E クロックを有効化 — Sleep モード時
- ビット 3 **GPIODLPEN** : IO ポート D クロックイネーブル — Sleep モード時
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 : IO ポート D クロックを無効化 — Sleep モード時
1 : IO ポート D クロックを有効化 — Sleep モード時
- ビット 2 **GPIOCLPEN** : IO ポート C クロックイネーブル — Sleep モード時
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 : IO ポート C クロックを無効化 — Sleep モード時
1 : IO ポート C クロックを有効化 — Sleep モード時
- ビット 1 **GPIOBLPEN** : IO ポート B クロックイネーブル — Sleep モード時
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 : IO ポート B クロックを無効化 — Sleep モード時
1 : IO ポート B クロックを有効化 — Sleep モード時
- ビット 0 **GPIOTALPEN** : IO ポート A クロックイネーブル — Sleep モード時
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 : IO ポート A クロックを無効化 — Sleep モード時
1 : IO ポート A クロックを有効化 — Sleep モード時

6.3.12 低消費電力モードレジスタでの APB2 ペリフェラルクロックイネーブル (RCC_APB2LPENR)

アドレス : 0x2C
リセット値 : 0x0000 521D
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

注 : ペリフェラルクロックは、**APB2ENR** レジスタで予め有効化されている場合にのみ、**sleep モード**で有効化することができます。

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	USART1 LPEN	予約済み	SPI1 LPEN	SDIO LPEN	予約済み	ADC1 LPEN	予約済み				TIM11 LPEN	TIM10 LPEN	TIM9 LPEN	予約済み	SYSCFG LPEN
	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 **USART1LPEN** : USART1 クロックイネーブル — Sleep モード時
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 クロックを無効化 — Sleep モード時
1 : USART1 クロックを有効化 — Sleep モード時
- ビット 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 **SPI1LPEN** : SPI 1 クロックイネーブル — Sleep モード時
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 : SPI 1 クロックを無効化 — Sleep モード時
1 : SPI 1 クロックを有効化 — Sleep モード時

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。

ビット 11 **SDIOLPEN** : SDIO クロックイネーブル — Sleep モード時

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 : SDIO クロックを無効化 — Sleep モード時

1 : SDIO クロックを有効化 — Sleep モード時

注 : このビットは、Cat.4 デバイスでのみ使用可能です。

ビット 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 **ADC1LPEN** : ADC1 インタフェースクロックイネーブル — Sleep モード時

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 : ADC1 インタフェースを無効化 — Sleep モード時

1 : ADC1 インタフェースクロックを有効化 — Sleep モード時

ビット 8:5 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 **TIM11LPEN** : TIM11 タイマクロックイネーブル — Sleep モード時

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 : TIM11 タイマクロックを無効化 — Sleep モード時

1 : TIM11 タイマクロックを有効化 — Sleep モード時

ビット 3 **TIM10LPEN** : TIM10 タイマクロックイネーブル — Sleep モード時

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 : TIM10 タイマクロックを無効化 — Sleep モード時

1 : TIM10 タイマクロックを有効化 — Sleep モード時

ビット 2 **TIM9LPEN** : TIM9 タイマクロックイネーブル — Sleep モード時

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 : TIM9 タイマクロックを無効化 — Sleep モード時

1 : TIM9 タイマクロックを有効化 — Sleep モード時

ビット 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 **SYSCFGLPEN** : システム設定コントローラクロックイネーブル — Sleep モード時

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を無効化 — Sleep モード時

1 : システム設定コントローラクロックを有効化 — Sleep モード時

6.3.13 低消費電力モードレジスタでの APB1 ペリフェラルクロックイネーブル (RCC_APB1LPENR)

アドレス : 0x30

リセット値 : 0xB0E6 4A37

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

注 : ペリフェラルクロックは、APB1ENR レジスタで予め有効化されている場合にのみ、sleep モードで有効化することができます。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
COMPLPEN	予約済み	DACL PEN	PWR LPEN	予約済み				USB LPEN	I2C2 LPEN	I2C1 LPEN	UART5 LPEN	UART4 LPEN	USART3 LPEN	USART2 LPEN	予約済み
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
SPI3 LPEN	SPI2 LPEN	予約済み		WWDG LPEN	予約済み	LCD LPEN	予約済み			TIM7 LPEN	TIM6 LPEN	TIM5 LPEN	TIM4 LPEN	TIM3 LPEN	TIM2 LPEN
rw	rw			rw		rw				rw	rw	rw	rw	rw	rw

ビット 31 **COMPLPEN** : COMP インタフェースクロックイネーブル — Sleep モード時

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 : COMP インタフェースクロックを無効化 — Sleep モード時
1 : COMP インタフェースクロックを有効化 — Sleep モード時

ビット 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 **DACL PEN** : DAC インタフェースクロックイネーブル — Sleep モード時

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を無効化 — Sleep モード時
1 : DAC インタフェースクロックを有効化 — Sleep モード時

ビット 28 **PWRLPEN** : 電源インタフェースクロックイネーブル — Sleep モード時

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を無効化 — Sleep モード時
1 : 電源インタフェースクロックを有効化 — Sleep モード時

ビット 27:24 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 **USBLPEN** : USB クロックイネーブル — Sleep モード時

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 クロックを無効化 — Sleep モード時
1 : USB クロックを有効化 — Sleep モード時

ビット 22 **I2C2LPEN** : I²C 2 クロックイネーブル — Sleep モード時

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 : I²C 2 クロックを無効化 — Sleep モード時
1 : I²C 2 クロックを有効化 — Sleep モード時

ビット 21 **I2C1LPEN** : I²C 1 クロックイネーブル — Sleep モード時

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 : I²C 1 クロックを無効化 — Sleep モード時
1 : I²C 1 クロックを有効化 — Sleep モード時



ビット 20 **UART5LPEN** : USART 5 クロックイネーブル — Sleep モード時

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 5 クロックを無効化 — Sleep モード時

1 : USART 5 クロックを有効化 — Sleep モード時

注: このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 19 **UART4LPEN** : USART 4 クロックイネーブル — Sleep モード時

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 4 クロックを無効化 — Sleep モード時

1 : USART 4 クロックを有効化 — Sleep モード時

注: このビットは、Cat.4とCat.5 デバイスでのみ使用可能です。

ビット 18 **USART3LPEN** : USART 3 クロックイネーブル — Sleep モード時

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 3 クロックを無効化 — Sleep モード時

1 : USART 3 クロックを有効化 — Sleep モード時

ビット 17 **USART2LPEN** : USART 2 クロックイネーブル — Sleep モード時

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 2 クロックを無効化 — Sleep モード時

1 : USART 2 クロックを有効化 — Sleep モード時

ビット 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 **SPI3LPEN** : SPI 3 クロックイネーブル — Sleep モード時

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 : SPI 3 クロックを無効化 — Sleep モード時

1 : SPI 3 クロックを有効化 — Sleep モード時

注: このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 14 **SPI2LPEN** : SPI 2 クロックイネーブル — Sleep モード時

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 : SPI 2 クロックを無効化 — Sleep モード時

1 : SPI 2 クロックを有効化 — Sleep モード時

ビット 13:12 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 **WWDGLPEN** : ウィンドウ型ウォッチドッグクロックイネーブル — Sleep モード時

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を無効化 — Sleep モード時

1 : ウィンドウ型ウォッチドッグクロックを有効化 — Sleep モード時

ビット 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 **LCDLPEN** : LCD クロックイネーブル — Sleep モード時

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 クロックを無効化 — Sleep モード時

1 : LCD クロックを有効化 — Sleep モード時

ビット 8:6 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 **TIM7LPEN** : タイマ 7 クロックイネーブル — Sleep モード時

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 : タイマ 7 クロックを無効化 — Sleep モード時

1 : タイマ 7 クロックを有効化 — Sleep モード時

ビット 4 **TIM6LPEN** : タイマ 6 クロックイネーブル — Sleep モード時
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 : タイマ 6 クロックを無効化 — Sleep モード時
1 : タイマ 6 クロックを有効化 — Sleep モード時

ビット 3 **TIM5LPEN** : タイマ 5 クロックイネーブル — Sleep モード時
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 : タイマ 5 クロックを無効化 — Sleep モード時
1 : タイマ 5 クロックを有効化 — Sleep モード時

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 2 **TIM4LPEN** : タイマ 4 クロックイネーブル — Sleep モード時
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 : タイマ 4 クロックを無効化 — Sleep モード時
1 : タイマ 4 クロックを有効化 — Sleep モード時

ビット 1 **TIM3LPEN** : タイマ 3 クロックイネーブル — Sleep モード時
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 : タイマ 3 クロックを無効化 — Sleep モード時
1 : タイマ 3 クロックを有効化 — Sleep モード時

ビット 0 **TIM2LPEN** : タイマ 2 クロックイネーブル — Sleep モード時
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 : タイマ 2 クロックを無効化 — Sleep モード時
1 : タイマ 2 クロックを有効化 — Sleep モード時

6.3.14 制御/ステータスレジスタ (RCC_CSR)

アドレス : 0x34

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/ステータスレジスタ (RCC_CSR) の LSEON、LSEBYP、RTCSEL、RTCEN ビットは RTC ドメインに属します。このため、リセット後これらのビットは書き込み禁止状態にあり、ビットの値を変更するには、その前に電源制御レジスタ (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、RTC レジスタと RTC バックアップレジスタのセクション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 (RTC レジスタとバックアップレジスタのリセット参照)。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	RTC RST	RTC EN	予約済み				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
予約済み			LSECSS D	LSECSS ON	LSE BYP	LSE RDY	LSEON	予約済み						LSI RDY	LSION
			r	rw	rw	r	rw							r	rw



ビット 31 LPWRRSTF : 低消費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、ハードウェアからセットされます。
RMVF ビットへの書き込み、または POR により、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、[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 ビットへの書き込み、または POR により、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 ビットへの書き込み、または POR により、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、ハードウェアによってセットされます。

RMVF ビットへの書き込み、または POR により、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 ビットへの書き込み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 ビットへの書き込み、または POR により、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 ビットへの書き込み、または POR により、クリアされます。

0 : OBL リセット発生していません。

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

このビットはソフトウェアからセットされて、リセットフラグをクリアします。

0 : 影響無し

1 : リセットフラグをクリアします。

ビット 23 RTCRST : RTC ソフトウェアリセット

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 : リセットは有効ではありません。

1 : RTC ペリフェラルと、そのクロックソース選択、バックアップレジスタをリセット

ビット 22 **RTCEN** : RTC 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。

このビットは、RTCRST ビットのセットまたは POR によりリセットされます。

0 : RTC クロック無効

1 : RTC クロック有効

ビット 21:18 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 17:16 **RTCSEL[1:0]** : RTC および LCD クロックソース選択

これらのビットはソフトウェアからセットされ、RTC のクロックソースを選択します。

RTC と LCD のクロックソースを選択した後は、RTCRST がセットされるか、または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、これが CSS から検出された場合で、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を RTC/LCD クロックとして使用

10 : LSI オシレータクロックを RTC/LCD クロックとして使用

11 : プログラマブルプリスケアラ (RCC クロック制御レジスタ (RCC_CR) の RTCPRE[1:0] ビットで選択) で分周した HSE オシレータクロックを RTC/LCD クロックとして使用

LSE または LSI を RTC クロックソースとして使用する場合、RTC は Stop 低消費電力モードと Standby 低消費電力モードで動作を続けるため、ウェイクアップソースとして使用することができます。ただし、HSE が RTC クロックソースの場合、RTC は Stop 低消費電力モードと Standby 低消費電力モードで使用できません。

ビット 15:13 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 **LSECSSD** : LSE 故障検出時の CSS

ハードウェアからセットされ、外部 32 kHz オシレータ (LSE) でクロックセキュリティシステムにより故障が検出されたタイミングを示します。

パワーオンリセットと RTC ソフトウェアリセット (RTCRST ビット) によりリセットされます。

0 : LSE (32 kHz オシレータ) で故障非検出

1 : LSE (32 kHz オシレータ) で故障検出

注 : このビットは、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 11 **LSECSSON** LSE 上での CSS イネーブル

ソフトウェアからセットされ、LSE (32 kHz オシレータ) でクロックセキュリティシステムを有効化します。

LSECSSON は、LSE と LSI オシレータを有効化し (LSEON ビットと LSION ビットを有効化)、レディになり (ハードウェアから LSERDY フラグと LSIRDY フラグをセット)、RTCSEL ビットを選択した後に有効化する必要があります。

このビットを有効化した後、無効化することはできません。ただし、LSE 故障検出時 (LSECSSD =1) は除きます。この場合、ソフトウェアから LSECSSON ビットを無効化する必要があります。

パワーオンリセットと RTC ソフトウェアリセット (RTCRST ビット) によりリセットされます。

0 : LSE (32 kHz オシレータ) で CSS が OFF

1 : LSE (32 kHz オシレータ) で CSS が ON

注 : このビットは、Cat.2、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 10 **LSEBYP** : 外部低速オシレータバイパス

このビットはソフトウェアセット/クリアされて、デバッグモードでオシレータをバイパスします。このビットは、LSE 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、RTCRST ビットのセットまたは POR によりリセットされます。

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

このビットはハードウェアからセット/クリアされて、LSE オシレータが安定したタイミングを示します。LSEON ビットのクリア後、LSE オシレータクロックの 6 サイクル後に LSERDY はローになります。

このビットは、RTCRST ビットのセットまたは POR によりリセットされます。

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

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、RTCRST ビットのセットまたは POR によりリセットされます。

0 : LSE オシレータ OFF

1 : LSE オシレータ ON

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 LSIRDY : 内部低速オシレータレディ

このビットはハードウェアからセット/クリアされ、LSI オシレータが安定したタイミングを示します。LSION ビットのクリア後、LSE オシレータクロックの 3 サイクル後に LSIRDY はローになります。

このビットは、システムリセットでリセットされます。

0 : LSI オシレータが非レディ

1 : LSI オシレータがレディ

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。

このビットは、システムリセットでリセットされます。

0 : LSI オシレータが OFF

1 : LSI オシレータが ON

6.3.15 RCC レジスタマップ

次の表に、RCC レジスタマップおよびリセット値を示します。

表 35. 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	予約済み	RTCPRE1	RTCPRE0	CSSON	予約済み	PLL RDY	PLL ON	予約済み	予約済み	HSEBYP	HSERDY	HSEON	予約済み	MSIRDY	MSION	予約済み	HSIRDY	HSION																							
	リセット値	0	0	0	0															0	0	0	x	0	0	1	1	0	0													
0x04	RCC_ICSCR	MSITRIM[7:0]																		MSICAL[7:0]								MSIRANGE[2:0]		HSITRIM[4:0]				HSICAL[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
0x08	RCC_CFGR	予約済み	MCOPRE[2:0]		予約済み	MCOSEL[2:0]		PLL DIV[1:0]		PLLMUL[3:0]			予約済み	PLLSRC	予約済み	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	0	0								
0x0C	RCC_CIR	予約済み								CSSC	LSECSSC	MSIRDYC	PLLRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC	予約済み	LSECSSIE	MSIRDYIE	PLLRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE	CSSF	予約済み	MSIRDYF	PLLRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF									
	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		
0x10	RCC_AHBSTR	予約済み	FSMCRST	予約済み	AESRST	予約済み	DMA2RST	DMA1RST	予約済み							FLITRST	予約済み	CRCRST	予約済み						GPIORST	GPIORST	GPIORST	GPIORST	GPIORST	GPIORST	GPIORST	GPIORST	GPIORST	GPIORST								
	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
0x14	RCC_APB2RSTR	予約済み														USART1RST	予約済み	SPI1RST	SDIORST	予約済み	ADC1RST	予約済み						TM11RST	TM10RST	TM9RST	予約済み	SYSCFGRST										
	リセット値															0		0	0		0							0	0	0		0	0	0	0	0	0	0	0	0	0	0
0x18	RCC_APB1RSTR	COMPRST	予約済み	DACRST	PWRST	予約済み				USBRST	I2C2RST	I2C1RST	UART5RST	UART4RST	USART3RST	USART2RST	予約済み	SPI3RST	SPI2RST	予約済み		WWDRST	予約済み	LCDRST	予約済み		TM7RST	TM6RST	TM5RST	TM4RST	TM3RST	TM2RST										
	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
0x1C	RCC_AHBENR	予約済み	FSMCEN	予約済み	AESEN	予約済み	DMA2EN	DMA1EN	予約済み							FLITEN	予約済み	CRCEEN	予約済み						GPIOPGEN	GPIOPGEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN	GPIOPHEN							
	リセット値	0	0				0	0								0									0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	RCC_APB2ENR	予約済み														USART1EN	予約済み	SPI1EN	SDIOEN	予約済み	ADC1EN	予約済み						TM11EN	TM10EN	TM9EN	予約済み	SYSCFGEN										
	リセット値															0		0	0		0							0	0	0		0	0	0	0	0	0	0	0	0	0	0
0x24	RCC_APB1ENR	COMPEN	予約済み	DACEN	PWREN	予約済み				USBEN	I2C2EN	I2C1EN	USART5EN	USART4EN	USART3EN	USART2EN	予約済み	SPI3EN	SPI2EN	予約済み		WWDGEN	予約済み	LCDEN	予約済み		TM7EN	TM6EN	TM5EN	TM4EN	TM3EN	TM2EN										
	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

表 35. 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					
0x28	RCC_AHB1P ENR	予約済み	FSMC1PEN	予約済み		AES1PEN	予約済み	DMA21PEN	DMA11PEN	予約済み							SRAM1PEN	FLIT1PEN	予約済み		CRCLPEN	予約済み				GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN	GPIOD1PEN			
	リセット値		1			1		1	1								1	1			1					1	1	1	1	1	1	1	1	1	1	1		
0x2C	RCC_APB21P ENR	予約済み																		USART11PEN	予約済み		SP11PEN	SDIOLPEN	予約済み		ADC11PEN	予約済み				TIM111PEN	TIM101PEN	TIM91PEN	予約済み		SYSOCFGLPEN	
	リセット値																			1			1	1			1					1	1	1			1	
0x30	RCC_APB11P ENR	COM1PEN	予約済み		DACL1PEN	PWR1PEN	予約済み					USBL1PEN	I2C21PEN	I2C11PEN	USART51PEN	USART41PEN	USART31PEN	USART21PEN	予約済み		SP131PEN	SP121PEN	予約済み		VWDG1PEN	予約済み		LDOLPEN	予約済み				TIM71PEN	TIM61PEN	TIM51PEN	TIM41PEN	TIM31PEN	TIM21PEN
	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
0x034	RCC_CSR	LPWRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	PORRSTF	PINRSTF	OBLRSTF	RMVF	RTCRST	RTCEN	予約済み					RTCSEL [1:0]	予約済み				LSECSSD	LSECSSON	LSEBYP	LSERDY	LSEON	予約済み					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				

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

7 汎用 I/O (GPIO)

7.1 GPIO の概要

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)、32 ビットセット/リセットレジスタ (GPIOx_BSRR)、32 ビットロックレジスタ (GPIOx_LCKR)、2 つの 32 ビットオルタネート機能選択レジスタ (GPIOx_AFRH、GPIOx_AFLR) があります。

7.2 GPIO の主な機能

- 最大 16 個の I/O を制御
- 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 への速度選択
- 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) からの入力データ
- GPIOx_ODR へのビット単位の書き込みアクセスのためのビットのセット/リセットレジスタ (GPIOx_BSRR)
- I/O 設定を凍結可能にするロック機構 (GPIOx_LCKR)
- アナログ機能
- オルタネート機能入力/出力選択レジスタ (入出力ごとに最大 16 のオルタネート機能)
- 2 クロックサイクル毎に変更可能な高速反転
- I/O ピンを GPIO またはいくつかのペリフェラル機能の 1 つとして使用することができる柔軟性の高いピンのマルチプレクス

7.3 GPIO 機能の説明

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複数のモードで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
- 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付き)
- 出力プッシュプル (プルアップまたはプルダウン機能付き)
- 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付き)
- オルタネート機能オープンドレイン (プルアップまたはプルダウン機能付き)

各 I/O ポートビットは自由にプログラミングできますが、I/O ポートレジスタには 32 ビットワード単位、ハーフワード単位、またはバイト単位でアクセスする必要があります。GPIOx_BSRR レジスタを使用すると、GPIO レジスタにアトミックな読み出し/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 18 および 図 19 は、それぞれ標準および 5 V 耐性 I/O ポートビットの基本構造を示しています。
表 39 には、実行可能なポートビットの設定を示します。

図 18. 標準 I/O ポートビットの基本構造

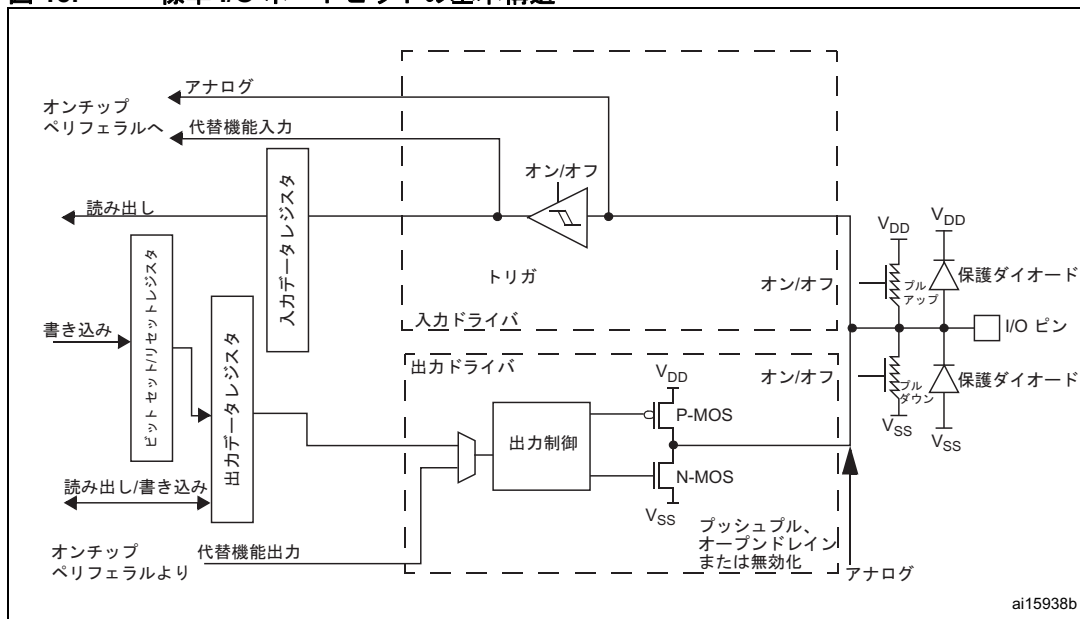
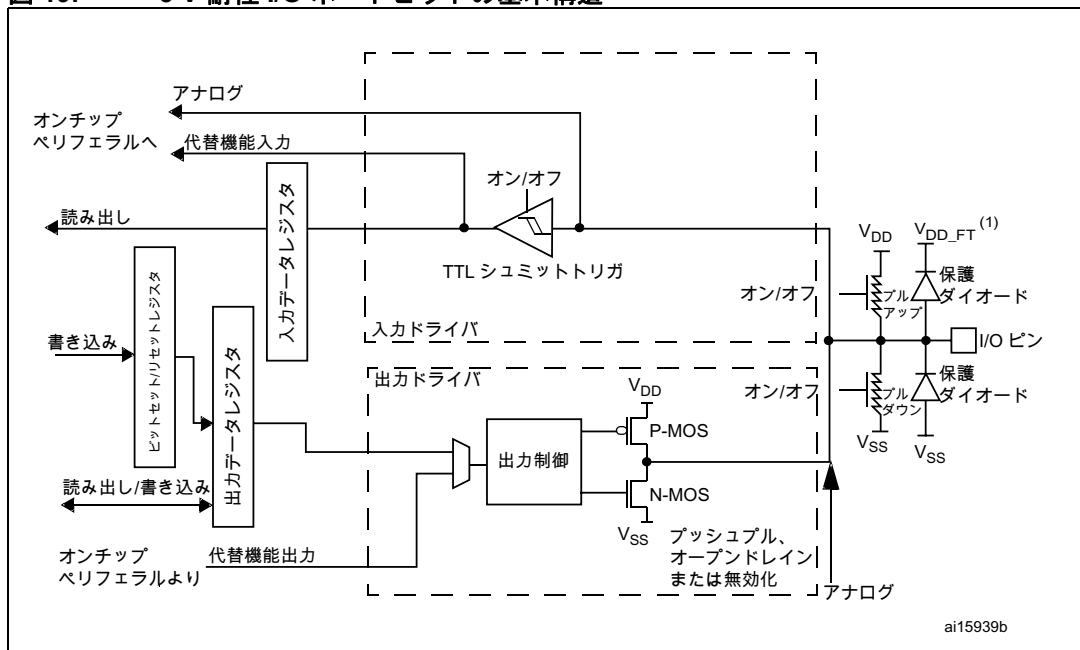


図 19. 5 V 耐性 I/O ポートビットの基本構造



1. V_{DD_FT} は、5 V 耐性 I/O に固有の電位であり、 V_{DD} とは異なります。

表 36. ポートビット設定表⁽¹⁾

MODER(i) [1:0]	OTYPER(i)	OSPEEDR(i) [B:A]		PUPDR(i) [1:0]		I/O 設定	
01	0	SPEED [B:A]		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 [B:A]		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

7.3.1 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アクティブではなく、I/O ポートは入力フローティングモードに設定されています。

リセット後、デバッグモードピンは AF プルアップ／プルダウンになっています。

- PA15 : JTDI プルアップ
- PA14 : JTCK/SWCLK プルダウン
- PA13 : JTMS/SWDAT プルダウン
- PB4 : NJTRST プルアップ
- PB3 : JTDO フローティング状態

ピンが出力として設定されると、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。出力ドライバは、プッシュプルモードまたはオープンドレインモードで使用できます (0 を出力すると、N-MOS のみが有効になります)。

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 ピンには、GPIOx_PUPDR レジスタの値によって有効／無効にできる内部ウィークプルアップおよびウィークプルダウン抵抗があります。

7.3.2 I/O ピンマルチプレクサおよびマッピング

マイクロコントローラ I/O ピンは、1 つのペリフェラルのオルタネート機能 (AF) を一度に 1 つの I/O ピンにしか接続できない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の発生が無くなります。

各 I/O ピンには、16 のオルタネート機能入力 (AF0 から AF15) を持つマルチプレクサを装備しており、これらの入力、GPIOx_AFR1 (ピン 0 から 7) と GPIOx_AFR2 (ピン 8 から 15) レジスタを通じて設定することができます。

- リセット後、すべての I/O はシステムのオルタネート機能 0 (AF0) に接続されます。
- ペリフェラルのオルタネート機能は、AF1 から AF14 にマッピングされます。
- Cortex®-M3 EVENTOUT は AF15 にマッピングされます。

この構造を、以下の図 20 に示します。

この柔軟な I/O マルチプレクサアーキテクチャに加え、各ペリフェラルには異なる I/O ピンに割り付けられたオルタネート機能を装備し、より小型のパッケージ内で、使用可能なペリフェラル数を最適化しています。

I/O を希望の設定で使用するには、次の手順に従います。

- **システム機能**
I/O を AF0 に接続し、使用する機能に応じて設定します。
 - JTAG/SWD : 各デバイスのリセット後、このピンはデバッグホストが直ちに利用可能な専用ピンとして割り当てられます (GPIO コントローラには制御されません)
 - RTC_AF1 : このピン設定の詳細は表 38 : RTC_AF1 ピンを参照してください
 - RTC_50Hz : このピンは、入力フローティングモードに設定します
 - MCO : このピンは、オルタネート機能モードに設定する必要があります

注 : JTAG/SWD ピンの一部またはすべてを無効にし、関連したピンを解放して GPIO に使用できます。

詳細については、セクション 6.2.13 : クロック出力機能を参照してください。

表 37. フレキシブルな SWJ-DP ピン配置

使用可能なデバッグポート	SWJ I/O ピンの割り当て				
	PA13/ JTMS/ SWDIO	PA14/ JTCK/ SWCLK	PA15/ JTDI	PB3/ JTDO	PB4/ NJTRST
全 SWJ (JTAG-DP + SW-DP)、リセット状態	X	X	X	X	X
全 SWJ (JTAG-DP + SW-DP)、NJTRST なし	X	X	X	X	
JTAG-DP 無効、SW-DP 有効	X	X			
JTAG-DP 無効、SW-DP 無効					解放

- **GPIO**

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**

ADC および DAC の場合、GPIOx_MODER レジスタで I/O をアナログとして設定します。

その他のペリフェラルの場合：

- 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。
- GPIOx_OTYPER、GPIOx_PUPDR、GPIOx_OSPEEDER レジスタを使って、それぞれタイプ、プルアップ／プルダウン、出力速度を選択します。
- GPIOx_AFR1 または GPIOx_AFRH レジスタで、I/O を AFx に接続します。

- **EVENTOUT**

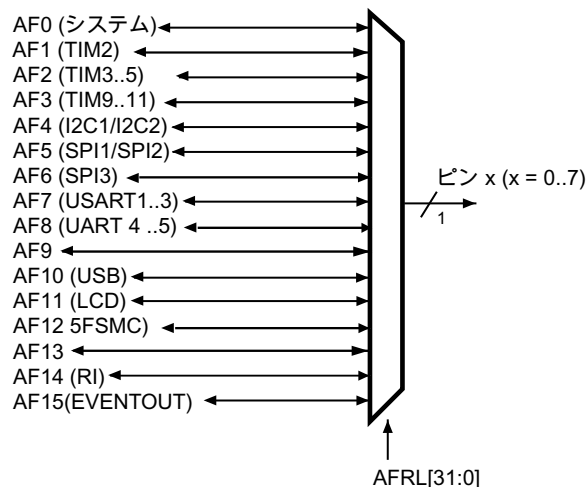
Cortex®-M3 EVENTOUT 信号の出力に使用する I/O ピンを AF15 に接続して設定します。

注： *EVENTOUT は、PH0、PH1、PH2 の I/O ピンには配置されません。*

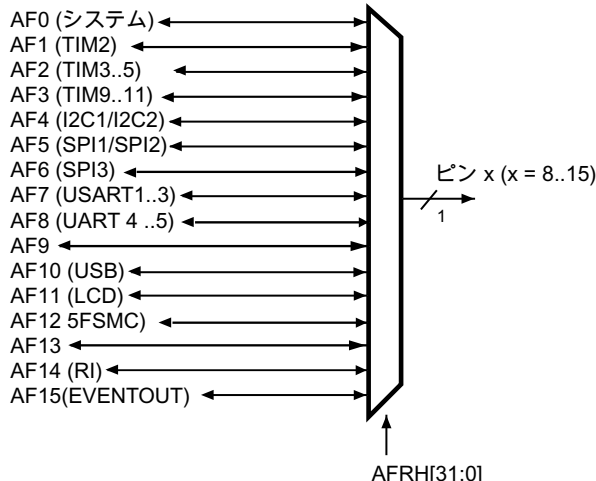
システムおよびペリフェラルのオルタネート機能 I/O ピンの配置詳細については、データシートの「オルタネート機能配置」表を参照してください。

図 20. オルタネート機能の選択

ピン 0 から 7 に対し、GPIOx_AFRL[31:0] レジスタによって専用の代替機能を選択します。



ピン 8 から 15 に対し、GPIOx_AFRH[31:0] レジスタによって専用の代替機能を選択します。



MS18927V1

7.3.3 I/O ポート制御レジスタ

各 GPIO には、最大 16 個の I/O を設定する 32 ビットのメモリマップされた制御レジスタが 4 つ (GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR) が あります。GPIOx_MODER レジスタは、I/O 方向 (入力、出力、AF、アナログ) の選択に使用します。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) と速度の選択に使用します (I/O 速度ピンは、I/O 方向に関わらず、対応する GPIOx_OSPEEDR レジスタビットに直接接続されています)。GPIOx_PUPDR レジスタは、I/O 方向に関わらず、プルアップ/プルダウンの選択に使用します。

7.3.4 I/O ポートデータレジスタ

各 GPIO には、16 ビットのメモリマップされた I/O データレジスタが 2 つ (GPIOx_IDR および GPIOx_ODR) があります。GPIOx_ODR は出力するデータを保存します。読み出し/書き込みが可能

です。I/O から入力されるデータは読み出し専用レジスタである入力データレジスタ (GPIOx_IDR) に保存されます。

レジスタの説明については、[セクション 7.4.5 : GPIO ポート入力データレジスタ \(GPIOx_IDR\) \(x = A..H\)](#) および [セクション 7.4.6 : GPIO ポート出力データレジスタ \(GPIOx_ODR\) \(x = A..H\)](#) を参照してください。

7.3.5 I/O データのビット単位の処理

ビットセット/リセットレジスタ (GPIOx_BSRR) は、アプリケーションが出力データレジスタ (GPIOx_ODR) で各ビットを個別にセット/リセットできる 32 ビットのレジスタです。ビットセット/リセットレジスタのサイズは、GPIOx_ODR の 2 倍です。

GPIOx_ODR の各ビットは、GPIOx_BSRR: BSRR(i) および BSRR(i+SIZE) の 2 つの制御ビットに対応します。1 が書き込まれると、ビット BSRR(i) は対応する ODR(i) ビットを**セット**します。1 が書き込まれると、ビット BSRR(i+SIZE) は ODR(i) の対応するビットを**リセット**します。

GPIOx_BSRR のビットいずれかに 0 を書き込んでも、GPIOx_ODR の対応するビットに変化はありません。GPIOx_BSRR でビットのセットおよびリセットの両操作が行われると、セットの操作が優先します。

GPIOx_BSRR レジスタを使用した GPIOx_ODR の個々のビットの値の変更は、「一度限り」の効果しかなく、GPIOx_ODR ビットをロックすることはありません。GPIOx_ODR ビットは、常に直接アクセスできます。GPIOx_BSRR レジスタを使用することによって、ビット単位の細かな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ミングするときには、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。ほんの 1 回の AHB 書き込みアクセスで 1 つ以上のビットを変更することができます。

7.3.6 GPIO ロック機構

GPIOx_LCKR レジスタに固有の書き込みシーケンスを適用して GPIO 制御レジスタを凍結することができます。凍結されるレジスタは 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、固有の書き込み/読み出しシーケンスを適用する必要があります。このレジスタのビット 16 に正しい LOCK シーケンスが適用されると、LCKR[15:0] の値を使用して I/O の設定がロックされます（書き込みシーケンス中は、LCKR[15:0] の値が同じである必要があります）。LOCK シーケンスがポートビットに適用されると、ポートビットの値は次のリセットまで変更できなくなります。各 GPIOx_LCKR ビットは制御レジスタ (GPIOx_MODER、GPIOx_OTYPER、GPIOx_OSPEEDR、GPIOx_PUPDR、GPIOx_AFRL、GPIOx_AFRH) の対応するビットを凍結します。

GPIOx_LCKR のビット 16 は [15:0] ビットと同時にセットされなければならないため、LOCK シーケンス ([セクション 7.4.8 : GPIO ポート設定ロックレジスタ \(GPIOx_LCKR\) \(x = A..H\)](#) 参照) は、GPIOx_LCKR レジスタへのワード (32 ビット長) アクセスによってのみ実行可能です。

詳細については、[セクション 7.4.8 : GPIO ポート設定ロックレジスタ \(GPIOx_LCKR\) \(x = A..H\)](#) にある LCKR レジスタの説明を参照してください。

7.3.7 I/O オルタネート機能の入力／出力

各 I/O に対して 16 個あるオルタネート機能の I/O の中から 1 個を選択するためのレジスタが 2 個あります。このレジスタを使うと、アプリケーションでの必要に応じて、オルタネート機能を他のピンに接続することができます。

これは、GPIOx_AFRL および GPIOx_AFRH オルタネート機能レジスタを使って各 GPIO 上で多数の

ペリフェラル機能がマルチプレクスされることを意味します。したがってアプリケーションは、各 I/O に対して可能な機能をどれでも選択することができます。AF 選択信号はオルタネート機能入力およびオルタネート機能出力に共通なので、1 つの I/O におけるオルタネート機能入力／出力に対し 1 つのチャンネルが選択されます。

各 GPIO ピン上でどの機能がマルチプレクスされているかについては、データシートを参照してください。

注： アプリケーションでは、各 I/O に対し、可能なペリフェラル機能の中から一度に 1 つを選択することができます。

7.3.8 外部割り込み／ウェイクアップライン

すべてのポートには外部割り込み機能があります。外部割り込みラインを使用するには、ポートを入力モードにする必要があります。[セクション 10.2：外部割り込み／イベントコントローラ \(EXTI\)](#) および [セクション 10.2.3：ウェイクアップイベント管理](#) を参照してください。

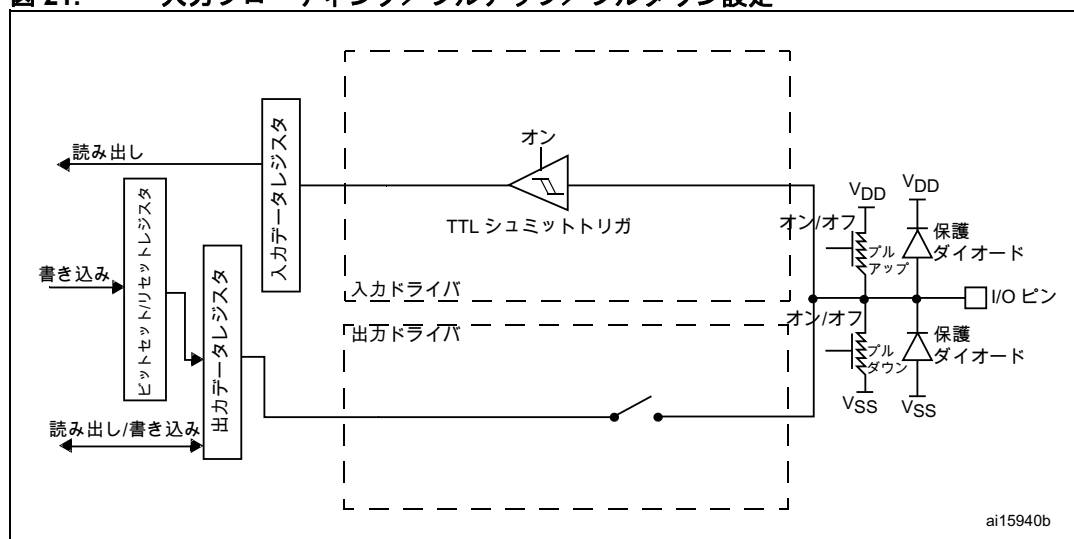
7.3.9 入力設定

I/O ポートが入力としてプログラミングされたときには、

- 出力バッファが無効になります。
- シュミットトリガ入力が入アクティブになります。
- GPIOx_PUPDR レジスタの値により、プルアップおよびプルダウン抵抗がアクティブになります。
- I/O ピン上のデータが 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 状態を知ることができます。

[図 21](#) は、I/O ポートビットの入力設定を示します。

図 21. 入力フローティング／プルアップ／プルダウン設定



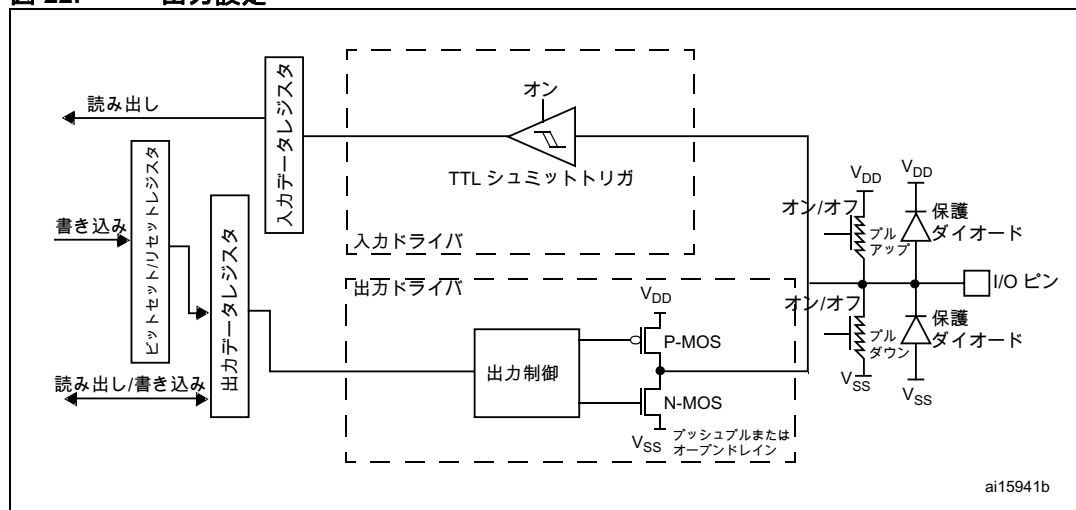
7.3.10 出力設定

I/O ポートが出力としてプログラミングされたときには、

- 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”のときには N-MOS が有効になり、一方、出力レジスタが“1”のときに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ありません)。
 - プッシュプルモード：出力レジスタが“0”のとき N-MOS が有効になり、一方、“1”のときには P-MOS が有効になります。
- シュミットトリガ入力がアクティブになります。
- ウィークプルアップおよびプルダウン抵抗は、GPIOx_PUPDR レジスタの値により、アクティブまたはインアクティブになります。
- I/O ピン上のデータが 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 状態を知ることができます。
- 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を知ることができます。

図 22 は、I/O ポートビットの出力設定を示します。

図 22. 出力設定



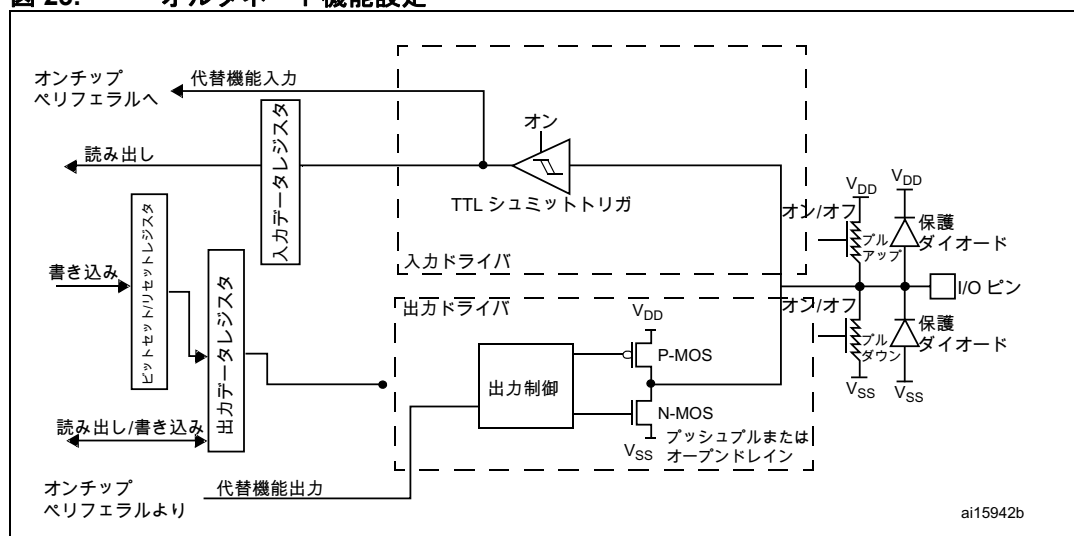
7.3.11 オルタネート機能設定

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ブッシュブルに設定できます。
- 出力バッファは、ペリフェラルからの信号で駆動されます (トランスミッタ有効およびデータ)。
- シュミットトリガ入力がアクティブになります。
- ウィークプルアップおよびプルダウン抵抗は、GPIOx_PUPDR レジスタの値により、アクティブまたはインアクティブになります。
- I/O ピン上のデータが 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 状態を知ることができます。

図 23 は、I/O ポートビットのオルタネート機能設定を示します。

図 23. オルタネート機能設定



7.3.12 アナログ設定

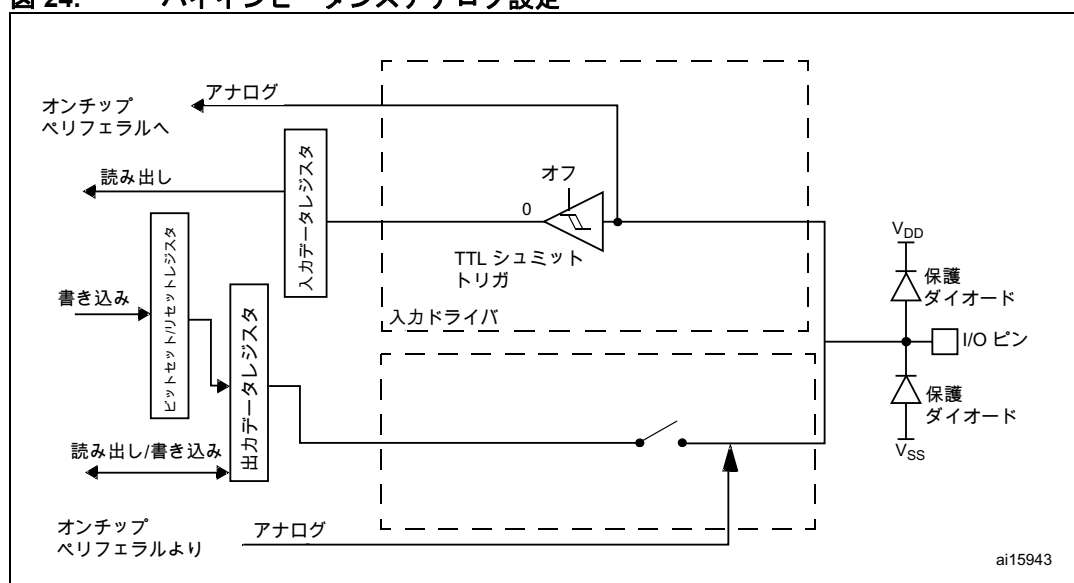
I/O ポートがアナログ設定としてプログラミングされたときには、

- 出力バッファが無効になります。
- シュミットトリガ入力動作は停止し、I/O ピンにおける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が常に 0 になります。
- ウィークプルアップおよびプルダウン抵抗が無効になります。
- 入力データレジスタへの読み出しアクセスでは、値「0」が得られます。

注： 上記に説明したオルタネート機能設定は、選択したオルタネート機能が LCD 機能の場合には適用されません。その場合、オルタネート機能出力としてプログラムされた I/O は、アナログ設定で説明されているように設定されます。

図 24 は、I/O ポートビットのハイインピーダンスのアナログ入力設定を示します。

図 24. ハイインピーダンスアナログ設定



7.3.13 OSC32_IN/OSC32_OUT ピンの GPIO PC14/PC15 ポートピンとしての使用

LSE オシレータのピン OSC32_IN および OSC32_OUT は、LSE オシレータがオフのとき、それぞれ汎用 PC14 および PC15 I/O として使用できます。LSE オシレータがオンのとき、PC14 および PC15 I/O は LSE オシレータピン OSC32_IN および OSC32_OUT としか設定されません。このためには、RCC_BDCR レジスタの LSEON ビットをセットします。LSE は、GPIO 機能より高い優先順位を持っています。

注： PC14/PC15 GPIO 機能が無効になるのは、 V_{CORE} ドメインの電源がオフになったとき (STANDBY モードになったとき) です。この場合、I/O はアナログ入力モードにセットされます。

7.3.14 OSC_IN/OSC_OUT ピンの GPIO PH0/PH1 ポートピンとしての使用

HSE オシレータがオフのとき、HSE オシレータのピン OSC_IN/OSC_OUT は、それぞれ汎用 I/O PH0/PH1 として使用できます。(リセット後、HSE オシレータはオフになります)。HSE オシレータがオンのとき、PH0/PH1 I/O は、OSC_IN/OSC_OUT HSE オシレータピンとしてのみ設定できます。このためには、RCC_CR レジスタの HSEON ビットをセットします。HSE は、GPIO 機能より高い優先順位を持っています。

7.3.15 RTC_AF1 オルタネート機能の選択

STM32L1xxxx の機能は以下のとおりです。

- ウェイクアップピン (WKUP1 および WKUP3) として使用可能な GPIO ピン 2 個。.
- タンパまたはタイムスタンプイベントを検出または RTC AFO_ALARM や AFO_CALIB を出力するウェイクアップピン (WKUP2) として使用可能な GPIO ピン 1 個。.

RTC_AF1 ピン (PC13) は以下の目的で使用できます。

- ウェイクアップピン 2 (WKUP2) : この機能は、PWR_CSR レジスタの EWUP2 をセットすると有効になります。
- RTC AFO_ALARM 出力 : この出力は、RTC_CR レジスタの OSEL[1:0] ビットの値により、RTC Alarm A、RTC Alarm B または RTC ウェイクアップのいずれかとなります。
- RTC AFO_CALIB 出力 : この機能は、RTC_CR レジスタの COE[23] ビットをセットすると有効になります。
- RTC AFL_TAMPER1 : タンパイベント検出
- タイムスタンプイベント検出

RTC AFO_ALARM 出力の選択は、次のようにして RTC_TAFCR レジスタで行われます。ALARMOUTTYPE を使用し、RTC AFO_ALARM 出力をプッシュプルまたはオープンドレインモードのいずれかに設定するかを選択します。

出力メカニズムは、表 38 で示す優先順位に従います。

表 38. RTC_AF1 ピン⁽¹⁾

ピン設定と機能	AFO_ALARM 有効	AFO_CALIB 有効	タンパ有効	タイムスタンプ 有効	EWUP2 有効	ALARMOUTTYPE AFO_ALARM 設定
アラーム出力 OD	1	0	無視	無視	無視	0
アラーム出力 PP	1	0	無視	無視	無視	1
較正出力 PP	0	1	無視	無視	無視	無視
TAMPER 入力フローティング	0	0	1	0	無視	無視
TIMESTAMP および TAMPER 入力フローティング	0	0	1	1	無視	無視
TIMESTAMP 入力フローティング	0	0	0	1	無視	無視
ウェイクアップピン 2	0	0	0	0	1	無視
標準 GPIO	0	0	0	0	0	無視

1. OD : オープンドレイン、PP : プッシュプル

7.4 GPIO レジスタ

このセクションでは、GPIO レジスタの詳細について説明します。
レジスタビット、レジスタアドレスオフセット、リセット値の概要については、表 39 を参照してください。

GPIO 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。

7.4.1 GPIO ポートモードレジスタ (GPIOx_MODER) (x = A..H)

アドレスオフセット : 0x00

リセット値 :

- ポート A : 0xA800 0000
- ポート B : 0x0000 0280
- その他のポート : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[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
MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[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:2y+1 **MODERy[1:0]** : ポート x 設定ビット (y = 0..15)

- これらのビットは、I/O 方向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 (リセット状態)
 - 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

7.4.2 GPIO ポート出力タイプレジスタ (GPIOx_OTYPER) (x = A..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
予約済み															
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、リセット値のままにしておかなければなりません。

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 ポートの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

7.4.3 GPIO ポート出力速度レジスタ (GPIOx_OSPEEDR) (x = A..H)

アドレスオフセット : 0x08

リセット値 :

- ポート B : 0x0000 00C0
- その他のポート : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OSPEEDR15 [1:0]		OSPEEDR14 [1:0]		OSPEEDR13 [1:0]		OSPEEDR12 [1:0]		OSPEEDR11 [1:0]		OSPEEDR10 [1:0]		OSPEEDR9 [1:0]		OSPEEDR8 [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
OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1 [1:0]		OSPEEDR0 [1:0]	
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 2y:2y+1 OSPEEDRy[1:0] : ポート x 設定ビット (y = 0..15)

これらのビットは、I/O の出力速度を設定するために、ソフトウェアによって書き込まれます。

00 : 超低速

01 : 低速

10 : 中速

11 : 高速

注 : OSPEEDRy ビットの値と V_{DD} の範囲および外部負荷については、製品データシートを参照してください。

7.4.4 GPIO ポートプルアップ/プルダウンレジスタ (GPIOx_PUPDR) (x = A..H)

アドレスオフセット : 0x0C

リセット値 :

- ポート A : 0x6400 0000
- ポート B : 0x0000 0100
- その他のポート : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[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
PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[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:2y+1 **PUPDRy[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

7.4.5 GPIO ポート入力データレジスタ (GPIOx_IDR) (x = A..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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0
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 **IDRy** : ポート入力データ (y = 0..15)

これらのビットは、読み出し専用であり、ワードモードでのみアクセスできます。対応する I/O ポートの入力値を含みます。

7.4.6 GPIO ポート出力データレジスタ (GPIOx_ODR) (x = A..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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0
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 15:0 **ODRy** : ポート出力データ (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 : ビット単位のセット/リセットを行うにあたり、GPIOx_BSRR レジスタ (x = AH) へ書き込むことによって、ODR ビットを個々にセット/リセットできます。

7.4.7 GPIO ポートビットセット/リセットレジスタ (GPIOx_BSRR) (x = A..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、書き込み専用であり、ワード、ハーフワードまたはバイトモードでのみアクセスできます。これらのビットを読み出すと、値 0x0000 が返ります。

0 : 対応する ODRx ビットの値は変化しません。

1 : 対応する ODRx 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、書き込み専用であり、ワード、ハーフワードまたはバイトモードでのみアクセスできます。これらのビットを読み出すと、値 0x0000 が返ります。

0 : 対応する ODRx ビットの値は変化しません。

1 : 対応する ODRx ビットをセットします。

7.4.8 GPIO ポート設定ロックレジスタ (GPIOx_LCKR) (x = A..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適用されたときに、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] の値は同じでなければなりません。LOCK シーケンスがポートビットに適用されると、このポートビットの値は次のリセットまで変更できなくなります。

注： GPIOx_LCKR レジスタの書き込みには、特定の書き込みシーケンスを使用します。この書き込みシーケンス中は、ワード単位でのアクセス (32 ビット長) のみが許可されます。

各ロックビットは、特定の設定レジスタ (制御およびオルタネート機能レジスタ) を凍結します。

アドレスオフセット : 0x1C

リセット値 : 0x0000 0000

アクセス : 32 ビットのワードのみ。読み出し/書き込みレジスタ

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															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、リセット値のままにしておかなければなりません。

ビット 16 LCKK[16] : 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インアクティブです。

1 : ポート設定ロックキーはアクティブです。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 :

WR LCKR[16] = "1" + LCKR[15:0]

WR LCKR[16] = "0" + LCKR[15:0]

WR LCKR[16] = "1" + LCKR[15:0]

RD LCKR

RD LCKR[16] = "1" (この読み出し操作はオプションですが、ロックがアクティブであることを確認します)。

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。

ポートのビットいずれかに初めてロックシーケンスが適用された後、LCKK ビットいずれかに読み出しアクセスを行うと次の CPU リセットまでは"1"が返ります。

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。



7.4.9 GPIO オルタネート機能ローレジスタ (GPIOx_AFRL) (x = A..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
AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[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
AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[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 **AFRLy** : ポート x ビットのオルタネート機能選択 (y = 0..7)

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。

AFRLy の選択 :

0000 : AF0	1000 : AF8
0001 : AF1	1001 : AF9
0010 : AF2	1010 : AF10
0011 : AF3	1011 : AF11
0100 : AF4	1100 : AF12
0101 : AF5	1101 : AF13
0110 : AF6	1110 : AF14
0111 : AF7	1111 : AF15



7.4.10 GPIO オルタネート機能ハイレジスタ (GPIOx_AFRH) (x = A..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
AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[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
AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[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 **AFRHy** : ポート x ビットのオルタネート機能選択 (y = 8..15)

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。

AFRHy の選択 :

0000 : AF0	1000 : AF8
0001 : AF1	1001 : AF9
0010 : AF2	1010 : AF10
0011 : AF3	1011 : AF11
0100 : AF4	1100 : AF12
0101 : AF5	1101 : AF13
0110 : AF6	1110 : AF14
0111 : AF7	1111 : AF15

7.4.11 GPIO ポートビットリセットレジスタ (GPIOx_BRR) (x=A..H)

これらのレジスタは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

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
予約済み															
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 が返ります。

- 0 : 対応する ODRx ビットの値は変化しません。
- 1 : 対応する ODRx ビットをリセットします。



7.4.12 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 39. 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	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[1:0]	
	リセット値	1	0	1	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
0x00	GPIOB_ MODER	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[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	1	0	1	0	0	0	0	0	0	0	0
0x00	GPIOx_MODER (ここで x = C..H)	MODER15[1:0]		MODER14[1:0]		MODER13[1:0]		MODER12[1:0]		MODER11[1:0]		MODER10[1:0]		MODER9[1:0]		MODER8[1:0]		MODER7[1:0]		MODER6[1:0]		MODER5[1:0]		MODER4[1:0]		MODER3[1:0]		MODER2[1:0]		MODER1[1:0]		MODER0[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
0x04	GPIOx_ OTYPER (ここで x = A..H)	予約済み																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	GPIOx_ OSPEEDER (こ こで x = A..H)	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[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
0x08	GPIOB_ OSPEEDER	OSPEEDR15[1:0]		OSPEEDR14[1:0]		OSPEEDR13[1:0]		OSPEEDR12[1:0]		OSPEEDR11[1:0]		OSPEEDR10[1:0]		OSPEEDR9[1:0]		OSPEEDR8[1:0]		OSPEEDR7[1:0]		OSPEEDR6[1:0]		OSPEEDR5[1:0]		OSPEEDR4[1:0]		OSPEEDR3[1:0]		OSPEEDR2[1:0]		OSPEEDR1[1:0]		OSPEEDR0[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	1	1	0	0	0	0	0	0
0x0C	GPIOA_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[1:0]	
	リセット値	0	1	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	GPIOB_PUPDR	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[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	1	0	0	0	0	0	0	0	0	0
0x0C	GPIOx_PUPDR (ここで x = C..H)	PUPDR15[1:0]		PUPDR14[1:0]		PUPDR13[1:0]		PUPDR12[1:0]		PUPDR11[1:0]		PUPDR10[1:0]		PUPDR9[1:0]		PUPDR8[1:0]		PUPDR7[1:0]		PUPDR6[1:0]		PUPDR5[1:0]		PUPDR4[1:0]		PUPDR3[1:0]		PUPDR2[1:0]		PUPDR1[1:0]		PUPDR0[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

表 39. 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	
0x10	GPIOx_IDR（こ で x = A..H）	予約済み																IDR15	IDR14	IDR13	IDR12	IDR11	IDR10	IDR9	IDR8	IDR7	IDR6	IDR5	IDR4	IDR3	IDR2	IDR1	IDR0	
	リセット値																	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
0x14	GPIOx_ODR（こ で x = A..H）	予約済み																ODR15	ODR14	ODR13	ODR12	ODR11	ODR10	ODR9	ODR8	ODR7	ODR6	ODR5	ODR4	ODR3	ODR2	ODR1	ODR0	
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x18	GPIOx_BSRR （ここで x = A..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		
0x1C	GPIOx_LCKR（こ で x = A..H）	予約済み																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（こ で x = A..H）	AFRL7[3:0]				AFRL6[3:0]				AFRL5[3:0]				AFRL4[3:0]				AFRL3[3:0]				AFRL2[3:0]				AFRL1[3:0]				AFRL0[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		
0x24	GPIOx_AFRH（こ で x = A..H）	AFRH15[3:0]				AFRH14[3:0]				AFRH13[3:0]				AFRH12[3:0]				AFRH11[3:0]				AFRH10[3:0]				AFRH9[3:0]				AFRH8[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		
0x28	GPIOx_BRR （ここで x = A..H）	予約済み																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	0

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

8 システム設定コントローラ (SYSCFG) と ルーティングインタフェース (RI)

このセクションは、特に指定がない限り、STM32L1xxxx ファミリ全体に適用されます。

8.1 SYSCFG と RI について

システム設定コントローラは、主にコード領域でアクセス可能なメモリの再配置や GPIO への外部割り込みライン接続の管理に使用します。

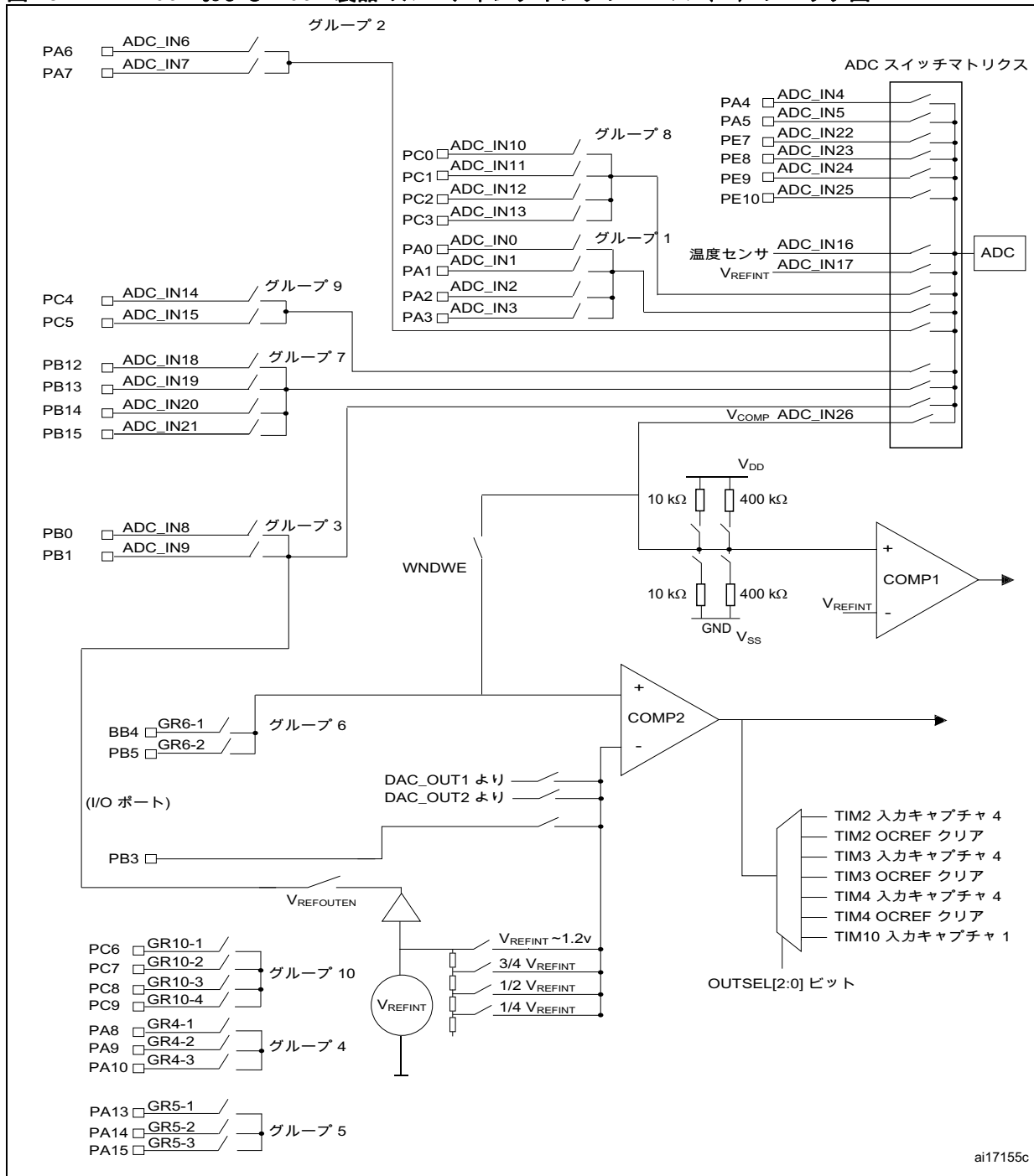
ルーティングインタフェースは、STM32L1xxxx の 3 個のハイエンドタイマ (TIM2、TIM3、TIM4) の入力キャプチャに対する I/O のソフトウェアルーティングを可能にして、高い柔軟性を提供します。STM32L1xxxx の ADC には、通常特定の ADC インタフェースが管理するアナログ入力マトリックスがあります。ルーティングインタフェースを使うと、各 I/O のアナログスイッチを管理して複数の I/O アナログピンを ADC マトリックスの与えられたチャンネルに接続することができます。

8.2 RI の主な特徴

- 選択可能な I/O からの TIM2/TIM3/TIM4 の入力キャプチャ 1、2、3 および 4 個のルーティング選択
- すべてのパッケージの選択可能な I/O に対する内部基準電圧 V_{REFINT} のルーティング
- 最大 40 個の外部 I/O + 3 個の内部ノード (内部基準電圧 + 温度センサ + V_{COMP} による V_{DD} および $V_{DD/2}$ の測定) を、ADC インタフェースと併せてデータ取得目的で使用可能使用
- COMP1 および COMP2 の I/O ルーティング

注： RI レジスタにアクセスできるのは、RCC_APB1ENR レジスタの COMPEN ビットをセットしてコンパレータインタフェースクロックが有効になっているときに限られます。セクション 6.3.10 (161 ページ) を参照してください。

図 25. Cat.1 および Cat.2 製品のルーティングインタフェース (RI) ブロック図



ai17155c

図 26. Cat.3 製品のルーティングインタフェース (RI) ブロック図

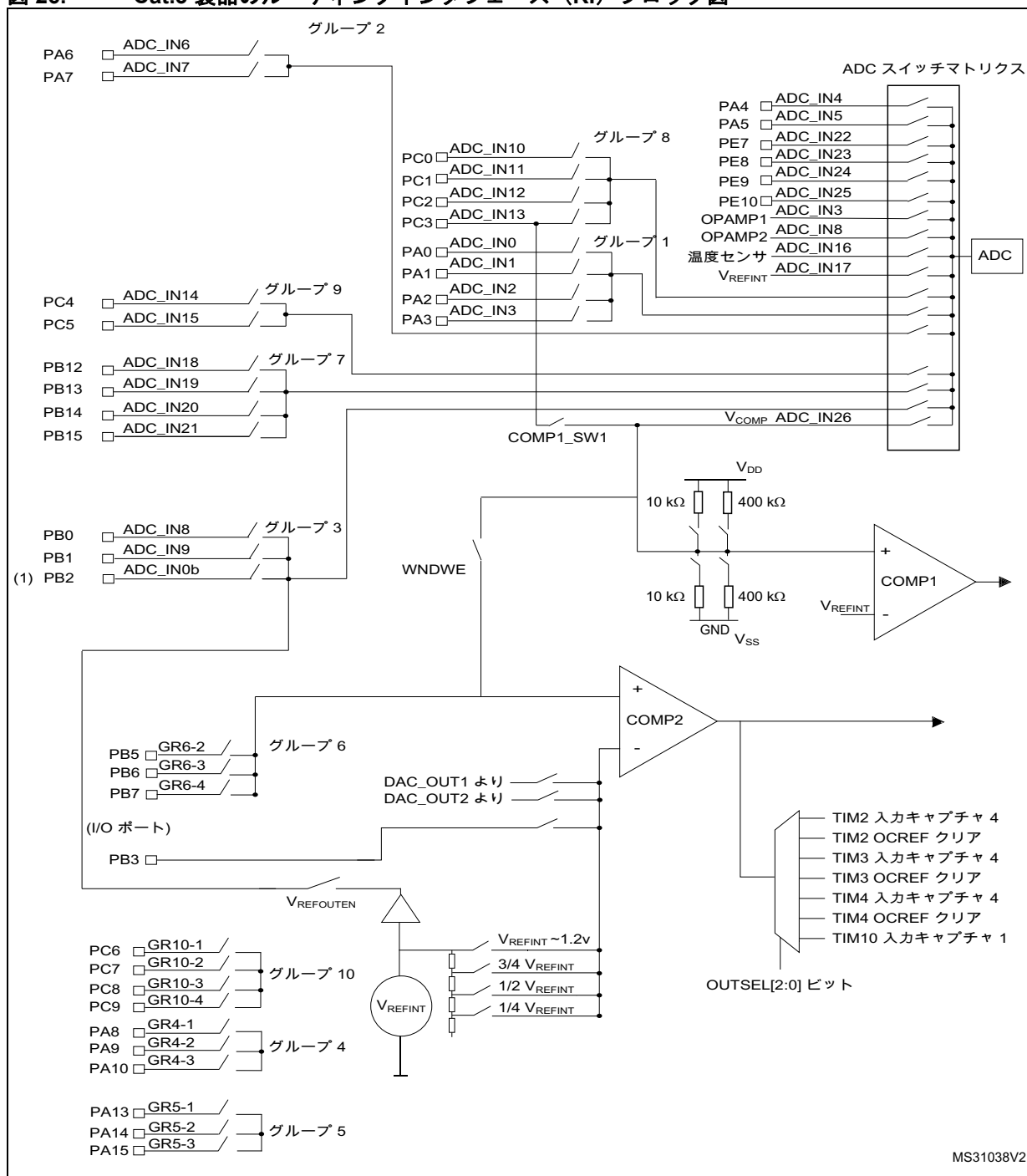
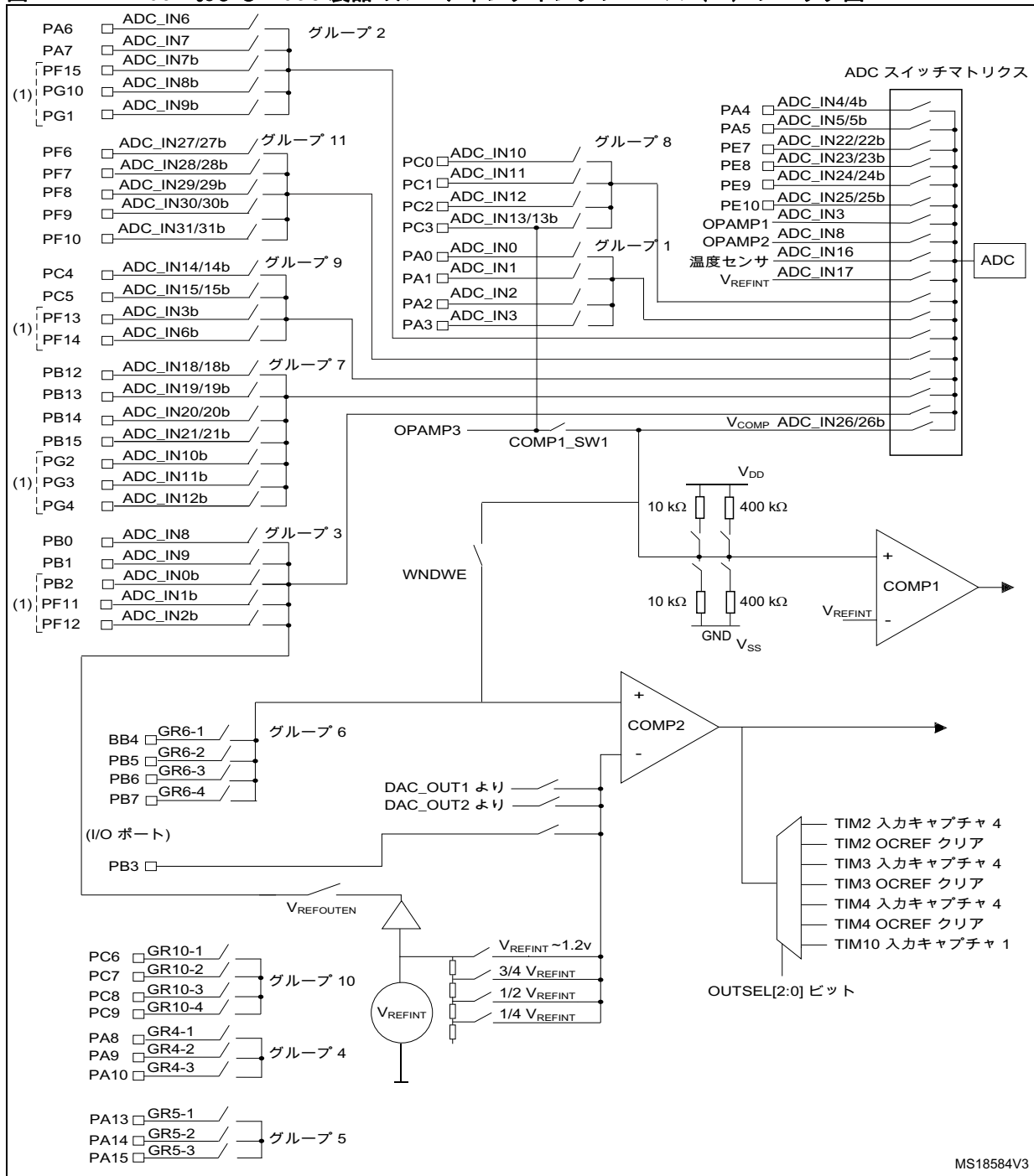


図 27. Cat.4 および Cat.5 製品のルーティングインタフェース (RI) ブロック図



MS18584V3

1. これらの I/O ピンは、COMP1 入力として使用できません。

8.3 RI の機能詳細

8.3.1 特別 I/O 設定

I/O スイッチと ADC スイッチというスイッチの 2 つのマトリックスがアナログブロック (ADC またはコンパレータ) に対する I/O のルーティングを制御します (図 25 : Cat.1 および Cat.2 製品のルーティングインタフェース (RI) ブロック図参照)。

- I/O がデータ取得以外のアナログ目的で 사용되는場合、I/O および ADC スイッチマトリックスは、RI_ASCR1 および RI_ASCR2 レジスタによって制御される必要があります。その後これらのレジスタを使用して、ソフトウェアによってスイッチを閉じたり開いたりします。スイッチを閉じると対応するビットがセットされ、スイッチを開くとビットがクリアされます。
- I/O をデータ取得目的で ADC 入力として使用する場合、I/O および ADC スイッチマトリックスは直接 ADC インタフェースで制御されます。RI_ASCR1 および RI_ASCR2 レジスタの対応するビットはクリアされたままである必要があります (スイッチ開)。

Cat.3、Cat.4、Cat.5 製品 :

ADC には最大 6 個の I/O が直接独立して接続され、これらはデータを高速で取得することができます (最大 ADC 取得時間)。その他すべての ADC チャンネルではサンプリングレートが低くなっています。製品データシートを参照してください。オペアンプ 1 および 2 の出力は、直接 ADC スイッチマトリックスに接続されており、フルサンプリングレートでも使用できます。オペアンプ 3 の出力は、Cat.4 および Cat.5 製品でしか利用できないさらに別の COMP1_SW1 スイッチを通過して ADC スイッチマトリックスに接続されるため、これは適用されません。

表 40 : I/O グループと選択 (202 ページ) に示すように、50 個の I/O が 11 個のグループにまとめられ、付録に記載されたアプリケーションに対応しています。標準 GPIO レジスタによって入力モードでプログラムされると、シュミットトリガおよびヒステリシスがデフォルトで有効になります。このモードでは、レジスタ RI_ASCRx および RI_HYSCR によって対応するアナログスイッチパッドを閉じ、同時にシュミットトリガヒステリシスを無効にすることができます。したがって、VDDIO/2 のトリガレベルを使って対応するポートを読み出すことができます。

11 個あるグループのうち、アナログスイッチを拡張することによって ADC に対してマルチプレクスされるのは 7 個のグループのみです (34 個の I/O)。高速独立チャンネルが 6 つありますので、データ取得には 40 個の I/O が利用できますが、COMP1 の Vref 帯域幅との比較に利用できるのは 29 個だけです (図 63 : COMP1 の接続 (Cat.1 と Cat.2 デバイス) (ページ) 346 および 図 64 : COMP1 の接続 (Cat.3、Cat.4、Cat.5 デバイス) (ページ) 347 参照)。

これらのアンプを選択した場合、各 OPAMP 専用の特定のチャンネルは、上述した 40 個の I/O のうち 3 個の I/O を使用します。(24 ページの「2 章 : オペアンプインタフェース」を参照)

Cat.1 および Cat.2 製品 :

注 : コンパレータ入力として使用されるすべての I/O においては、I/O ポート設定をアナログモードにしておく必要があります。

表 40 に、I/O のグループ分け、I/O をアナログ入力または出力として設定するのに使用する制御レジスタビット (標準 I/O ポートプログラミングとは無関係)、関連 ADC チャンネル番号を示します。

表 40. I/O グループと選択

グループ番号		GPIO ポート	アナログ ADC チャンネル	I/O +ADC アナログスイッチ	I/O 機能
グループ 1	GR1-1	PA0	ADC_IN0	RI_ASCR1->CH0	COMP1_INP
	GR1-2	PA1	ADC_IN1	RI_ASCR1->CH1	
	GR1-3	PA2	ADC_IN2	RI_ASCR1->CH2	
	GR1-4	PA3	ADC_IN3	RI_ASCR1->CH3	
グループ 2	GR2-1	PA6	ADC_IN6	RI_ASCR1->CH6	COMP1_INP
	GR2-2	PA7	ADC_IN7	RI_ASCR1->CH7	
	GR2-3	PF15	ADC_IN7b	RI_ASCR2->CH7b	
	GR2-4 ⁽¹⁾	PG0 ⁽²⁾	ADC_IN8b	RI_ASCR2->CH8b	
	GR2-5 ⁽¹⁾	PG1 ⁽²⁾	ADC_IN9b	RI_ASCR2->CH9b	
グループ 3	GR3-1	PB0	ADC_IN8	RI_ASCR1->CH8	COMP1_INP /VREF_OUT
	GR3-2	PB1	ADC_IN9	RI_ASCR1->CH9	
	GR3-3 ⁽¹⁾	PB2	ADC_IN0b	RI_ASCR2->CH0b	COMP1_INP
	GR3-4 ⁽¹⁾	PF11	ADC_IN1b	RI_ASCR2->CH1b	
	GR3-5 ⁽¹⁾	PF12	ADC_IN2b	RI_ASCR2->CH2b	
グループ 4	GR4-1	PA8	該当なし	RI_ASCR2->GR4-1	-
	GR4-2	PA9		RI_ASCR2->GR4-2	
	GR4-3	PA10		RI_ASCR2->GR4-3	
グループ 5	GR5-1	PA13	該当なし	RI_ASCR2->GR5-1	-
	GR5-2	PA14		RI_ASCR2->GR5-2	
	GR5-3	PA15		RI_ASCR2->GR5-3	
グループ 6	GR6-1	PB4	該当なし	RI_ASCR2->GR6-1	COMP2_INP
	GR6-2	PB5		RI_ASCR2->GR6-2	
	GR6-3 ⁽¹⁾	PB6		RI_ASCR2->GR6-3	
	GR6-4 ⁽¹⁾	PB7		RI_ASCR2->GR6-4	COMP2_INP/PVD_IN
グループ 7	GR7-1	PB12	ADC_IN18 /ADC_IN18b	RI_ASCR1->CH18	COMP1_INP
	GR7-2	PB13	ADC_IN19/ ADC_IN19b	RI_ASCR1->CH19	
	GR7-3	PB14	ADC_IN20/ ADC_IN20b	RI_ASCR1->CH20	
	GR7-4	PB15	ADC_IN21/ ADC_IN21b	RI_ASCR1->CH21	
	GR7-5 ⁽¹⁾	PG2 ⁽²⁾	ADC_IN10b	RI_ASCR2->CH10b	
	GR7-6 ⁽¹⁾	PG3 ⁽²⁾	ADC_IN11b	RI_ASCR2->CH11b	
	GR7-7 ⁽¹⁾	PG4 ⁽²⁾	ADC_IN12b	RI_ASCR2->CH12b	

表 40. I/O グループと選択 (続き)

グループ番号		GPIO ポート	アナログ ADC チャンネル	I/O +ADC アナログスイッチ	I/O 機能
グループ 8	GR8-1	PC0	ADC_IN10	RI_ASCR1->CH10	COMP1_INP
	GR8-2	PC1	ADC_IN11	RI_ASCR1->CH11	
	GR8-3	PC2	ADC_IN12	RI_ASCR1->CH12	
	GR8-4	PC3	ADC_IN13/ ADC_IN13b	RI_ASCR1->CH13	
グループ 9	GR9-1	PC4	ADC_IN14/ ADC_IN14b	RI_ASCR1->CH14	COMP1_INP
	GR9-2	PC5	ADC_IN15/ ADC_IN15b	RI_ASCR1->CH15	
	GR9-3 ⁽¹⁾	PF13	ADC_IN3b	RI_ASCR2->GR9-3	
	GR9-4 ⁽¹⁾	PF14	ADC_IN6b	RI_ASCR2->GR9-4	
グループ 10	GR10-1	PC6	該当なし	RI_ASCR2->GR10-1	-
	GR10-2	PC7		RI_ASCR2->GR10-2	
	GR10-3	PC8		RI_ASCR2->GR10-3	
	GR10-4	PC9		RI_ASCR2->GR10-4	
グループ 11	GR11-1	PF6	ADC_IN27/ ADC_IN27b	RI_ASCR1->CH27	COMP1_INP
	GR11-2	PF7	ADC_IN28/ ADC_IN28b	RI_ASCR1->CH28	
	GR11-3	PF8	ADC_IN29/ ADC_IN29b	RI_ASCR1->CH29	
	GR11-4	PF9	ADC_IN30/ ADC_IN30b	RI_ASCR1->CH30	
	GR11-5	PF10	ADC_IN31/ ADC_IN31b	RI_ASCR1->CH31	
直接チャンネル		PA4	ADC_IN4/ ADC_IN4b	RI_ASCR1->CH4	COMP1_INP/DAC1
		PA5	ADC_IN5/ ADC_IN5b	RI_ASCR1->CH5	COMP1_INP/DAC2
		PE7	ADC_IN22/ ADC_IN22b	RI_ASCR1->CH22	COMP1_INP
		PE8	ADC_IN23/ ADC_IN23b	RI_ASCR1->CH23	COMP1_INP
		PE9	ADC_IN24/ ADC_IN24b	RI_ASCR1->CH24	COMP1_INP
		PE10	ADC_IN25/ ADC_IN25b	RI_ASCR1->CH25	COMP1_INP
OPAMP1_VOUT		-	ADC_IN3	該当なし	-
直接チャンネル		PA3	ADC_IN3	COMP_CSR->FCH3	-
OPAMP2_VOUT		-	ADC_IN8	該当なし	-
直接チャンネル		PB0	ADC_IN8	COMP_CSR->FADC_IN8	-

表 40. I/O グループと選択 (続き)

グループ番号	GPIO ポート	アナログ ADC チャンネル	I/O +ADC アナログスイッチ	I/O 機能
OPAMP3_VOUT		ADC_IN13	該当なし	-
-	PC3	ADC_IN13	COMP_CSR->RCH13	-
-	V _{TS}	ADC_IN16/ ADC_IN16b	該当なし	-
-	V _{REFINT}	ADC_IN17/ ADC_IN17b	該当なし	-
-	V _{COMP}	ADC_IN26/ ADC_IN26b	該当なし	-
該当なし	PB3		該当なし	COMP2_INM
該当なし	PB6 ⁽¹⁾		該当なし	COMP2_INP
該当なし	PB7		該当なし	PVD_IN/COMP2_INP

1. Cat.3、Cat.4、Cat.5 製品でのみ利用可能。

2. タッチセンシングソリューションで使用する場合、これらの GPIO はサンプリングコンデンサ I/O としてしか設定できません。

8.3.2 入力キャプチャのルーティング

デフォルト（リセット時）では、3つの汎用タイマ（TIM2、TIM3、TIM4）の4つの入力キャプチャがSTM32L1xxx データシートの「ピン説明」表に記載されている入力ポートに接続されています。

I/O ルーティングを変更するには、以下に示すように RI_ICR レジスタをプログラミングします。

- TIM2、TIM3、TIM4 の入力キャプチャ 1 は、RI_ICR の IC1IOS[3:0] ビットを設定すると、任意の I/O から再ルーティングできます。
- TIM2、TIM3、TIM4 の入力キャプチャ 2 は、RI_ICR の IC2IOS[3:0] ビットを設定すると、任意の I/O から再ルーティングできます。
- TIM2、TIM3、TIM4 の入力キャプチャ 3 は、RI_ICR の IC3IOS[3:0] ビットを設定すると、任意の I/O から再ルーティングできます。
- TIM2、TIM3、TIM4 の入力キャプチャ 4 は、RI_ICR の IC4IOS[3:0] ビットを設定すると、任意の I/O から再ルーティングできます。

入力キャプチャタイマへの I/O ルーティングについては、以下の表を参照してください。

この機能は、RI_ICR の TIM[1:0] を設定して一度に3つのタイマのうち1つだけに適用できます。TIM[1:0]=00 のときは、3つのタイマのいずれも I/O ルーティングに影響されません。デフォルト接続が有効になっています。

さらに、タイマを選択したときに、RI_ICR の IC1、IC2、IC3、IC4 ビットを設定すると、1つ以上の入力キャプチャに対して I/O ルーティングを有効にできます。

I/O の対応については表 41 を、またタイマ選択については表 42 を参照してください。

表 41. 入力キャプチャの割り当て

IC1IOS/IC2IOS/IC3IOS/IC4IOS	TIMx IC1/TIMx IC2/TIMx IC3/TIMx IC4
0000	PA0/PA1/PA2/PA3
0001	PA4/PA5/PA6/PA7
0010	PA8/PA9/PA10/PA11
0011	PA12/PA13/PA14/PA15
0100	PC0/PC1/PC2/PC3
0101	PC4/PC5/PC6/PC7
0110	PC8/PC9/PC10/PC11
0111	PC12/PC13/PC14/PC15
1000	PD0/PD1/PD2/PD3
1001	PD4/PD5/PD6/PD7
1010	PD8/PD9/PD10/PD11
1011	PD12/PD13/PD14/PD15
1100	PE0/PE1/PE2/PE3
1101	PE4/PE5/PE6/PE7
1110	PE8/PE9/PE10/PE11
1111	PE12/PE13/PE14/PE15

注： I/O はオルタネート機能モード (AF14) で設定します。



表 42. タイマ選択

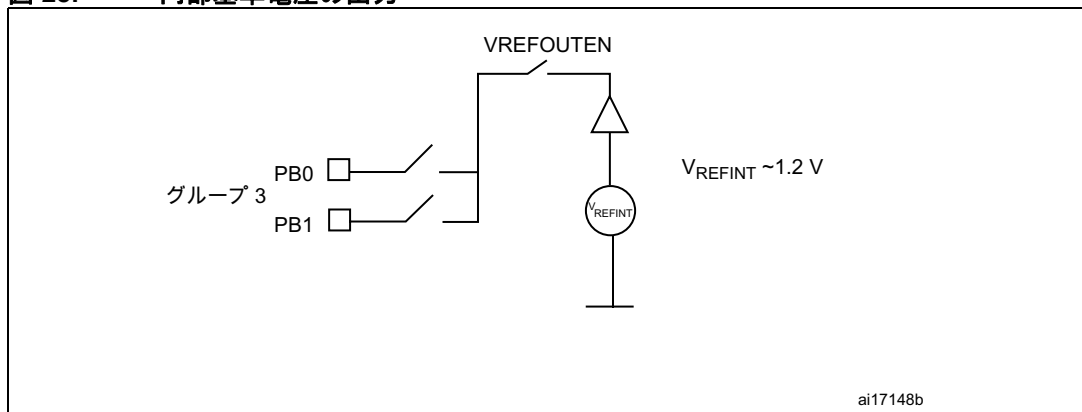
TIM[1:0]	選択したタイマ
00	選択されているタイマはありません。すべてのタイマでデフォルトのルーティングです。
01	TIM2 を選択
10	TIM3 を選択
11	TIM4 を選択

表 43. 入力キャプチャの選択

IC4/IC3/IC2/IC1	選択した入力キャプチャ
0	IC が非選択になっています。入力キャプチャ (AF) ではデフォルトのルーティングです。
1	入力キャプチャのルーティングは表 42 のようになっています。

8.3.3 基準電圧のルーティング

図 28. 内部基準電圧の出力



V_{REFINT} の出力は、次の手順に従って、グループ 3 の任意の I/O にルーティングできます。

1. COMP_CSR の VREFOUTEN ビットをセットします。
2. RI_ASCR1 の CH8 または CH9 をセットして、グループ 3 のすべての I/O のアナログスイッチを閉じます。

8.4 RI 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。

8.4.1 RI 入力キャプチャレジスタ (RI_ICR)

RI_ICR レジスタは、TIM2、TIM3、TIM4 の入力キャプチャに対する 4 つのフルポートのルーティングの選択に使用します。

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										IC4	IC3	IC2	IC1	TIM[1:0]	
										rW	rW	rW	rW	rW	rW
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IC4IOS[3:0]				IC3IOS[3:0]				IC2IOS[3:0]				IC1IOS[3:0]			
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 31:22 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 **IC4** : このビットは、ソフトウェアによってセット/クリアされ、TIM[1:0] (ビット 17:16) で選択したタイムの入力キャプチャ 4 における標準 AF または大規模なルーティング機能を選択します。

- 0 : IC4 で AF
- 1 : IC4IOS[3:0] (ビット 15:12) に従った複数ポートルーティング機能

ビット 20 **IC3** : このビットは、ソフトウェアによってセット/クリアされ、TIM[1:0] (ビット 17:16) で選択したタイムの入力キャプチャ 3 における標準 AF または大規模なルーティング機能を選択します。

- 0 : IC3 で AF
- 1 : IC3IOS[3:0] (ビット 11:8) に従った複数ポートルーティング機能

ビット 19 **IC2** : このビットは、ソフトウェアによってセット/クリアされ、TIM[1:0] (ビット 17:16) で選択したタイムの入力キャプチャ 2 における標準 AF または大規模なルーティング機能を選択します。

- 0 : IC2 で AF
- 1 : IC2IOS[3:0] (ビット 7:4) に従った複数ポートルーティング機能

ビット 18 **IC1** : このビットは、ソフトウェアによってセット/クリアされ、TIM[1:0] (ビット 17:16) で選択したタイムの入力キャプチャ 1 における標準 AF または大規模なルーティング機能を選択します。

- 0 : IC1 で AF
- 1 : IC1IOS[3:0] (ビット 3:0) に従った複数ポートルーティング機能

ビット 17:16 **TIM[1:0]** : タイマ選択ビット

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。3 つのタイムのうち 1 つを選択するか、どのタイムも選択しません。

- 00 : タイマは選択されていません。
- 01 : TIM2 が選択されています。
- 10 : TIM3 が選択されています。
- 11 : TIM4 が選択されています。

ビット 15:12 **IC4IOS[3:0]** : 入力キャプチャ 4 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。選択したタイマの IC4 にルーティングされる入力ポートを選択します (ビット 16:17 参照)。

0000 : PA3	1000 : PD3
0001 : PA7	1001 : PD7
0010 : PA11	1010 : PD11
0011 : PA15	1011 : PD15
0100 : PC3	1100 : PE3
0101 : PC7	1101 : PE7
0110 : PC11	1110 : PE11
0111 : PC15	1111 : PE15

ビット 11:8 **IC3IOS[3:0]** : 入力キャプチャ 3 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。選択したタイマの IC3 にルーティングされる入力ポートを選択します (ビット 16:17 参照)。

0000 : PA2	1000 : PD2
0001 : PA6	1001 : PD6
0010 : PA10	1010 : PD10
0011 : PA14	1011 : PD14
0100 : PC2	1100 : PE2
0101 : PC6	1101 : PE6
0110 : PC10	1110 : PE10
0111 : PC14	1111 : PE14

ビット 7:4 **IC2IOS[3:0]** : 入力キャプチャ 2 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。選択したタイマの IC2 にルーティングされる入力ポートを選択します (ビット 16:17 参照)。

0000 : PA1	1000 : PD1
0001 : PA5	1001 : PD5
0010 : PA9	1010 : PD9
0011 : PA13	1011 : PD13
0100 : PC	1100 : PE1
0101 : PC5	1101 : PE5
0110 : PC9	1110 : PE9
0111 : PC13	1111 : PE13

ビット 3:0 **IC1IOS[3:0]** : 入力キャプチャ 1 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。選択したタイマの IC1 にルーティングされる入力ポートを選択します (ビット 16:17 参照)。

0000 : PA0	1000 : PD0
0001 : PA4	1001 : PD4
0010 : PA8	1010 : PD8
0011 : PA12	1011 : PD12
0100 : PC0	1100 : PE0
0101 : PC4	1101 : PE4
0110 : PC8	1110 : PE8
0111 : PC12	1111 : PE12

注 :

TIM2 専用の標準 AF :

IC4-> PA3、PB11 または PE12

IC3-> PA2、PB10 または PE11

IC2-> PA1、PB3 または PE10

IC1-> PA0、PA5、PA15 または PE9

TIM3 専用の標準 AF :

IC4-> PB1 または PC9

IC3-> PB0 または PC8
 IC2-> PA7、PC7、PB5 または PE4
 IC1-> PA6、PC6、PB4 または PE3
 TIM4 専用の標準 AF :
 IC4-> PD15 または PB9
 IC3-> PD14 または PB8
 IC2-> PD13 または PB7
 IC1-> PD12 または PB6

8.4.2 RI アナログスイッチ制御レジスタ (RI_ASCR1)

RI_ASCR1 レジスタは、ADC にリンクされる I/O のアナログスイッチの設定に使用します。これらの I/O は、ADC チャンネル番号で参照されます。

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
SCM	CH30 GR11-4	CH29 GR11-3	CH28 GR11-2	CH27 GR11-1	Vcomp	CH25	CH24	CH23	CH22	CH21 GR7-4	CH20 GR7-3	CH19 GR7-2	CH18 GR7-1	予約済み	CH31 GR11-5
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
CH15 GR9-2	CH14 GR9-1	CH13 GR8-4	CH12 GR8-3	CH11 GR8-2	CH10 GR8-1	CH9 GR3-2	CH8 GR3-1	CH7 GR2-2	CH6 GR2-1	CH5	CH4	CH3 GR1-4	CH2 GR1-3	CH1 GR1-2	CH0 GR1-1
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	r/w

ビット 31 **SCM** : ADC スイッチ制御モード

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ると、アナログ I/O スイッチを制御する RI_ASCRx のビットのセットによって ADC スイッチマトリックスの対応スイッチも閉じます。このビットがリセットされると、RI_ASCRx のその他のビットは ADC スイッチマトリックスのスイッチを制御しません。

0 : アナログスイッチが開いているまたは ADC インタフェースで制御されます。

1 : 対応する I/O スイッチが閉じていれば、ADC アナログスイッチも閉じています。

ビット 30:27 **CH[30:27]/GR11[4:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

0 : アナログスイッチが開いているまたは ADC インタフェースで制御されます。

1 : アナログスイッチは閉じています。

注 : これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 26 **VCOMP** : コンパレータ 1 に対する内部ノードの ADC アナログスイッチ選択

このビットは、VCOMP ADC アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。図 63 (ページ) 346 および 図 64 (ページ) 347 を参照してください。

0 : アナログスイッチを開きます。

1 : アナログスイッチを閉じます。

ビット 25:22 **CH[25:22]** : チャンネル CH[25:22] のアナログ I/O スイッチ制御

これらのビットは、ADC スイッチマトリックスのアナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。I/O が ADC 入力に使用される場合、ADC から制御できるように、スイッチは開いた状態にしておく必要があります。

0 : アナログスイッチを開きます。

1 : アナログスイッチを閉じます。

ビット 21:18 **CH[21:18]/GR7[4:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

ビット 17 予約済み

ビット 16 **CH31/GR11-5** I/O アナログスイッチ制御

このビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

注： このビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 15:14 **CH[15:14] GR9[2:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

ビット 13:10 **CH[13:10] GR8[4:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

ビット 9:8 **CH[9:8] GR3[2:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

ビット 7:6 **CH[7:6] GR2[2:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます、または ADC インタフェースで制御します。
- 1 : アナログスイッチを閉じます。

ビット 5 **CH5** コンパレータ 1 アナログスイッチ

このビットは、COMP1 コンパレータの正の入力を接続する ADC スイッチマトリックスのコアアナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。ADC マトリックスまたは OPAMP3 出力のコンパレータ 1 の正の入力へのルーティングに使用できます。[図 64 \(ページ 347\)](#) を参照してください。

- 0 : アナログスイッチを開きます。
- 1 : アナログスイッチを閉じます。

ビット 4 **CH4** アナログスイッチ制御

このビットは、ADC スイッチマトリックスのアナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチを開きます。
- 1 : アナログスイッチを閉じます。

ビット 3:0 **CH[3:0] GR1[4:1]** I/O アナログスイッチ制御

これらのビットは、I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : アナログスイッチが開いているまたは ADC インタフェースで制御されます。
- 1 : アナログスイッチを閉じます。

注 : ADC_IN16 および ADC_IN17 チャンネルは内部にあり、データ取得目的で ADC インタフェースによって制御されます。

ADC_IN4、ADC_IN5、ADC_IN22、ADC_IN23、ADC_IN24、ADC_IN25 チャンネルは抵抗器を介して直接 ADC に接続しており、外部 I/O アナログスイッチを閉じる必要はありません。

SCM ビットがローの場合、CH ビットを使用して ADC とは独立したアナログスイッチで I/O のグループをまとめて接続します。

SCM ビットがハイの場合、VCOMP ビットがハイの場合に COMP1 によるウェイクアップが可能となるように、CH ビットを使用して ADC スイッチマトリックスを介して複数の I/O をまとめて接続します。

8.4.3 RI アナログスイッチ制御レジスタ 2 (RI_ASCR2)

RI_ASCR2 レジスタは、ADC にリンクされていない I/O グループのアナログスイッチの設定に使用します。こうすると、事前に定義された I/O グループをまとめて接続することができます。

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
予約済み			GR6-4	GR6-3	CH12b GR7-7	CH11b GR7-6	CH10bG R7-5	CH9b GR2-5	CH8b GR2-4	CH7b GR2-3	CH6b GR9-4	CH3b GR9-3	CH2b GR3-5	CH1b GR3-4	CH0b GR3-3
			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
予約済み			GR4-3	GR4-2	GR4-1	GR5-3	GR5-2	GR5-1	GR6-2	GR6-1	GR10-4	GR10-3	GR10-2	GR10-1	
			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

ビット 31:29 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 28:16 **GRx-x** : GRx-x I/O アナログスイッチ制御

これらのビットは、ADC インタフェースとは別に I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。表 40 : I/O グループと選択 (202 ページ) を参照してください。

0 : アナログスイッチを開きます、または ADC インタフェースで制御します。

1 : アナログスイッチを閉じます。

注 : これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

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 11:0 **GRx-x** : GRx-x I/O アナログスイッチ制御

これらのビットは、ADC インタフェースとは別に I/O アナログスイッチを制御するために、ソフトウェアによってセット/クリアされます。表 40 : I/O グループと選択 (202 ページ) を参照してください。

0 : アナログスイッチを開きます、または ADC インタフェースで制御します。

1 : アナログスイッチを閉じます。

8.4.4 RI ヒステリシス制御レジスタ（RI_HYSCR1）

RI_HYSCR1 レジスタは、ポート A および B の入力シュミットトリガのヒステリシスを有効/無効にするために使用します。

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
PB[15:0]															
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
PA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **PB[15:0]**：ポート B ヒステリシス制御のオン/オフ

これらのビットは、ポート B[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

0：ヒステリシスオン

1：ヒステリシスオフ

ビット 15:0 **PA[15:0]**：ポート A ヒステリシス制御のオン/オフ

これらのビットは、ポート A[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

0：ヒステリシスオン

1：ヒステリシスオフ

8.4.5 RI ヒステリシス制御レジスタ（RI_HYSCR2）

RI_HYSCR2 レジスタでは、ポート C および D の入力シュミットトリガのヒステリシスを有効/無効にすることができます。

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
PD[15:0]															
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
PC[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **PD[15:0]** : ポート D ヒステリシス制御のオン/オフ

これらのビットは、ポート D[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : ヒステリシスオン
- 1 : ヒステリシスオフ

ビット 15:0 **PC[15:0]** : ポート C ヒステリシス制御のオン/オフ

これらのビットは、ポート C[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : ヒステリシスオン
- 1 : ヒステリシスオフ

8.4.6 RI ヒステリシス制御レジスタ (RI_HYSCR3)

RI_HYSCR3 レジスタは、ポート E および F 全体の入力シュミットトリガのヒステリシスを有効/無効にするために使用します。

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
PF[15:0]															
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
PE[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:16 **PF[15:0]** : ポート F ヒステリシス制御のオン/オフ

これらのビットは、ポート F[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : ヒステリシスオン
- 1 : ヒステリシスオフ

注 : これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 15:0 **PE[15:0]** : ポート E ヒステリシス制御のオン/オフ

これらのビットは、ポート E[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

- 0 : ヒステリシスオン
- 1 : ヒステリシスオフ



8.4.7 RI ヒステリシス制御レジスタ（RI_HYSCR4）

RI_HYSCR4 レジスタは、ポート G 全体の入力シュミットトリガのヒステリシスを有効/無効にするために使用します。

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PG[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 **PG[15:0]**：ポート G ヒステリシス制御のオン/オフ

これらのビットは、ポート G[15:0] のシュミットトリガヒステリシスを制御するために、ソフトウェアによってセット/クリアされます。

0：ヒステリシスオン

1：ヒステリシスオフ

注： これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

8.4.8 アナログスイッチモードレジスタ（RI_ASMR1）

RI_ASMR1 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、ポート A のアナログスイッチをタイマ OC で制御するか、ADC インタフェースまたは RI_ASCRx レジスタを介して制御するかを選択します。

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
予約済み															
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 31:16 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 **PA[15:0]**：ポート A アナログスイッチモード選択

これらのビットは、ポート A のアナログスイッチの制御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：ADC インタフェースまたは RI_ASCRx で制御します。

1：タイマで制御します。

8.4.9 チャネルマスキングレジスタ (RI_CM1)

RI_CM1 は、Cat.3、Cat.4、Cat.5 製品のみで利用でき、タイマ入力キャプチャとして指定されたポート A チャネルをマスクします (複数回の検出をトリガしないように取得完了後)。

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
予約済み															
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 31:16 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 PA[15:0] : ポート A チャネルのマスキング

これらのビットは、容量センシング取得中にポート A の入力をマスクするために、ソフトウェアによってセット/クリアされます。

0 : マスクされています。

1 : マスクされていません。

8.4.10 キャプチャレジスタ (RI_CICR1) のチャネル識別

RI_CICR1 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、アナログスイッチがタイマ OC によって制御されるときに使用します。RI_CICR1 では、タイマ入力キャプチャに対してチャネルを識別することができます。

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
予約済み															
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 31:16 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 PA[15:0] : キャプチャに対するポート A チャネルの識別

これらのビットは、ポート A のサンプリングコンデンサ I/O を識別するために、ソフトウェアによってセット/クリアされます。

0 : チャネル I/O

1 : サンプリングコンデンサ I/O



8.4.11 アナログスイッチモードレジスタ (RI_ASMR2)

RI_ASMR2 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、ポート B のアナログスイッチをタイマ OC で制御するか、ADC インタフェースまたは RI_ASCRx レジスタを介して制御するかを選択します。

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PB[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 **PB[15:0]** : ポート B アナログスイッチモード選択

これらのビットは、ポート B のアナログスイッチの制御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 : ADC インタフェースまたは RI_ASCRx で制御します。

1 : タイマで制御します。

8.4.12 チャネルマスクレジスタ (RI_CMR2)

RI_CMR2 は、Cat.3、Cat.4、Cat.5 製品のみで利用でき、タイマ入力キャプチャとして指定されたポート B チャネルをマスクします (複数回の検出をトリガしないように取得完了後)。

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PB[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 **PB[15:0]** : ポート B チャネルのマスク

これらのビットは、容量センシング取得中にポート B の入力をマスクするために、ソフトウェアによってセット/クリアされます。

0 : マスクされています。

1 : マスクされていません。

8.4.13 キャプチャレジスタ (RI_CICR2) のチャンネル識別

RI_CICR2 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、アナログスイッチがタイマ OC によって制御されるときに使用します。RI_CICR2 では、タイマ入力キャプチャに対してポート B チャンネルを識別することができます。

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PB[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 **PB[15:0]** : キャプチャに対するポート B チャンネルの識別

これらのビットは、ポート B のサンプリングコンデンサ I/O を識別するために、ソフトウェアによってセット/クリアされます。

0 : チャンネル I/O

1 : サンプリングコンデンサ I/O

8.4.14 アナログスイッチモードレジスタ (RI_ASMR3)

RI_ASMR3 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、ポート C のアナログスイッチをタイマ OC で制御するか、ADC インタフェースまたは RI_ASCRx レジスタを介して制御するかを選択します。

アドレスオフセット : 0x38

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
PC[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 **PC[15:0]** : ポート C アナログスイッチモード選択

これらのビットは、ポート C のアナログスイッチの制御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 : ADC インタフェースまたは RI_ASCRx で制御します。

1 : タイマで制御します。



8.4.15 チャネルマスクレジスタ（RI_CM3）

RI_CM3 は、Cat.3、Cat.4、Cat.5 製品のみで利用でき、タイマ入力キャプチャとして指定されたポート C チャンネルをマスクします（複数回の検出をトリガしないように取得完了後）。

アドレスオフセット：0x3C

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
PC[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 **PC[15:0]**：ポート C チャンネルのマスクング

これらのビットは、容量センシング取得中にポート C の入力をマスクするために、ソフトウェアによってセット/クリアされます。

0：マスクされています。

1：マスクされていません。

8.4.16 キャプチャレジスタ（RI_CICR3）のチャンネル識別

RI_CICR3 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、アナログスイッチがタイマ OC によって制御されるときに使用します。RI_CICR3 では、タイマ入力キャプチャに対してポート C チャンネルを識別することができます。

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PC[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 **PC[15:0]**：キャプチャに対するポート C チャンネルの識別

これらのビットは、ポート C のサンプリングコンデンサ I/O を識別するために、ソフトウェアによってセット/クリアされます。

0：チャンネル I/O

1：サンプリングコンデンサ I/O

8.4.17 アナログスイッチモードレジスタ (RI_ASMR4)

RI_ASMR4 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、ポート F のアナログスイッチをタイマ OC で制御するか、ADC インタフェースまたは RI_ASCRx レジスタを介して制御するかを選択します。

アドレスオフセット : 0x44

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
PF[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 **PF[15:0]** : ポート F アナログスイッチモード選択

これらのビットは、ポート F のアナログスイッチの制御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 : ADC インタフェースまたは RI_ASCRx で制御します。

1 : タイマで制御します。

注: これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

8.4.18 チャネルマスクレジスタ (RI_CMR4)

RI_CMR4 は、Cat.3、Cat.4、Cat.5 製品のみで利用でき、タイマ入力キャプチャとして指定されたポート F チャネルをマスクします (複数回の検出をトリガしないように取得完了後)。

アドレスオフセット : 0x48

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
PF[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 **PF[15:0]** : ポート F チャネルのマスクング

これらのビットは、容量センシング取得中にポート F の入力をマスクするために、ソフトウェアによってセット/クリアされます。

0 : マスクされています。

1 : マスクされていません。

注: これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。



8.4.19 キャプチャレジスタ (RI_CICR4) のチャネル識別

RI_CICR4 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、アナログスイッチがタイマ OC によって制御されるときに使用します。RI_CICR4 では、タイマ入力キャプチャに対してポート F チャネルを識別することができます。

アドレスオフセット : 0x4C

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
PF[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 **PF[15:0]** : キャプチャに対するポート F チャネルの識別

これらのビットは、ポート F のサンプリングコンデンサ I/O を識別するために、ソフトウェアによってセット/クリアされます。

0 : チャネル I/O

1 : サンプリングコンデンサ I/O

注 : これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

8.4.20 アナログスイッチモードレジスタ (RI_ASMR5)

RI_ASMR5 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、ポート G のアナログスイッチをタイマ OC で制御するか、ADC インタフェースまたは RI_ASCRx レジスタを介して制御するかを選択します。

アドレスオフセット : 0x50

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
PG[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 **PG[15:0]** : ポート G アナログスイッチモード選択

これらのビットは、ポート G のアナログスイッチの制御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 : ADC インタフェースまたは RI_ASCRx で制御します。

1 : タイマで制御します。

注 : これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

8.4.21 チャネルマスクレジスタ (RI_CM5)

RI_CM5 は、Cat.3、Cat.4、Cat.5 製品のみで利用でき、タイマ入力キャプチャとして指定されたポート G チャンネルをマスクします（複数回の検出をトリガしないように取得完了後）。

アドレスオフセット：0x54

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
PG[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 **PG[15:0]**：ポート G チャンネルのマスキング

これらのビットは、容量センシング取得中にポート G の入力をマスクするために、ソフトウェアによってセット/クリアされます。

0：マスクされています。

1：マスクされていません。

注： これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

8.4.22 キャプチャレジスタ (RI_CICR5) のチャンネル識別

RI_CICR5 レジスタは、Cat.3、Cat.4、Cat.5 製品でのみ利用でき、アナログスイッチがタイマ OC によって制御されるときに使用します。RI_CICR5 では、タイマ入力キャプチャに対してポート G チャンネルを識別することができます。

アドレスオフセット：0x58

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
PG[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 **PG[15:0]**：キャプチャに対するポート G チャンネルの識別

これらのビットは、ポート G のサンプリングコンデンサ I/O を識別するために、ソフトウェアによってセット/クリアされます。

0：チャンネル I/O

1：サンプリングコンデンサ I/O

注： これらのビットは、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。



8.4.23 RI レジスタマップ

表 44に RI レジスタの要約を示します。

表 44. RI レジスタマップとリセット値

[illegible]

表 44. RI 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	RI_CICR3	予約済み																PC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x44	RI_ASMR4	予約済み																PF[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x48	RI_CMR4	予約済み																PF[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x4C	RI_CICR4	予約済み																PF[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x50	RI_ASMR5	予約済み																PG[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x54	RI_CMR5	予約済み																PG[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x58	RI_CICR5	予約済み																PG[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

8.5 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。

8.5.1 SYSCFG メモリ再配置レジスタ (SYSCFG_MEMRMP)

このレジスタは、メモリ再配置時の特定の設定に使用されます。

- アドレス 0x0000 0000 でアクセス可能なメモリのタイプの設定には、2 つのビットを使用します。これらのビットを用いて、ソフトウェアによる物理的な再配置を選択しますので、BOOT ピンはバイパスされます。
- リセット後、これらのビットは、BOOT ピンで選択された値になります。

注： このレジスタは、RCC_APB2RSTR レジスタの SYSCFGRST ビットを通じてリセットされません。

アドレスオフセット：0x00

リセット値：0x0000 00XX (X は BOOT ピンが選択したメモリモード)

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						BOOT_MODE		予約済み						MEM_MODE	
						r	r							rw	rw

ビット 31:10 予約済み

ビット 9:8 **BOOT_MODE**：ブートピンが選択したブートモード

これらのビットは読み出し専用です。ブートピンが選択したブートモードを示します。ビット 9 は、BOOT1 ピン上でサンプリングされる値に対応し、ビット 8 は BOOT0 ピン上でサンプリングされる値に対応します。 [セクション 2.7：ブート設定 \(51 ページ\)](#) も参照してください。

- 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：エンベデッド SRAM ブートモード

ビット 7:2 予約済み

ビット 1:0 **MEM_MODE**：メモリマッピング選択

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、アドレス 0x0000 0000 におけるメモリの内部マッピングを制御します。リセット後、これらのビットは BOOT ピンで選択したメモリマッピングを使用します。

- 00：0x0000 0000 にメインフラッシュメモリがマッピングされます。
- 01：0x0000 0000 にシステムフラッシュメモリがマッピングされます。
- 10：FSMC
- 11：0x0000 0000 に SRAM がマッピングされます。

8.5.2 SYSCFG ペリフェラルモード設定レジスタ (SYSCFG_PMC)

USB データ+ (DP) ライン上で、ソフトウェアによって内部プルアップ抵抗 (1.5 kW) を接続できます。この内部プルアップ抵抗は、USB がパワーダウンモードになっておらず、USB_PU ビットがセットされていれば有効になります。

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										LCD_CAPA			USB_PU		
										rw			rw		

ビット 31:6 予約済み

ビット 5:1 **LCD_CAPA** デカップリングコンデンサ接続 (このデバイスの機能については、デバイスのデータシートを参照)

ビット 1 は PB2/LCD_VCAP2 で V_{LCDrail2} の接続を制御します。

0 : V_{LCDrail2} は PB2/LCD_VCAP2 に接続されていません。

1 : V_{LCDrail2} は PB2/LCD_VCAP2 に接続されています。

ビット 2 は PB12/LCD_VCAP1 で V_{LCDrail1} の接続を制御します。

ビット 3 は PB0/LCD_VCAP3 で V_{LCDrail3} の接続を制御します。

ビット 4 は PE11/LCD_VCAP1 で V_{LCDrail1} の接続を制御します。

ビット 5 は PE12/LCD_VCAP3 で V_{LCDrail3} の接続を制御します。

ビット 0 **USB_PU** DP ラインでの USB プルアップ有効

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、USP DP ラインで内部プルアップ (1.5 kW) を制御します。

0 : USB DP ラインにプルアップはありません (USB がパワーダウンモードでない場合でも)

1 : USB DP ラインに内部プルアップが接続されています (USB がパワーダウンモードでない場合のみ)

8.5.3 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]			
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 = 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、ソフトウェアによって書き込まれます。0000 : PA[x] ピン
- 0001 : PB[x] ピン
- 0010 : PC[x] ピン
- 0011 : PD[x] ピン
- 0100 : PE[x] ピン
- 0101 : PH[x](PH[2:0] のみ、PH3 は利用不可能)
- 0110 : PF[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
- 0111 : PG[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)

8.5.4 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]			
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 = 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、ソフトウェアによって書き込まれます。

0000 : PA[x] ピン
0001 : PB[x] ピン
0010 : PC[x] ピン
0011 : PD[x] ピン
0100 : PE[x] ピン
0110 : PF[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
0111 : PG[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
PH[7:4] は利用できません。

8.5.5 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、ソフトウェアによって書き込まれます。

0000 : PA[x] ピン
0001 : PB[x] ピン
0010 : PC[x] ピン
0011 : PD[x] ピン
0100 : PE[x] ピン
0110 : PF[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
0111 : PG[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
PH[11:8] は利用できません。



8.5.6 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]			
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 = 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、ソフトウェアによって書き込まれます。

0000 : PA[x] ピン
0001 : PB[x] ピン
0010 : PC[x] ピン
0011 : PD[x] ピン
0100 : PE[x] ピン
0110 : PF[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
0111 : PG[x] ピン (Cat.3、Cat.4 および Cat.5 製品のみ)
PH[5:12] は利用できません。

8.5.7 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 45. 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_ MEMRMP	予約済み																					BOOT_MODE		予約済み												MEM_MODE	
	リセット値																																					
0x04	SYSCFG_PMC	予約済み																																		USB PU		
	リセット値																																					0
0x08	SYSCFG_ EXTICR1	予約済み																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	0	0	0	0	0	0	0
0x0C	SYSCFG_ EXTICR2	予約済み																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	0	0	0	0	0	0	0
0x10	SYSCFG_ EXTICR3	予約済み																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	0	0	0	0	0	0	0
0x14	SYSCFG_ EXTICR4	予約済み																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	0	0	0	0	0	0	0

表 5 : レジスタ境界アドレス (47 ページ) を参照してください。

9 タッチセンシング I/O

9.1 概要

バリュウラインを除くすべての STM32L151xx 製品は、あらゆるアプリケーションに容量センシング機能を追加するシンプルなソリューションを提供しています。容量センシング技術では、電極付近に指があるかどうかを検出することができ、電極は誘電体（ガラス、プラスチック）によって直接触れられないように保護されています。表面電荷移動アクイジション原理に基づく実証済みの実装を使用して、指（またはその他のあらゆる導電性物質）によって発生する容量変化を測定します。電極容量に帯電させた後、蓄積電荷の一部を、サンプリングコンデンサの電圧が指定しきい値に達するまでこのコンデンサに移動させることによって成り立っています。STM32L1xxxx では、このアクイジションを直接 GPIO、タイマ、アナログ I/O グループで管理します（[セクション 8：システム設定コントローラ \(SYSCFG\) とルーティングインタフェース \(RI\)](#) 参照）。

信頼性の高いタッチセンシングソリューションは、無料の STM32L1xx STMTouch ファームウェアライブラリを使用すれば迅速かつ簡単に実装できます。

9.2 主な特長

- 実績があり耐久性のある表面電荷移動アクイジション原理
- 電荷移動アクイジションシーケンスをソフトウェアモードまたはタイマモードの 2 つのモードで管理
- 最大 34 個の容量センシングチャネルに対応
- 最大 11 個の容量センシングチャネルを平行してアクイジションでき、非常に優れた応答時間を提供
- 1 つのサンプリングコンデンサで最大 4 つの容量センシングチャネルに対応し、システム BOM を軽減
- タッチキー、近接センサ、リニアタッチセンサ、ロータリータッチセンサに対応
- STM32L1xx STMTouch ファームウェアライブラリで動作するように設計

9.3 機能の説明

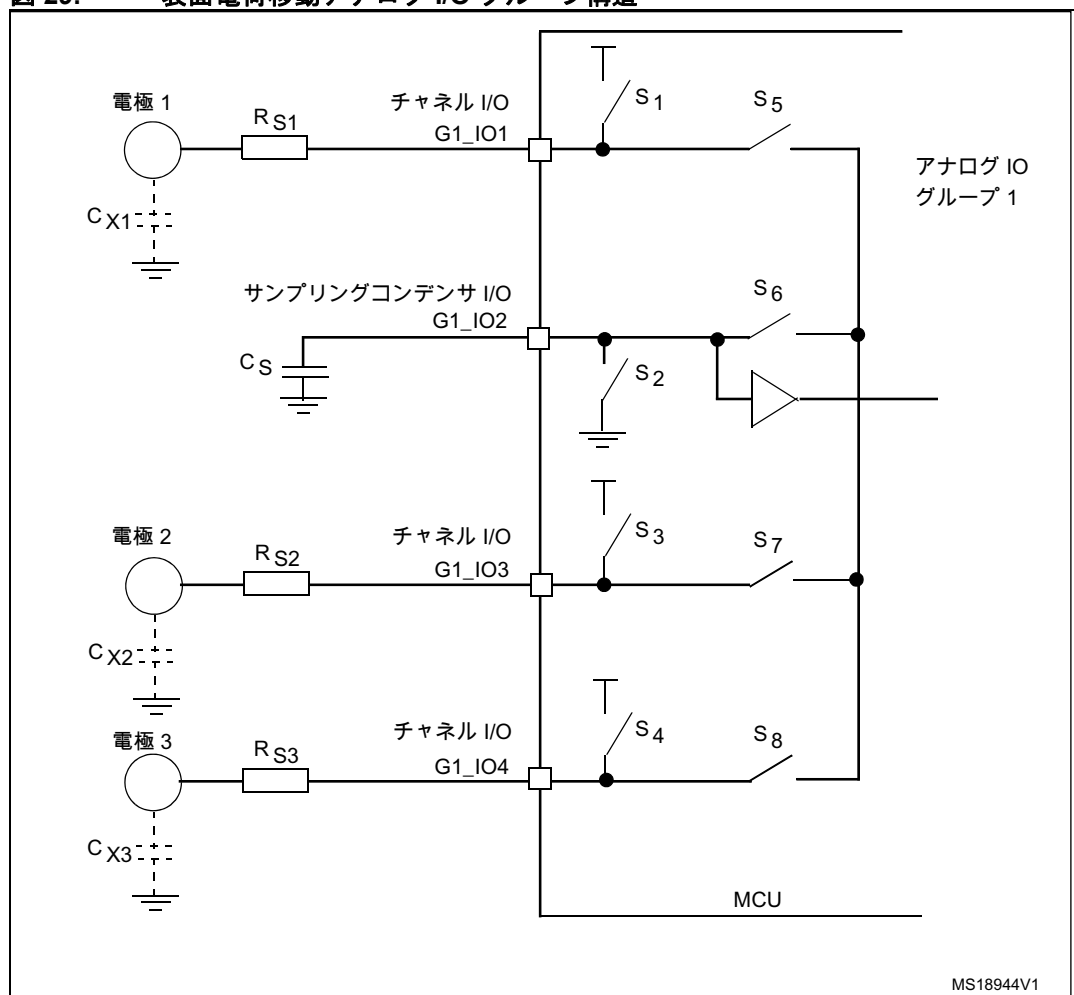
9.3.1 表面電荷移動アクイジションの概要

表面電荷移動アクイジションは、実績があり信頼性の高い効率的な容量測定方法です。最低限の外部コンポーネントを使用し、シングルエンドの電極タイプで動作します。このアクイジションは、最大 7 個の GPIO で構成されるアナログ I/O グループを中心に設計されています (図 29 参照)。本製品は、複数のアナログ I/O グループを提供して複数の容量センシングチャネルを同時にアクイジションし、多数のチャネルに対応できるようになっています。同じアナログ I/O グループ内では、容量センシングチャネルアクイジションはシーケンシャルになっています。

GPIO の 1 つがサンプリングコンデンサ C_S 専用となり、アナログ I/O グループ 1 つにつき、サンプリングコンデンサを一度に 1 つだけ有効にする必要があります。

残りの GPIO は電極専用となり、一般にチャネルと呼ばれます。

図 29. 表面電荷移動アナログ I/O グループ構造



注: Gx_IOy の x はアナログ I/O グループ番号、 y は選択したグループ内での GPIO 番号です。

一部の特定のニーズ（近接検出など）においては、アナログ I/O グループ 1 つにつき同時に 1 つ以上のチャネルを有効にすることができます。

容量センシング I/O の表については、[表 40: I/O グループと選択 \(202 ページ\)](#) を参照してください。

表面電荷移動アクイジション原理では、電極容量 (C_X) を帯電させ、蓄積電荷の一部をサンプリングコンデンサ (C_S) に移動します。このシーケンスを、 C_S の電圧が所与のしきい値 (通常は V_{IH}) に達するまで繰り返します。しきい値に達するまでに必要な電荷移動回数が、直接電極容量の大きさを表します。

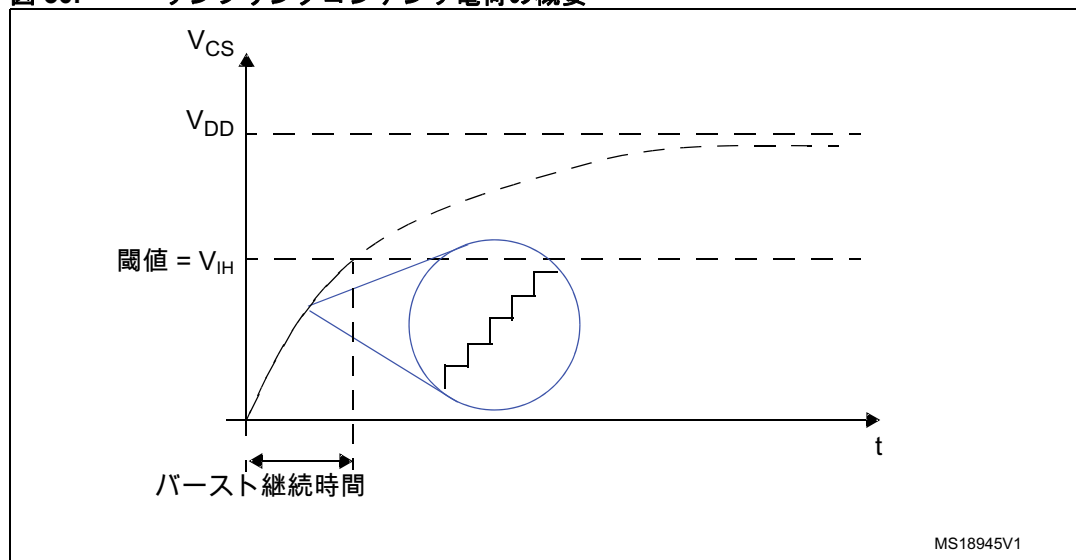
[表 46](#) は、容量センシングチャネル 1 のアクイジションシーケンスを詳述しています。 C_S の電圧が所定のしきい値に達するまで 3 から 7 の状態が繰り返されます。同じシーケンスが、他のチャネルのアクイジションにも適用されます。電極の直列抵抗 R_S により、製品の ESD 耐性を改善できます。

表 46. アクイジションスイッチングシーケンスの概要

状態	S1	S2	S3	S4	S5	S6	S7	S8	状態の説明
#1	開	閉	開	開	閉	閉	閉	閉	すべての C_X と C_S を放電
#2	開	開	開	開	開	開	開	開	デッドタイム
#3	閉	開	開	開	開	開	開	開	C_{X1} を充電
#4	開	開	開	開	開	開	開	開	デッドタイム
#5	開	開	開	開	閉	閉	開	開	C_{X1} から C_S への電荷移動
#6	開	開	開	開	開	開	開	開	デッドタイム
#7	開	開	開	開	開	開	開	開	C_S の電圧を測定

サンプリングコンデンサ C_S の時間経過による電圧の変化を以下に詳述します。

図 30. サンプリングコンデンサ電荷の概要



9.3.2 電荷移動アクイジションの管理

STM32L1xxxx 製品では、アクイジションを 2 つのモードで管理できます。

- **ソフトウェアモード**：GPIO ポートの切り替えやパルス回数の計測は、すべて CPU が処理します。
- **タイマモード**：GPIO ポートの切り替えやパルス回数の計測は、タイマが処理します（Cat.3、Cat.4、Cat.5 製品でのみ）。

ソフトウェアモードアクイジション

このモードでは、次の 2 つのペリフェラルを使い、以下に従ってチャンネルとサンプリングコンデンサの I/O を制御します。

- 汎用 I/O（[セクション 7：汎用 I/O \(GPIO\)](#) 参照）
- ルーティングインタフェース（[セクション 8：システム設定コントローラ \(SYSCFG\) と ルーティングインタフェース \(RI\) \(197 ページ\)](#) 参照）

表 47. チャンネルとサンプリングコンデンサ I/O の設定概要

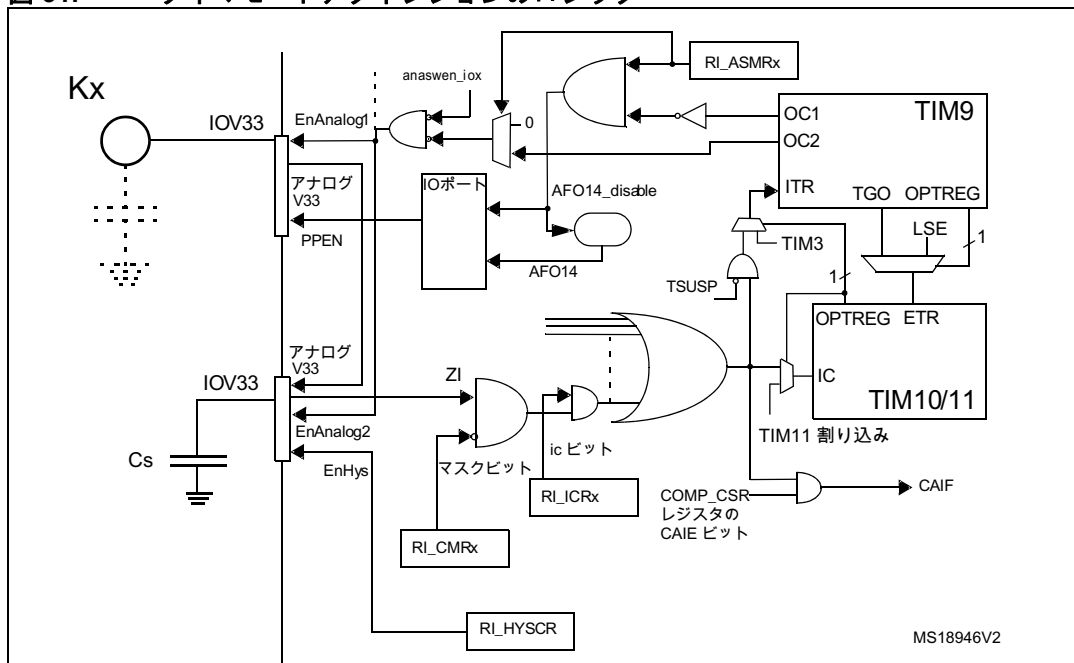
状態	チャンネル I/O 設定	サンプリングコンデンサ I/O 設定
#1	出力プッシュプルロー	出力プッシュプルロー
#2	入力フローティング	ヒステリシスを無効にしての入力フローティング
#3	出力プッシュプルハイ	
#4	入力フローティング	
#5	アナログスイッチを閉じての入力フローティング	ヒステリシスを無効にし、アナログスイッチを閉じての入力フローティング
#6	入力フローティング	ヒステリシスを無効にしての入力フローティング
#7		

タイマモードアクイジション

このモードでは、次のペリフェラルの使用が必要になります。

- 汎用 I/O (セクション 7 (175 ページ) 参照)
- ルーティングインタフェース (セクション 8 (197 ページ) 参照)
- 汎用タイマ 9 (セクション 18 (455 ページ) 参照)
- 汎用タイマ 10 または 11 (セクション 18 (455 ページ) 参照)

図 31. タイマモードアクイジションのロジック



どちらのタイマも GPIO ポート切り替えの管理に使用されており、CPU の負荷を大きく軽減します。TIM9 は、直接チャネルおよびサンプリングコンデンサ I/O を制御して電荷移動シーケンスの生成を処理します。TIM10 または 11 は、有効になっているアナログ I/O グループの 1 つでアクイジション終了が検出される前に生成された電荷移動回数を計測します。

ソフトウェアシーケンスを使用すると、さまざまなペリフェラルを設定してアクイジションを実施することができます。

設定ステップ：

1. 容量センシングチャネル I/O がアキュイジションされるようにオルタネートプッシュプル出力モードで I/O ポートを設定します。各グループにつき一度に有効にしなければならないチャネルは 1 つだけですので注意してください。
2. ヒステリシスを無効にした状態で、フローティング入力モードでサンプリングコンデンサ I/O を設定します。
3. センターアラインモードで TIM9 タイマを設定し、OC1 および OC2 で PWM 信号を生成します。PWM 信号の通常周波数は 250 KHz です。
4. TIM9 が生成したクロック信号を使用して、スレーブモードで TIM10 または 11 を設定します。さらに、IC1 を有効にしてアキュイジション終了検出時にカウンタ値をキャプチャする必要があります。オプションで割り込み生成を有効にすることができます。
5. TIM10 または 11 を有効にします。
6. RI_ASMRx レジスタを使って TIM9 でチャネルのアナログスイッチおよびサンプリングコンデンサ I/O の制御を有効にします（[セクション 8.4 : RI レジスタ \(207 ページ\)](#) 参照）。
7. レジスタ RI_CICRx を使用してサンプリングコンデンサ I/O を識別します。
8. TIM9 を有効にしてアキュイジションを開始します。

測定ステップ：

- 1 つのサンプリングコンデンサでアキュイジションが終了したら、次の操作を行います。
9. 割り込み生成が有効になっている場合は、TIM10 または 11 のキャプチャ割り込みフラグをクリアします。
10. IC レジスタを読み出し、その内容を RAM に保存します。
11. RI_CMRx および GPIOx_IDR レジスタ間で XOR 演算を実施し、どの新しいチャネルがアキュイジション終了をトリガしたのかを判断し、カウンタ値で対応するチャネル情報を更新します。
12. RI_CMRx レジスタを更新し、既に検出されたチャネルでのそれ以降のアキュイジション終了検出をマスクします。これで TIM9 が自動的に再起動します。

9.4 タッチセンシングライブラリ

STM32L1xxxx デバイスを用いたタッチセンシングソリューションの開発を容易にするため、STMicroelectronics では信頼性の高い完全な C ソースコードソリューションが入手できる STM32L1xx STMTouch センシングライブラリを提供しています。このファームウェアライブラリは、www.st.com より無料ダウンロードとして利用可能です。STM32L1xx STMTouch センシングライブラリのダウンロード方法についての詳細は、担当の ST 担当者にお問い合わせください。

10 割り込みとイベント

このセクションは、特に指定がない限り、STM32L1xxx ファミリ全体に適用されます。

10.1 ネスト化されたベクタ割り込みコントローラ (NVIC)

主な機能

- Cat.1Cat.2 デバイス (表 48 参照) でマスク可能な 45 の割り込みチャンネル、Cat.3 デバイス (表 49 参照) でマスク可能な 54 の割り込みチャンネル、Cat.4Cat.5 デバイス (表 50 参照) で 57 のチャンネル。これには Cortex[®]-M3 の 16 本の割り込みラインは含まれません。
- 16 段階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を実装

NVIC とプロセッサコアのインタフェースが密結合され、割り込み処理の遅延時間を短くし、遅れて到着した割り込みを効率的に処理できます。

コアの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、PM0056 プログラミングマニュアルを参照してください。

10.1.1 SysTick 較正值レジスタ

SysTick 較正值は 4000 に固定され、SysTick クロックを 4 MHz (最大 HCLK/8) に設定した状態で 1 ms という基準タイムベースを与えます。

10.1.2 割り込みベクタおよび例外ベクタ

表 48 は STM32L1xxx デバイスのベクタテーブルです。

表 48. ベクタテーブル (Cat.1 および Cat.2 デバイス)

位置	優先順位	優先順位 設定種別	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	すべての種類の障害	0x0000_000C
-	0	設定可能	MemManage_Handler	メモリ管理	0x0000_0010
-	1	設定可能	BusFault_Handler	プリフェッチ障害、メモリアクセス障害	0x0000_0014
-	2	設定可能	UsageFault_Handler	未定義命令または不正な状態	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
-	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	TAMPER_STAMP	タンパおよびタイムスタンプ割り込み (EXTI ライン経由)	0x0000_0048
3	10	設定可能	RTC_WKUP	RTC ウェイクアップ割り込み (EXTI ライン経由)	0x0000_004C
4	11	設定可能	FLASH	フラッシュグローバル割り込み	0x0000_0050
5	12	設定可能	RCC	RCC グローバル割り込み	0x0000_0054
6	13	設定可能	EXTI0	EXTI ライン 0 割り込み	0x0000_0058
7	14	設定可能	EXTI1	EXTI ライン 1 割り込み	0x0000_005C
8	15	設定可能	EXTI2	EXTI ライン 2 割り込み	0x0000_0060
9	16	設定可能	EXTI3	EXTI ライン 3 割り込み	0x0000_0064
10	17	設定可能	EXTI4	EXTI ライン 4 割り込み	0x0000_0068
11	18	設定可能	DMA1_Channel1	DMA チャンネル 1 グローバル割り込み	0x0000_006C
12	19	設定可能	DMA1_Channel2	DMA1 チャンネル 2 グローバル割り込み	0x0000_0070
13	20	設定可能	DMA1_Channel3	DMA1 チャンネル 3 グローバル割り込み	0x0000_0074
14	21	設定可能	DMA1_Channel4	DMA1 チャンネル 4 グローバル割り込み	0x0000_0078
15	22	設定可能	DMA1_Channel5	DMA1 チャンネル 5 グローバル割り込み	0x0000_007C

表 48. ベクタテーブル (Cat.1 およびCat.2 デバイス) (続き)

位置	優先順位	優先順位 設定種別	略称	説明	アドレス
16	23	設定可能	DMA1_Channel6	DMA1 チャンネル 6 グローバル割り込み	0x0000_0080
17	24	設定可能	DMA1_Channel7	DMA1 チャンネル 7 グローバル割り込み	0x0000_0084
18	25	設定可能	ADC1	ADC1 グローバル割り込み	0x0000_0088
19	26	設定可能	USB_HP	USB 高優先順位割り込み	0x0000_008C
20	27	設定可能	USB_LP	USB 低優先順位割り込み	0x0000_0090
21	28	設定可能	DAC	DAC 割り込み	0x0000_0094
22	29	設定可能	COMP、TSC ⁽¹⁾	EXTI ライン (21 と 22) 割り込み、タッチセンス割り込み ⁽¹⁾ を経由するコンパレータウェイクアップ	0x0000_0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割り込み	0x0000_009C
24	31	設定可能	LCD	LCD グローバル割り込み	0x0000_00A0
25	32	設定可能	TIM9	TIM9 グローバル割り込み	0x0000_00A4
26	33	設定可能	TIM10	TIM10 グローバル割り込み	0x0000_00A8
27	34	設定可能	TIM11	TIM11 グローバル割り込み	0x0000_00AC
28	35	設定可能	TIM2	TIM2 グローバル割り込み	0x0000_00B0
29	36	設定可能	TIM3	TIM3 グローバル割り込み	0x0000_00B4
30	37	設定可能	TIM4	TIM4 グローバル割り込み	0x0000_00B8
31	38	設定可能	I2C1_EV	I ² C1 イベント割り込み	0x0000_00BC
32	39	設定可能	I2C1_ER	I ² C1 エラー割り込み	0x0000_00C0
33	40	設定可能	I2C2_EV	I ² C2 イベント割り込み	0x0000_00C4
34	41	設定可能	I2C2_ER	I ² C2 エラー割り込み	0x0000_00C8
35	42	設定可能	SPI1	SPI1 グローバル割り込み	0x0000_00CC
36	43	設定可能	SPI2	SPI2 グローバル割り込み	0x0000_00D0
37	44	設定可能	USART1	USART1 グローバル割り込み	0x0000_00D4
38	45	設定可能	USART2	USART2 グローバル割り込み	0x0000_00D8
39	46	設定可能	USART3	USART3 グローバル割り込み	0x0000_00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割り込み	0x0000_00E0
41	48	設定可能	RTC_Alarm	EXTI ライン割り込みを経由する RTC アラーム (A および B)	0x0000_00E4
42	49	設定可能	USB_FS_WKUP	EXTI ライン割り込みを経由する USB デバイス FS ウェイクアップ	0x0000_00E8
43	50	設定可能	TIM6	TIM6 グローバル割り込み	0x0000_00EC
44	51	設定可能	TIM7	TIM7 グローバル割り込み	0x0000_00F0

1. タッチセンス割り込みは Cat.2 デバイスのみ。

表 49. ベクタテーブル (Cat.3 デバイス)

位置	優先順位	優先順位 設定種別	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	ノンマスカブル割り込み。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	すべての種類の障害	0x0000_000C
-	0	設定可能	MemManage_Handler	メモリ管理	0x0000_0010
-	1	設定可能	BusFault_Handler	プリフェッチ障害、メモリアクセス障害	0x0000_0014
-	2	設定可能	UsageFault_Handler	未定義命令または不正な状態	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
-	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	EXTI ライン 16 検出割り込みを経由する PVD	0x0000_0044
2	9	設定可能	TAMPER_STAMP	EXTI ライン 19 割り込みを経由するタン パ、LSECSS、TimeStamp	0x0000_0048
3	10	設定可能	RTC_WKUP	RTC ウェイクアップ 経由する EXTI ラ イン 20 割り込み	0x0000_004C
4	11	設定可能	FLASH	フラッシュグローバル割り込み	0x0000_0050
5	12	設定可能	RCC	RCC グローバル割り込み	0x0000_0054
6	13	設定可能	EXTI0	EXTI ライン 0 割り込み	0x0000_0058
7	14	設定可能	EXTI1	EXTI ライン 1 割り込み	0x0000_005C
8	15	設定可能	EXTI2	EXTI ライン 2 割り込み	0x0000_0060
9	16	設定可能	EXTI3	EXTI ライン 3 割り込み	0x0000_0064
10	17	設定可能	EXTI4	EXTI ライン 4 割り込み	0x0000_0068
11	18	設定可能	DMA1_Channel1	DMA チャンネル 1 グローバル割り込み	0x0000_006C
12	19	設定可能	DMA1_Channel2	DMA1 チャンネル 2 グローバル割り込み	0x0000_0070
13	20	設定可能	DMA1_Channel3	DMA1 チャンネル 3 グローバル割り込み	0x0000_0074
14	21	設定可能	DMA1_Channel4	DMA1 チャンネル 4 グローバル割り込み	0x0000_0078
15	22	設定可能	DMA1_Channel5	DMA1 チャンネル 5 グローバル割り込み	0x0000_007C
16	23	設定可能	DMA1_Channel6	DMA1 チャンネル 6 グローバル割り込み	0x0000_0080
17	24	設定可能	DMA1_Channel7	DMA1 チャンネル 7 グローバル割り込み	0x0000_0084

表 49. ベクタテーブル (Cat.3 デバイス) (続き)

位置	優先順位	優先順位 設定種別	略称	説明	アドレス
18	25	設定可能	ADC1	ADC1 グローバル割り込み	0x0000_0088
19	26	設定可能	USB_HP	USB 高優先順位割り込み	0x0000_008C
20	27	設定可能	USB_LP	USB 低優先順位割り込み	0x0000_0090
21	28	設定可能	DAC	DAC 割り込み	0x0000_0094
22	29	設定可能	COMP/CA	EXTI ライン (21 および 22) 割り込み/ チャネルアクイジション割り込みを経由 するコンパレータウェイクアップ	0x0000_0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割り込み	0x0000_009C
24	31	設定可能	LCD	LCD グローバル割り込み	0x0000_00A0
25	32	設定可能	TIM9	TIM10 グローバル割り込み	0x0000_00A4
26	33	設定可能	TIM10	TIM10 グローバル割り込み	0x0000_00A8
27	34	設定可能	TIM11	TIM11 グローバル割り込み	0x0000_00AC
28	35	設定可能	TIM2	TIM2 グローバル割り込み	0x0000_00B0
29	36	設定可能	TIM3	TIM3 グローバル割り込み	0x0000_00B4
30	37	設定可能	TIM4	TIM4 グローバル割り込み	0x0000_00B8
31	38	設定可能	I2C1_EV	I ² C1 イベント割り込み	0x0000_00BC
32	39	設定可能	I2C1_ER	I ² C1 エラー割り込み	0x0000_00C0
33	40	設定可能	I2C2_EV	I ² C2 イベント割り込み	0x0000_00C4
34	41	設定可能	I2C2_ER	I ² C2 エラー割り込み	0x0000_00C8
35	42	設定可能	SPI1	SPI1 グローバル割り込み	0x0000_00CC
36	43	設定可能	SPI2	SPI2 グローバル割り込み	0x0000_00D0
37	44	設定可能	USART1	USART1 グローバル割り込み	0x0000_00D4
38	45	設定可能	USART2	USART2 グローバル割り込み	0x0000_00D8
39	46	設定可能	USART3	USART3 グローバル割り込み	0x0000_00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割り込み	0x0000_00E0
41	48	設定可能	RTC_Alarm	EXTI ライン 17 割り込みを経由する RTC アラーム (A および B)	0x0000_00E4
42	49	設定可能	USB_FS_WKUP	EXTI ライン 18 割り込みを経由する USB デバイス FS ウェイクアップ	0x0000_00E8
43	50	設定可能	TIM6	TIM6 グローバル割り込み	0x0000_00EC
44	51	設定可能	TIM7	TIM7 グローバル割り込み	0x0000_00F0
45	53	設定可能	TIM5	TIM5 グローバル割り込み	0x0000_00F8
46	54	設定可能	SPI3	SPI3 グローバル割り込み	0x0000_00FC
47	57	設定可能	DMA2_CH1	DMA2 チャンネル 1 割り込み	0x0000_0108
48	58	設定可能	DMA2_CH2	DMA2 チャンネル 2 割り込み	0x0000_010C
49	59	設定可能	DMA2_CH3	DMA2 チャンネル 3 割り込み	0x0000_0110
50	60	設定可能	DMA2_CH4	DMA2 チャンネル 4 割り込み	0x0000_0114
51	61	設定可能	DMA2_CH5	DMA2 チャンネル 5 割り込み	0x0000_0118



表 49. ベクタテーブル (Cat.3 デバイス) (続き)

位置	優先順位	優先順位 設定種別	略称	説明	アドレス
52	62	設定可能	AES	AES グローバル割り込み	0x0000_011C
53	63	設定可能	COMP_ACQ	コンパレータチャネルアキュイジション割り込み	0x0000_0120

表 50. ベクタテーブル (Cat.4およびCat.5 デバイス)

位置	優先順位	優先順位 設定種別	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	すべての異常	0x0000_000C
-	0	設定可能	MemManage_Handler	メモリ管理	0x0000_0010
-	1	設定可能	BusFault_Handler	プリフェッチ障害、メモリアクセス障害	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
-	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	EXTI ライン 16 検出割り込みを経由する PVD	0x0000_0044
2	9	設定可能	TAMPER_STAMP	EXTI ライン 19 割り込みを経由するタンパ、LSECSS、TimeStamp	0x0000_0048
3	10	設定可能	RTC_WKUP	EXTI ライン 20 割り込みを経由する RTC ウェイクアップ	0x0000_004C
4	11	設定可能	FLASH	フラッシュグローバル割り込み	0x0000_0050
5	12	設定可能	RCC	RCC グローバル割り込み	0x0000_0054
6	13	設定可能	EXTI0	EXTI ライン 0 割り込み	0x0000_0058
7	14	設定可能	EXTI1	EXTI ライン 1 割り込み	0x0000_005C
8	15	設定可能	EXTI2	EXTI ライン 2 割り込み	0x0000_0060
9	16	設定可能	EXTI3	EXTI ライン 3 割り込み	0x0000_0064
10	17	設定可能	EXTI4	EXTI ライン 4 割り込み	0x0000_0068
11	18	設定可能	DMA1_Channel1	DMA チャネル 1 グローバル割り込み	0x0000_006C

表 50. ベクタテーブル (Cat.4およびCat.5 デバイス) (続き)

位置	優先順位	優先順位 設定種別	略称	説明	アドレス
12	19	設定可能	DMA1_Channel2	DMA1 チャンネル 2 グローバル割り込み	0x0000_0070
13	20	設定可能	DMA1_Channel3	DMA1 チャンネル 3 グローバル割り込み	0x0000_0074
14	21	設定可能	DMA1_Channel4	DMA1 チャンネル 4 グローバル割り込み	0x0000_0078
15	22	設定可能	DMA1_Channel5	DMA1 チャンネル 5 グローバル割り込み	0x0000_007C
16	23	設定可能	DMA1_Channel6	DMA1 チャンネル 6 グローバル割り込み	0x0000_0080
17	24	設定可能	DMA1_Channel7	DMA1 チャンネル 7 グローバル割り込み	0x0000_0084
18	25	設定可能	ADC1	ADC1 グローバル割り込み	0x0000_0088
19	26	設定可能	USB_HP	USB 高優先順位割り込み	0x0000_008C
20	27	設定可能	USB_LP	USB 低優先順位割り込み	0x0000_0090
21	28	設定可能	DAC	DAC 割り込み	0x0000_0094
22	29	設定可能	COMP/CA	EXTI ライン (21 および 22) 割り込み/ チャンネルアキュイジション割り込みを経由 するコンパレータウェイクアップ	0x0000_0098
23	30	設定可能	EXTI9_5	EXTI ライン [9:5] 割り込み	0x0000_009C
24	31	設定可能	LCD	LCD グローバル割り込み	0x0000_00A0
25	32	設定可能	TIM9	TIM10 グローバル割り込み	0x0000_00A4
26	33	設定可能	TIM10	TIM10 グローバル割り込み	0x0000_00A8
27	34	設定可能	TIM11	TIM11 グローバル割り込み	0x0000_00AC
28	35	設定可能	TIM2	TIM2 グローバル割り込み	0x0000_00B0
29	36	設定可能	TIM3	TIM3 グローバル割り込み	0x0000_00B4
30	37	設定可能	TIM4	TIM4 グローバル割り込み	0x0000_00B8
31	38	設定可能	I2C1_EV	I ² C1 イベント割り込み	0x0000_00BC
32	39	設定可能	I2C1_ER	I ² C1 エラー割り込み	0x0000_00C0
33	40	設定可能	I2C2_EV	I ² C2 イベント割り込み	0x0000_00C4
34	41	設定可能	I2C2_ER	I ² C2 エラー割り込み	0x0000_00C8
35	42	設定可能	SPI1	SPI1 グローバル割り込み	0x0000_00CC
36	43	設定可能	SPI2	SPI2 グローバル割り込み	0x0000_00D0
37	44	設定可能	USART1	USART1 グローバル割り込み	0x0000_00D4
38	45	設定可能	USART2	USART2 グローバル割り込み	0x0000_00D8
39	46	設定可能	USART3	USART3 グローバル割り込み	0x0000_00DC
40	47	設定可能	EXTI15_10	EXTI ライン [15:10] 割り込み	0x0000_00E0
41	48	設定可能	RTC_Alarm	EXTI ライン 17 割り込みを経由する RTC アラーム (A および B)	0x0000_00E4
42	49	設定可能	USB_FS_WKUP	EXTI ライン 18 割り込みを経由する USB デバイス FS ウェイクアップ	0x0000_00E8
43	50	設定可能	TIM6	TIM6 グローバル割り込み	0x0000_00EC
44	51	設定可能	TIM7	TIM7 グローバル割り込み	0x0000_00F0
45	52	設定可能	SDIO	SDIO グローバル割り込み	0x0000_00F4



表 50. ベクタテーブル (Cat.4およびCat.5 デバイス) (続き)

位置	優先順位	優先順位 設定種別	略称	説明	アドレス
46	53	設定可能	TIM5	TIM5 グローバル割り込み	0x0000_00F8
47	54	設定可能	SPI3	SPI3 グローバル割り込み	0x0000_00FC
48	55	設定可能	UART4	UART4 グローバル割り込み	0x0000_0100
49	56	設定可能	UART5	UART5 グローバル割り込み	0x0000_0104
50	57	設定可能	DMA2_CH1	DMA2 チャンネル 1 割り込み	0x0000_0108
51	58	設定可能	DMA2_CH2	DMA2 チャンネル 2 割り込み	0x0000_010C
52	59	設定可能	DMA2_CH3	DMA2 チャンネル 3 割り込み	0x0000_0110
53	60	設定可能	DMA2_CH4	DMA2 チャンネル 4 割り込み	0x0000_0114
54	61	設定可能	DMA2_CH5	DMA2 チャンネル 5 割り込み	0x0000_0118
55	62	設定可能	AES	AES グローバル割り込み	0x0000_011C
56	63	設定可能	COMP_ACQ	コンパレータチャンネルアキュイジション割り込み	0x0000_0120

10.2 外部割り込み／イベントコントローラ (EXTI)

外部割り込み／イベントコントローラは、イベント／割り込みリクエストを発生するための最大 24 個 (またはCat.1 とCat.2 デバイスの場合 23 個) のエッジ検出回路で構成されています。各入力ラインは、個別に設定可能であり、タイプ (イベントまたは割り込み) と対応するトリガイイベント (立ち上り、立ち下り、または両方) を選択できます。また、ラインごとにマスクをかけることもできます。ペンディングレジスタは、割り込みリクエストのステータスラインを維持します。

10.2.1 主な機能

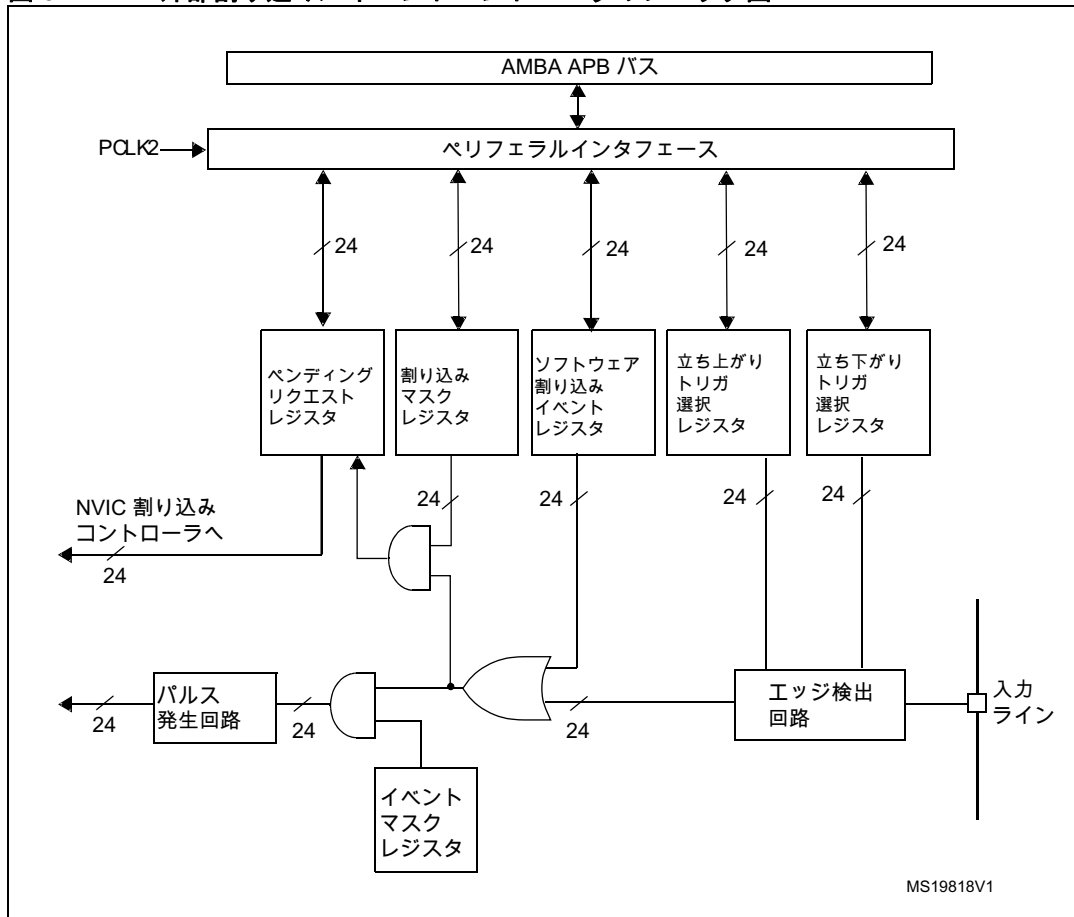
EXTI コントローラは次のような特長を備えています。

- 割り込み／イベントラインごとに独立したトリガとマスク
- 割り込みラインごとに専用のステータスビット
- 最大 24 個 (Cat.1とCat.2 デバイスの場合 23 個) のソフトウェアイベント／割り込みリクエスト
- APB2 クロック周期よりも短いパルス幅の外部信号の検出。このパラメータの詳細については、STM32L1xxxxデータシートの電気特性の項を参照してください。

10.2.2 ブロック図

図 32 にブロック図を示します。

図 32. 外部割り込み／イベントコントローラのブロック図



10.2.3 ウェイクアップイベント管理

STM32L1xxxx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（WFE）をウェイクアップできます。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®-M3 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 から処理を再開する場合、(NVIC 割り込みクリアペンディングレジスタの) ペリフェラル割り込みペンディングビットとペリフェラル 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。
- または、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 から再開するときには、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。

外部ラインをウェイクアップイベントとして使用する方法については、[セクション 10.2.4: 機能の説明](#)を参照してください。

10.2.4 機能の説明

割り込みを生成するには、割り込みラインを設定し、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ミングし、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。

イベントを発生するには、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ミングし、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。

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。

ハードウェア割り込みの選択

24本（またはCat.1とCat.2デバイスの場合23本）の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。

- 割り込みラインのマスクビットを設定します（EXTI_IMR）。
- 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）。
- いずれのラインからの割り込みでも正しく処理できるように、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。

ハードウェアイベントの選択

24本（またはCat.1とCat.2デバイスの場合23本）のラインをイベントソースとして設定するには、次の手順を使用します。

- イベントラインのマスクビットを設定します（EXTI_EMR）。
- イベントラインのトリガ選択ビットを設定します（EXTI_RTISR、EXTI_FTISR）。

ソフトウェア割り込み／イベントの選択

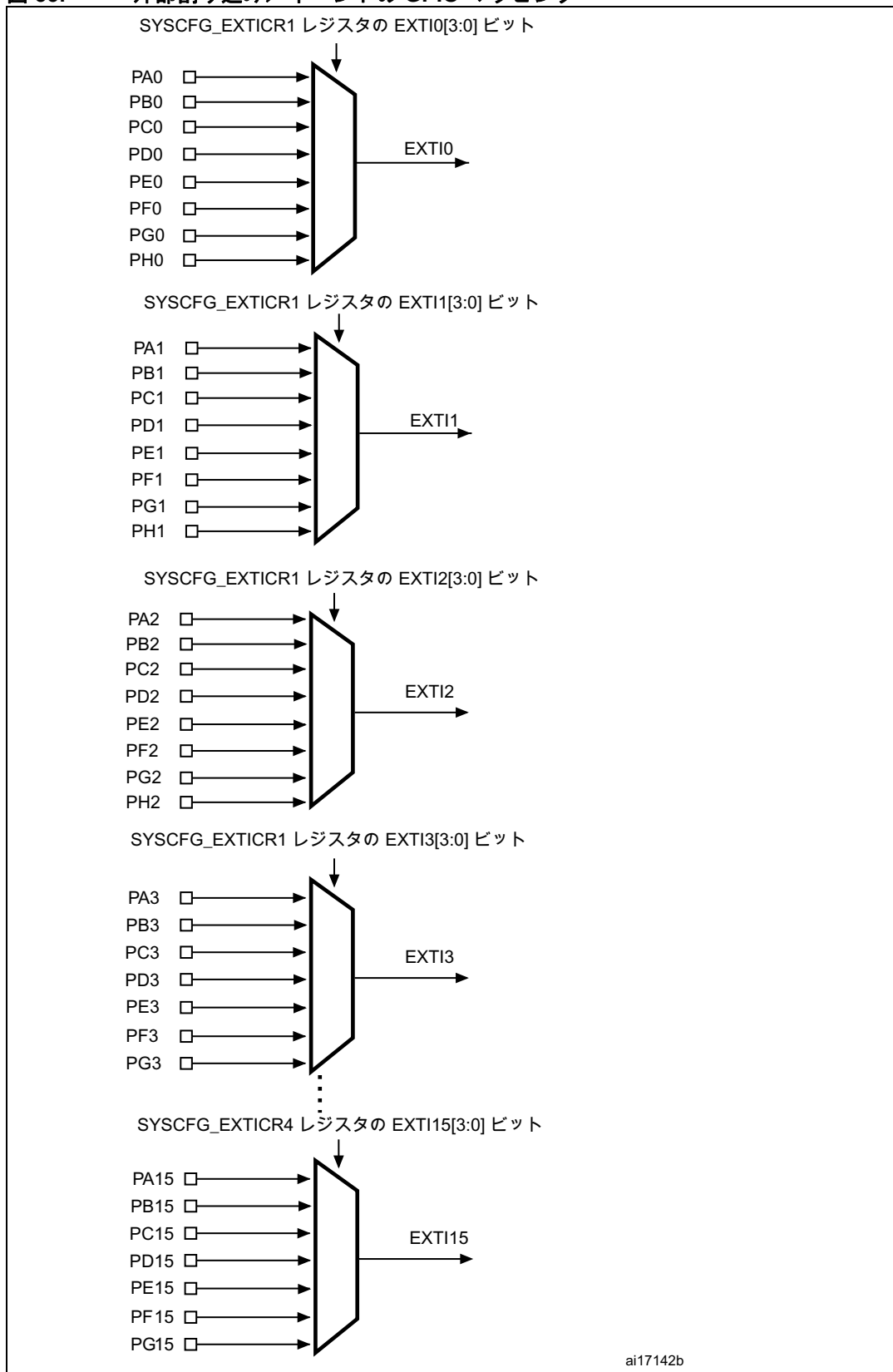
24本（またはCat.1とCat.2デバイスの場合23本）のラインは、ソフトウェア割り込み／イベントラインとして設定できます。ソフトウェア割り込みの生成には、次の手順を使用します。

- 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）。
- 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）。

10.2.5 外部割り込み／イベントラインのマッピング

最大116本のGPIOは、次のように、16本の外部割り込み／イベントラインに接続されます。

図 33. 外部割り込み/イベントの GPIO マッピング



- 他の EXTI ラインは次のように接続されます。
- EXTI ライン 16 は PVD 出力に接続されます。
 - EXTI ライン 17 は RTC アラームイベントに接続されます。
 - EXTI ライン 18 は USB デバイス FS ウェイクアップイベントに接続されます。
 - EXTI ライン 19 は RTC のタンパイイベントおよびタイムスタンプイベントに接続されます。
 - EXTI ライン 20 は RTC ウェイクアップイベントに接続されます。
 - EXTI ライン 21 はコンパレータ 1 ウェイクアップイベントに接続されます。
 - EXTI ライン 22 はコンパレータ 2 ウェイクアップイベントに接続されます。
 - EXTI ライン 23 はチャネルアキュイジション割り込みに接続されます。

10.3 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。

10.3.1 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：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み								MR23 ⁽¹⁾	MR22	MR21	MR20	MR19	MR18	MR17	MR16
								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
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

- 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 23:0 **MRx**：ライン x の割り込みマスク
- 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。



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。

10.3.2 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
予約済み								MR23 ⁽¹⁾	MR22	MR21	MR20	MR19	MR18	MR17	MR16
								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
MR15	MR14	MR13	MR12	MR11	MR10	MR9	MR8	MR7	MR6	MR5	MR4	MR3	MR2	MR1	MR0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

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 23:0 **MRx** : ライン x のイベントマスク

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。

10.3.3 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
予約済み								TR23 ⁽¹⁾	TR22	TR21	TR20	TR19	TR18	TR17	TR16
								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
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

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 23:0 **TRx** : ライン x の立ち上がりエッジトリガイイベント設定ビット

0 : 入力ライン x の立ち上がりエッジトリガを無効化 (イベントと割り込み)

1 : 入力ライン x の立ち上がりエッジトリガを有効化 (イベントと割り込み)

注 : 外部ウェイクアップラインはエッジトリガであるため、ライン上、グリッチの発生は避ける必要があります。

EXTI_RTSR レジスタへの書き込み中に外部割り込みラインで立ち上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、立ち上りエッジトリガと立ち下りエッジトリガを設定することができません。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。

10.3.4 立ち下がリエッジトリガ 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
予約済み								TR23 ⁽¹⁾	TR22	TR21	TR20	TR19	TR18	TR17	TR16
								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
TR15	TR14	TR13	TR12	TR11	TR10	TR9	TR8	TR7	TR6	TR5	TR4	TR3	TR2	TR1	TR0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

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 23:0 **TRx** : ライン x の立ち下がリエッジトリガイイベント 設定ビット

- 0 : 入力ライン x の立ち下がリエッジトリガを無効化 (イベントと割り込み)
- 1 : 入力ライン x の立ち下がリエッジトリガを有効化 (イベントと割り込み)

注 : 外部ウェイクアップラインはエッジトリガであるため、ライン上、グリッチの発生は避ける必要があります。

EXTI_FTSR レジスタへの書き込み中に外部割り込みラインで立ち下がリ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、立ち上リエッジトリガと立ち下リエッジトリガを設定することができません。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。

10.3.5 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
予約済み								SWIER23 ⁽¹⁾	SWIER22	SWIER21	SWIER20	SWIER19	SWIER18	SWIER17	SWIER16
								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
SWIER15	SWIER14	SWIER13	SWIER12	SWIER11	SWIER10	SWIER9	SWIER8	SWIER7	SWIER6	SWIER5	SWIER4	SWIER3	SWIER2	SWIER1	SWIER0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

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 22:0 **SWIERx** : ライン x のソフトウェア割り込み

割り込みが EXTI_IMR レジスタのこのラインで有効になっている場合、SWIERx ビットが 0 にセットされているときに 1 を書き込むと EXTI_PR レジスタの対応するペンディングビットがセットされるため、割り込みリクエストが発生します。

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。

10.3.6 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 : 不定

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み								PR23 ⁽¹⁾	PR22	PR21	PR20	PR19	PR18	PR17	PR16
								rw	rw	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
PR15	PR14	PR13	PR12	PR11	PR10	PR9	PR8	PR7	PR6	PR5	PR4	PR3	PR2	PR1	PR0
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

1. Cat.3、Cat.4、Cat.5 デバイスの場合のみ

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 23:0 **PRx** : ペンディングビット

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外部割り込みラインで発生したときにセットされます。このビットは、“1”を書き込むことによってクリアされます。

10.3.7 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 51. 外部割り込み／イベントコントローラ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	予約済み								MR[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
0x04	EXTI_EMR	予約済み								MR[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
0x08	EXTI_RTSR	予約済み								TR[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
0x0C	EXTI_FTSR	予約済み								TR[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
0x10	EXTI_SWIER	予約済み								SWIER[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
0x14	EXTI_PR	予約済み								PR[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

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。



11 ダイレクトメモリアクセス (DMA) コントローラ

11.1 DMA の概要

DMA (Direct Memory Access) は、ペリフェラルとメモリ間、およびメモリ内で高速データ転送を行うために使用されます。データは、CPU の動作を必要とすることなく、DMA によって高速に移動できます。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。

2 つの DMA コントローラは合計 12 チャンネル (DMA1 に 7 チャンネル、DMA2 に 5 チャンネル) をサポートし、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します。また、DMA リクエスト間の優先順位を管理するアービタを内蔵しています。

11.2 DMA の主な機能

- 独立して構成可能な 12 チャンネル (リクエスト) : DMA1 に 7、DMA2 に 5
- 12 チャンネルのそれぞれは専用のハードウェア DMA リクエストに接続されます。また、各チャンネルではソフトウェアトリガもサポートされます。この設定はソフトウェアによって行われます。
- 1 つの DMA の各チャンネルからのリクエスト間の優先順位は、ソフトウェアから設定できます (最高、高、中、低の 4 レベル)。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)。
- 転送元と転送先で転送サイズ (バイト、ハーフワード、ワード) を個別に指定でき、この際にパッキングとアンパッキングがエミュレートされます。転送元/転送先アドレスは、データサイズに合わせる必要があります。
- サークキュラバッファ管理のサポート
- チャンネルごとに 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 1 つの割り込みリクエストに格納されます。
- メモリ間転送
- 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相互間の転送
- 転送元/転送先として、SRAM、APB1、APB2、および AHB ペリフェラルへのアクセス、および転送元としてフラッシュへのアクセス
- 転送データ数のプログラミングが可能 : 最大 65536

ブロック図を 図 34 に示します。

図 34. Cat.1およびCat.2 STM32L1xxxデバイスの DMA ブロック図

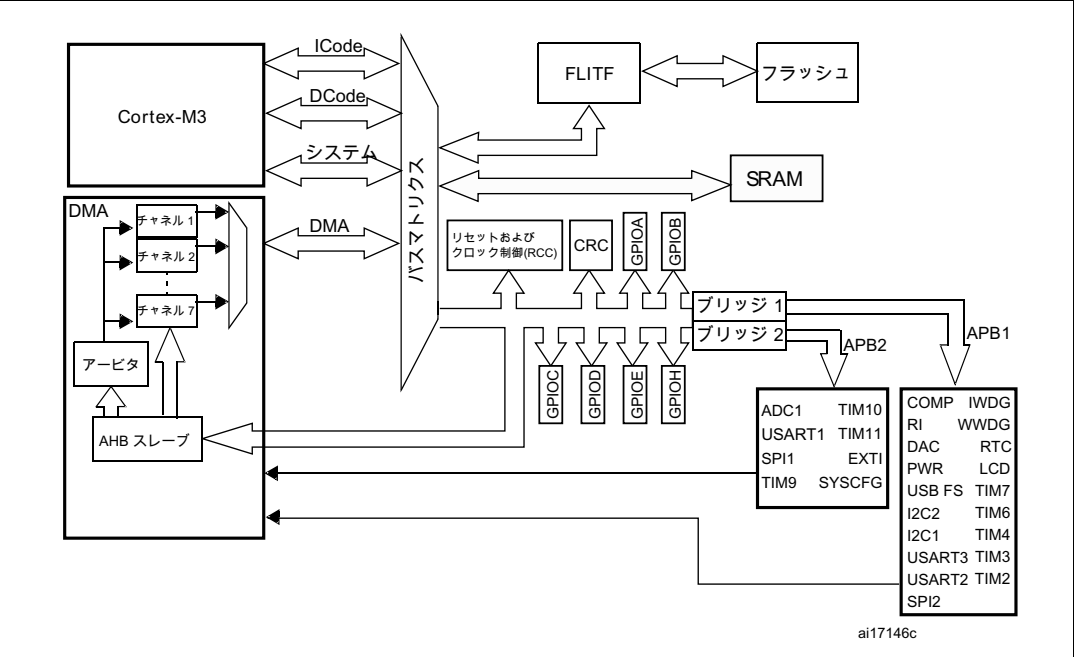


図 35. Cat.3STM32L1xxxx デバイスの DMA ブロック図

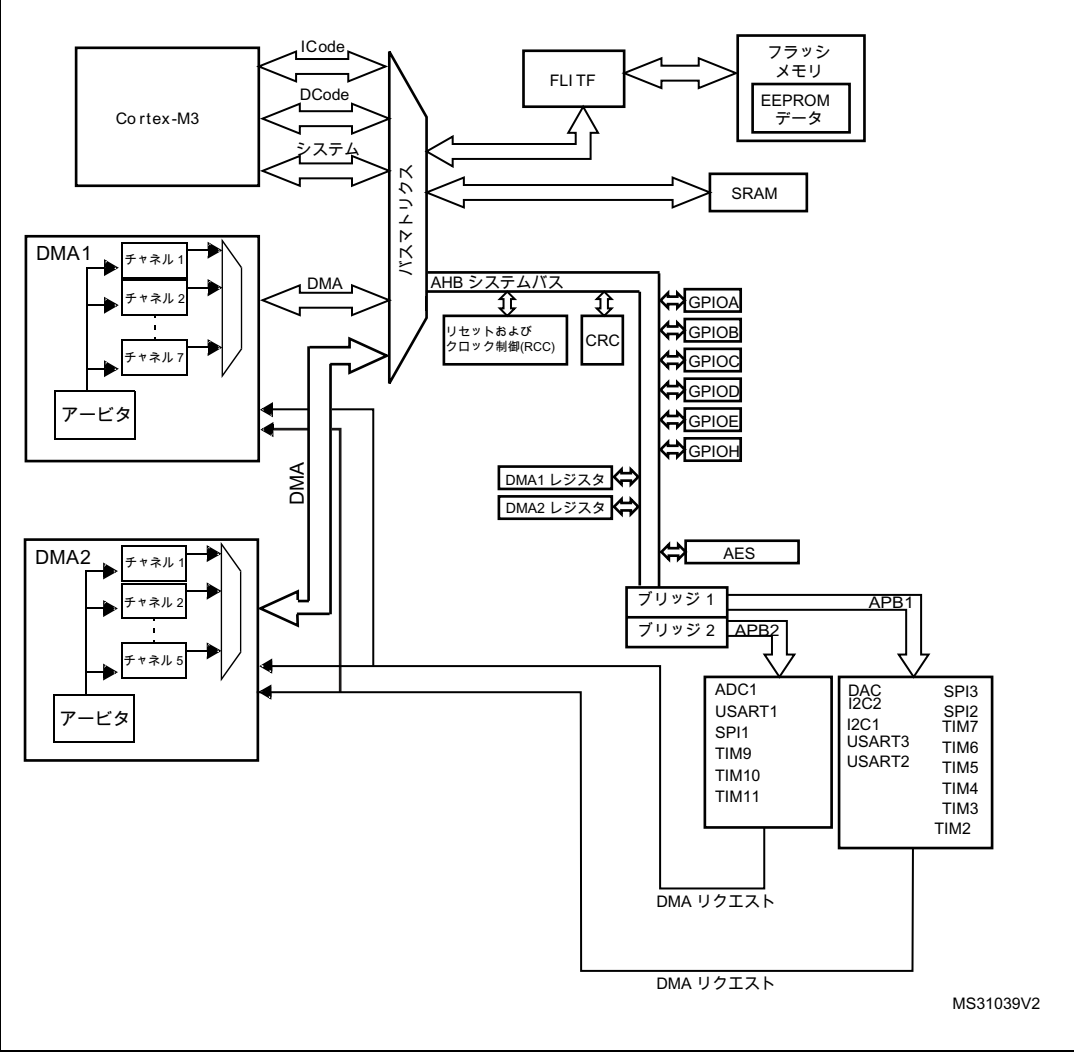
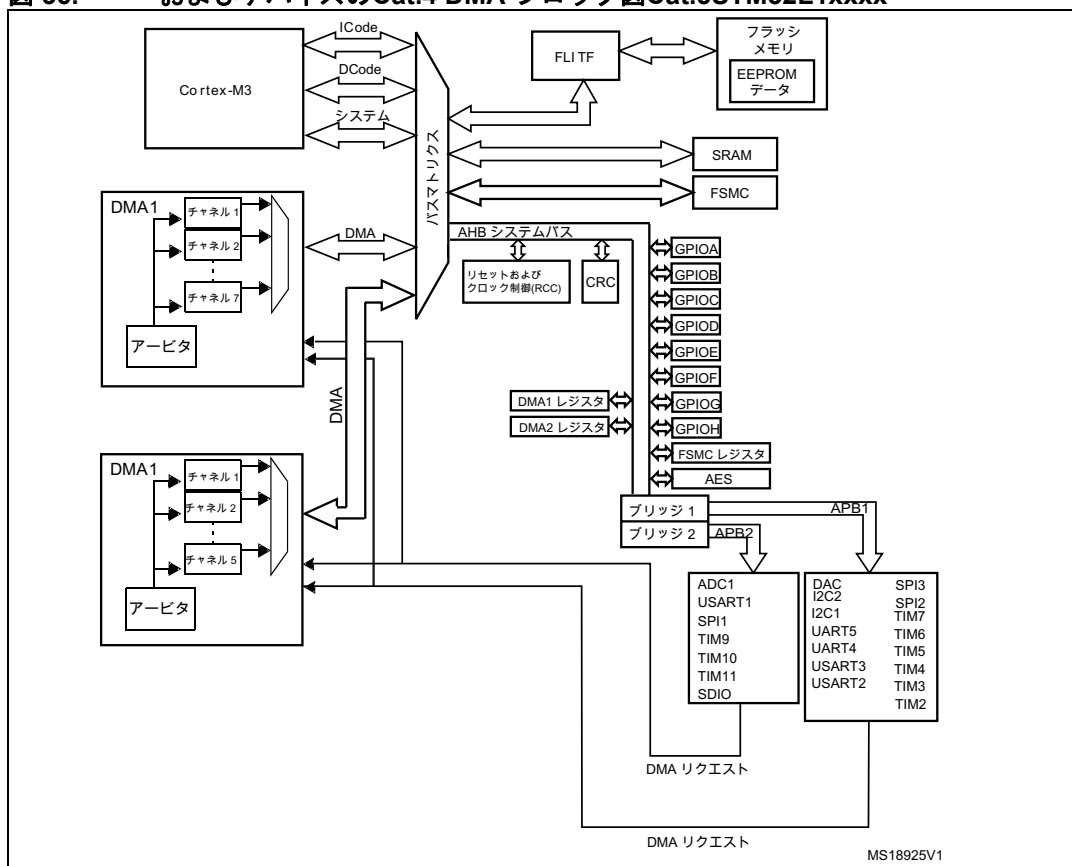


図 36. およびデバイスのCat.4 DMA ブロック図Cat.5STM32L1xxxx



注: DMA2 コントローラと関連リクエストはCat.3、Cat.4 およびCat.5 デバイスでのみ使用できます。
 SPI3 と TIM5 DMA リクエストはCat.3、Cat.4 およびCat.5 デバイスでのみ使用可能です。
 USART4 と USART5 はCat.4 およびCat.5 デバイスでのみ使用可能です。
 SDIO はCat.4 デバイスでのみ使用可能です。

11.3 DMA 機能の説明

DMA コントローラは、S-Bus を Cortex[®]-M3 コアと共有することによってダイレクトメモリ転送を実行します。CPU と DMA が同じ転送先 (メモリまたはペリフェラル) をターゲットにしている場合、DMA リクエストによって S-Bus への CPU アクセスが数バスサイクルの間、停止することがあります。バスマトリックスがラウンドロビン方式を採用することで、(メモリとペリフェラルに対する) S-Bus 帯域の少なくとも 1/2 が CPU に確保されます。

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、チャンネル間の優先順位に応じて、このリクエストに対応します。DMA コントローラがペリフェラルにアクセスすると、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、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ネゲートされると、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 つの操作で構成されます。

- 内部カレントペリフェラル/メモリアドレスレジスタによってアドレス指定されたメモリ位置またはペリフェラルデータレジスタ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ミングされたベースペリフェラル/メモリアドレスです。
- 内部カレントペリフェラル/メモリアドレスレジスタによってアドレス指定されたメモリ位置またはペリフェラルデータレジスタ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ミングされたベースペリフェラル/メモリアドレスです。
- DMA_CNDTRx レジスタのポストデクリメント。このレジスタには、まだ実行する必要があるトランザクション数が含まれています。

11.3.2 アービタ

アービタは、チャンネルリクエストを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開始します。

優先順位は 2 段階で管理されます。

- ソフトウェア : 各チャンネルの優先順位は、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レベルが同じ場合、小さな番号のチャンネルが大きな番号のチャンネルよりも優先されます。たとえば、チャンネル 2 はチャンネル 4 よりも優先されます。

11.3.3 DMA チャンネル

各チャンネルは、固定アドレスに置かれたペリフェラルレジスタとメモリアドレスとの間の DMA 転送を処理できます。転送されるデータ量（最大 65535）はプログラミング可能です。転送されるデータ項目の量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。

プログラミング可能なデータサイズ

ペリフェラルや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 ビットによって完全にプログラミング可能です。

ポインタインクリメント

ペリフェラルやメモリのポインタは、DMA_CCRx レジスタの PINC および MINC ビットに応じて、各トランザクションの終了後に自動的にポストインクリメントできます。インクリメントモードが有効な場合、次の転送のアドレスは、選択されたデータサイズに応じて、前回のアドレスに 1、2、または 4 をインクリメントしたアドレスになります。最初の転送アドレスは、DMA_CPARx/DMA_CMARx レジスタでプログラミングされたアドレスです。転送動作中、これらのレジスタは、最初にプログラミングされた値を保持します。（内部カレントペリフェラル/メモリアドレスレジスタにある）現在の転送アドレスには、ソフトウェアからアクセスできません。

チャンネルが非サーキュラモードに設定されている場合、最後の転送後（すなわち、転送されるデータ項目の数がゼロになると）、DMA リクエストは処理されません。転送される新しいデータ項目数を DMA_CNDTRx レジスタに再ロードするには、DMA チャンネルが無効でなければなりません。

注： *DMA チャンネルが無効の場合、DMA レジスタはリセットされません。DMA チャンネルレジスタ (DMA_CCRx、DMA_CPARx、および DMA_CMARx) は、チャンネル設定段階でプログラミングされた初期値を保持します。*

サーキュラモードでは、最後の転送の後、DMA_CNDTRx レジスタには、初めにプログラミングされ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 を設定するには、次の手順に従います（x はチャンネル番号です）。

1. DMA_CPARx レジスタで、ペリフェラルレジスタアドレスを設定します。ペリフェラルイベントの後、データはこのアドレスとメモリの間で転送されます。
2. メモリアドレスを DMA_CMARx レジスタにセットします。ペリフェラルイベントの後、データはこのメモリに対して読み書きされます。
3. 転送されるデータの合計数を DMA_CNDTRx レジスタで設定します。各ペリフェラルイベントの後、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. データ転送の方向、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、および 1/2 転送終了や転送完了後の割り込みを DMA_CCRx レジスタで設定します。
6. DMA_CCRx レジスタのイネーブルビットをセットして、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 リクエストを処理できます。

バイト数の半分が転送されると、1/2 転送終了フラグ (HTIF) がセットされ、1/2 転送終了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、転送されるデータ数にはチャンネル設定段階でプログラミング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。

メモリ間転送モード

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、メモリ間転送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た場合、ソフトウェアによって DMA_CCRx レジスタのイネーブルビット (EN) がセットされてチャンネルが有効になると、チャンネルは転送を開始します。DMA_CNDTRx レジスタの値が 0 になると、転送は停止します。メモリ間転送モードとサーキュラモードは、同時に使用できないことがあります。

11.3.4 プログラミング可能なデータ幅、データの配置、およびエンディアン

PSIZE と MSIZE が等しくないとき、表 52: プログラミング可能なデータ幅とエンディアン動作(ビット PINC = MINC = 1 のとき) で説明するように、DMA は何らかのデータ配置を実行します。

表 52. プログラミング可能なデータ幅とエンディアン動作 (ビット PINC = MINC = 1 のとき)

ソースポート幅	転送先ポート幅	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	転送操作	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: READ B0[7:0] @0x0 then WRITE B0[7:0] @0x0 2: READ B1[7:0] @0x1 then WRITE B1[7:0] @0x1 3: READ B2[7:0] @0x2 then WRITE B2[7:0] @0x2 4: READ B3[7:0] @0x3 then WRITE B3[7:0] @0x3	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3
8	16	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1: READ B0[7:0] @0x0 then WRITE 00B0[15:0] @0x0 2: READ B1[7:0] @0x1 then WRITE 00B1[15:0] @0x2 3: READ B3[7:0] @0x2 then WRITE 00B2[15:0] @0x4 4: READ B4[7:0] @0x3 then WRITE 00B3[15:0] @0x6	@0x0 / 00B0 @0x2 / 00B1 @0x4 / 00B2 @0x6 / 00B3
8	32	4	@0x0 / B0 @0x1 / B1 @0x2 / B2 @0x3 / B3	1: READ B0[7:0] @0x0 then WRITE 000000B0[31:0] @0x0 2: READ B1[7:0] @0x1 then WRITE 000000B1[31:0] @0x4 3: READ B3[7:0] @0x2 then WRITE 000000B2[31:0] @0x8 4: READ B4[7:0] @0x3 then WRITE 000000B3[31:0] @0xC	@0x0 / 000000B0 @0x4 / 000000B1 @0x8 / 000000B2 @0xC / 000000B3
16	8	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1: READ B1B0[15:0] @0x0 then WRITE B0[7:0] @0x0 2: READ B3B2[15:0] @0x2 then WRITE B2[7:0] @0x1 3: READ B5B4[15:0] @0x4 then WRITE B4[7:0] @0x2 4: READ B7B6[15:0] @0x6 then WRITE B6[7:0] @0x3	@0x0 / B0 @0x1 / B2 @0x2 / B4 @0x3 / B6
16	16	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1: READ B1B0[15:0] @0x0 then WRITE B1B0[15:0] @0x0 2: READ B3B2[15:0] @0x2 then WRITE B3B2[15:0] @0x2 3: READ B5B4[15:0] @0x4 then WRITE B5B4[15:0] @0x4 4: READ B7B6[15:0] @0x6 then WRITE B7B6[15:0] @0x6	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6
16	32	4	@0x0 / B1B0 @0x2 / B3B2 @0x4 / B5B4 @0x6 / B7B6	1: READ B1B0[15:0] @0x0 then WRITE 0000B1B0[31:0] @0x0 2: READ B3B2[15:0] @0x2 then WRITE 0000B3B2[31:0] @0x4 3: READ B5B4[15:0] @0x4 then WRITE 0000B5B4[31:0] @0x8 4: READ B7B6[15:0] @0x6 then WRITE 0000B7B6[31:0] @0xC	@0x0 / 0000B1B0 @0x4 / 0000B3B2 @0x8 / 0000B5B4 @0xC / 0000B7B6
32	8	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1: READ B3B2B1B0[31:0] @0x0 then WRITE B0[7:0] @0x0 2: READ B7B6B5B4[31:0] @0x4 then WRITE B4[7:0] @0x1 3: READ BBBAB9B8[31:0] @0x8 then WRITE B8[7:0] @0x2 4: READ BFBEBDBC[31:0] @0xC then WRITE BC[7:0] @0x3	@0x0 / B0 @0x1 / B4 @0x2 / B8 @0x3 / BC
32	16	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1: READ B3B2B1B0[31:0] @0x0 then WRITE B1B0[7:0] @0x0 2: READ B7B6B5B4[31:0] @0x4 then WRITE B5B4[7:0] @0x1 3: READ BBBAB9B8[31:0] @0x8 then WRITE B9B8[7:0] @0x2 4: READ BFBEBDBC[31:0] @0xC then WRITE BDBC[7:0] @0x3	@0x0 / B1B0 @0x2 / B5B4 @0x4 / B9B8 @0x6 / BDBC
32	32	4	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC	1: READ B3B2B1B0[31:0] @0x0 then WRITE B3B2B1B0[31:0] @0x0 2: READ B7B6B5B4[31:0] @0x4 then WRITE B7B6B5B4[31:0] @0x4 3: READ BBBAB9B8[31:0] @0x8 then WRITE BBBAB9B8[31:0] @0x8 4: READ BFBEBDBC[31:0] @0xC then WRITE BFBEBDBC[31:0] @0xC	@0x0 / B3B2B1B0 @0x4 / B7B6B5B4 @0x8 / BBBAB9B8 @0xC / BFBEBDBC

バイト/ハーフワード書き込み操作をサポートしない AHB ペリフェラルのアドレス指定

DMA が AHB バイトまたはハーフワードの書き込み動作を開始するとき、HWDATA[31:0] バスの未使用レーンでデータが複製されます。このため、使用される AHB スレーブペリフェラルがバイトまたはハーフワードの書き込み動作をサポートしない場合 (ペリフェラルが HSIZE を使用しない場合) さらにエラーを発生しない場合、DMA は、下の 2 つの例に示すように、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 HSIZE = HalfWord で HWDATA バスに“0xABCDABCD”を設定します。
- バイト“0xAB”を書き込むには、DMA は HSIZE = Byte で HWDATA バスに“0xABABABAB”を設定します。

AHB/APB ブリッジが HSIZE データを考慮しない AHB 32 ビットスレーブペリフェラルであるとするれば、以下の方法で AHB バイト/ハーフワード操作を 32 ビット APB 操作に変換します。

- データ“0xB0”から 0x0 (あるいは、0x1、0x2、または 0x3) への AHB バイト書き込み操作は、データ“0xB0B0B0B0”から 0x0 への APB ワード書き込み操作に変換されます。
- データ“0xB1B0”から 0x0 (または 0x2) への AHB ハーフワード書き込み操作は、データ“0xB1B0B1B0”から 0x0 への APB ワード書き込み操作に変換されます。

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配置された 16 ビットレジスタ) に書き込みたい場合は、メモリ転送元サイズ (MSIZE) を“16 ビット”に設定し、ペリフェラル転送先サイズ (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

予約済みアドレス空間からの読み出し/書き込みによって、DMA 転送エラーが生成されることがあります。DMA の読み出し/書き込みアクセス中に DMA 転送エラーが発生すると、障害のあるチャンネルは、対応するチャンネル設定レジスタ (DMA_CCRx) の EN ビットをハードウェアがクリアすることによって自動的に無効になります。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 割り込み

割り込みは、DMA チャンネルごとに、1/2 転送終了、転送完了、または転送エラーによって発生することがあります。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 53. 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 転送終了	HTIF	HTIE
転送完了	TCIF	TCIE
転送エラー	TEIF	TEIE

11.3.7 DMA リクエストのマッピング

DMA1 コントローラ

ペリフェラル (TIMx[2,3,4,6,7], ADC1, SPI[1,2], I2Cx[1,2], USARTx[1,2,3]) および DAC チャネル x[1,2] からの 7 つのリクエストは、論理和をとった後に DMA1 に渡されます。これは、リクエスト同時に 1 つのリクエストだけが有効になることを意味します。図 37 : DMA1 リクエストの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ミングすることによって、個別に有効/無効にできます。

図 37. DMA1 リクエストのマッピング

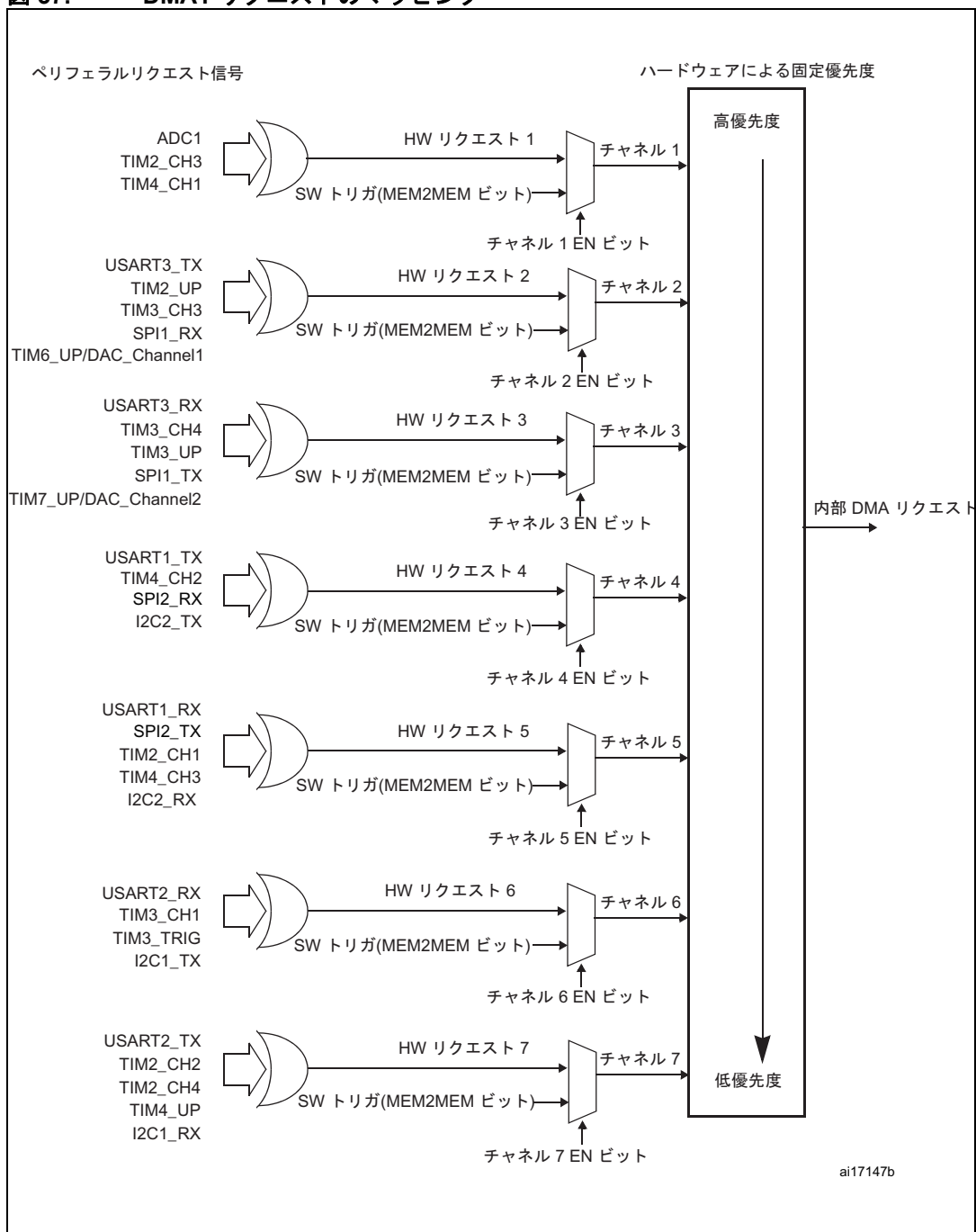


表 54 に各チャネルに対する DMA リクエストの一覧を示します。

表 54. 各チャネル に対する DMA1 リクエストの一覧

ペリフェラル	チャネル 1	チャネル 2	チャネル 3	チャネル 4	チャネル 5	チャネル 6	チャネル 7
ADC1	ADC1	-	-	-	-	-	-
SPI	-	SPI1_RX	SPI1_TX	SPI2_RX	SPI2_TX	-	-
USART	-	USART3_TX	USART3_RX	USART1_TX	USART1_RX	USART2_RX	USART2_TX
I ² C	-	-	-	I2C2_TX	I2C2_RX	I2C1_TX	I2C1_RX
TIM2	TIM2_CH3	TIM2_UP	-	-	TIM2_CH1	-	TIM2_CH2 TIM2_CH4
TIM3	-	TIM3_CH3	TIM3_CH4 TIM3_UP	-	-	TIM3_CH1 TIM3_TRIG	-
TIM4	TIM4_CH1	-	-	TIM4_CH2	TIM4_CH3	-	TIM4_UP
TIM6/DAC_C hannel1	-	TIM6_UP/DAC _Channel1	-	-	-	-	-
TIM7/DAC_C hannel2	-	-	TIM7_UP/DAC _Channel2	-	-	-	-

図 38. DMA2 リクエストのマッピング

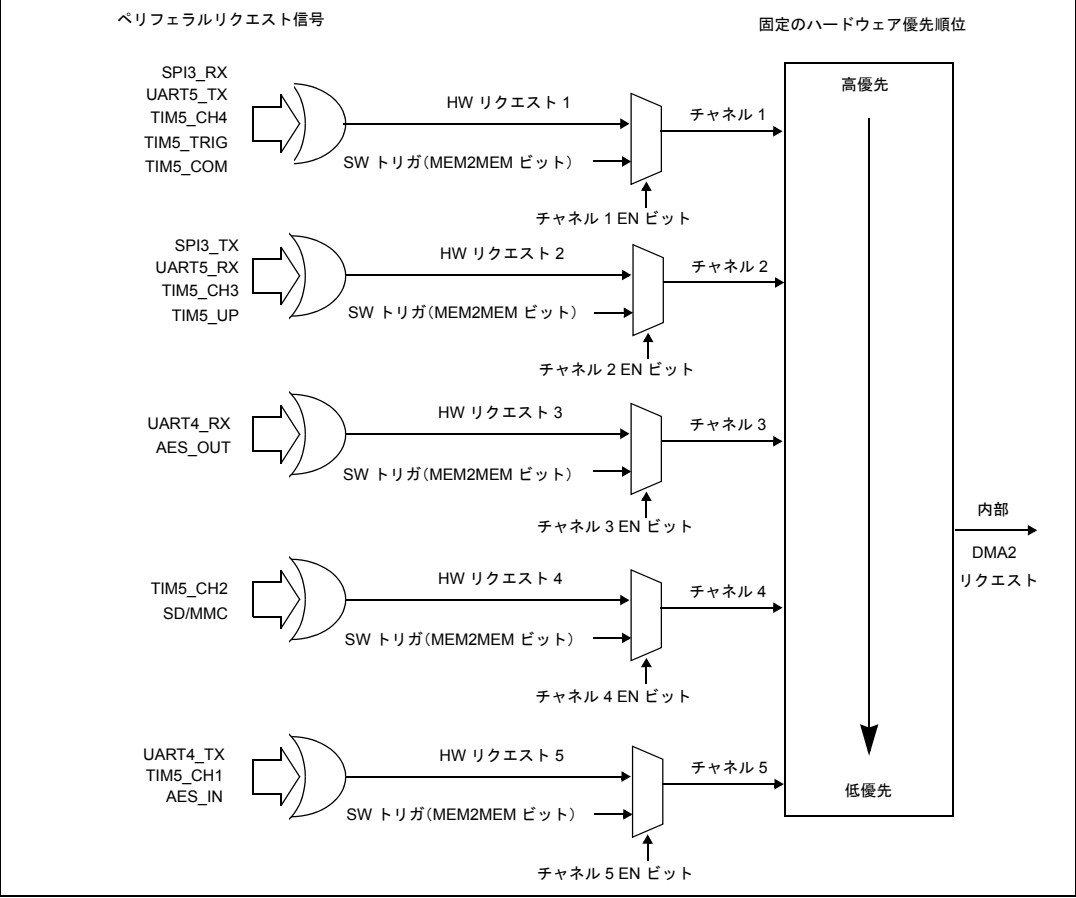


表 55 に、チャンネルに対する DMA2 リクエストを示します。

表 55. 各チャンネルに対する DMA2 リクエストの一覧

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
SPI3	SPI3_RX	SPI3_TX	-	-	-
UART4	-	-	UART4_RX	-	UART4_TX
UART5	UART5_TX	UART5_RX	-	-	-
TIM5	TIM5_CH4 TIM5_TRIG TIM5_COM	TIM5_CH3 TIM5_UP	-	TIM5_CH2	TIM5_CH1
SDIO	-	-	-	SD/MMC	-
AES	-	-	AES_OUT	-	AES_IN

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 : レジスタに関する略記 \(38 ページ\)](#) を参照してください。

注 : *DMA2 には 5 つのチャンネルしかないので、次に示すレジスタでは、チャンネル6 とチャンネル7 関連のすべてのビットは、DMA2 には無関係です。*

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
予約済み				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**: チャンネル 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。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、ソフトウェアによってクリアされます。
- 0 : チャンネル x に転送エラー (TE) は発生していません。
 - 1 : チャンネル x に転送エラー (TE) が発生しました。

ビット 26、22、18、**HTIFx**: チャンネル x ハーフ転送 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。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、ソフトウェアによってクリアされます。
- 0 : チャンネル x に 1/2 転送終了 (HT) イベントは発生していません。
 - 1 : チャンネル x に 1/2 転送終了 (HT) イベントが発生しました。

ビット 25、21、17、**TCIFx**: チャンネル 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。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、ソフトウェアによってクリアされます。
- 0 : チャンネル x に転送完了 (TC) イベントは発生していません。
 - 1 : チャンネル 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**: チャンネル 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。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、ソフトウェアによってクリアされます。
- 0 : チャンネル x に TE、HT、または TC イベントは発生していません。
 - 1 : チャンネル 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
予約済み				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: チャンネル 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/クリアします。
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: チャンネル x ハーフ転送クリア (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/クリアします。
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: チャンネル 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/クリアします。
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: チャンネル 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/クリアします。
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 チャンネル x 設定レジスタ (DMA_CCRx) (x = 1..7, x = チャンネル番号)

アドレスオフセット : $0x08 + 0d20 \times (\text{チャンネル番号} - 1)$

リセット値 : 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済 み	MEM2M EM	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**: メモリ間転送モード

このビットは、ソフトウェアによってセット/クリアします。

0 : メモリ間転送モードは無効です。

1 : メモリ間転送モードは有効です。

ビット 13:12 **PL[1:0]**: チャン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

01: 中

10 : 高

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/クリアします。

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/クリアします。

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/クリアします。

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/クリアします。
0 : ペリフェラルから読み出されます。
1 : メモリから読み出されます。
- 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/クリアします。
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**: ハーフ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/クリアします。
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/クリアします。
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/クリアします。
0 : チャネルは無効です。
1 : チャネルは有効です。

11.4.4 DMA チャネル x データ数レジスタ（DMA_CNDTRx）（x = 1..7） ,
x = チャネル番号）

アドレスオフセット : 0x0C + 0d20 × (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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NDT															
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]**: 転送データ数
転送されるデータ数（0～65535）。このレジスタには、チャネルが無効のときにのみ書き込みできます。チャ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、残りの転送バイト数を示します。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、このレジスタの値は 0 のままです。ただし、チャネルが自動再ロードモードに設定されている場合は、事前にプログラミングされた値が自動的に再ロードされます。
このレジスタの値が 0 の場合、チャネルの有効/無効に関係なく、トランザクションは処理されません。

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 : $0x10 + 0d20 \times (\text{チャンネル番号} - 1)$

リセット値 : 0x0000 0000

このレジスタには、チャンネルが有効のときに、書き込みはできません。

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
PA																															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	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配置されます。

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配置されます。

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 : $0x14 + 0d20 \times (\text{チャンネル番号} - 1)$

リセット値 : 0x0000 0000

このレジスタには、チャンネルが有効のときに、書き込みはできません。

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
MA																															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	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配置されます。

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配置されます。

11.4.7 DMA レジスタマップ

次の表に、DMA レジスタマップとリセット値を示します。

表 56. DMA 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	DMA_ISR	予約済み				TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF6	GIF6	TEIF5	HTIF5	TCIF5	GIF5	TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1
	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
0x004	DMA_IFCR	予約済み				CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5	CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
	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
0x008	DMA_CCR1	予約済み																	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	0																		0														
0x00C	DMA_CNDTR1	予約済み																	NDT[15:0]														
	0																												0	0	0	0	0
0x010	DMA_CPAR1	PA[31:0]																															
	0																																0
0x014	DMA_CMAR1	MA[31:0]																															
	0																																0
0x018	予約済み																																
0x01C	DMA_CCR2	予約済み																	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	0																		0														
0x020	DMA_CNDTR2	予約済み																	NDT[15:0]														
	0																												0	0	0	0	0
0x024	DMA_CPAR2	PA[31:0]																															
	0																																0
0x028	DMA_CMAR2	MA[31:0]																															
	0																																0
0x02C	予約済み																																
0x030	DMA_CCR3	予約済み																	MEM2MEM	PL [1:0]		MSIZE [1:0]		PSIZE [1:0]		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
	0																		0														
0x034	DMA_CNDTR3	予約済み																	NDT[15:0]														
	0																												0	0	0	0	0
0x038	DMA_CPAR3	PA[31:0]																															
	0																																0
0x03C	DMA_CMAR3	MA[31:0]																															
	0																																0
0x040	予約済み																																

[illegible]

表 56. 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
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	
0x090	予約済み																																

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

12 A/D コンバータ (ADC)

このセクションは、特に指定がない限り、STM32L1xxxファミリ全体に適用されます。

12.1 ADC の概要

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 42 のマルチプレクスチャネルを持ち、最大 40 の外部信号と 2 つの内部ソースの測定を行うことができます。これらのチャネルの 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 の変換結果は、左詰めまたは右詰め 16 ビット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。

変換は、与えられたシステムクロック周波数に対して最大の変換レートになるように常に最大速度で実行されます。自動電源制御によって、変換中のみ ADC をパワーオンさせることにより大幅に消費電力を削減できます。

12.2 ADC の主な機能

- 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
- レギュラ変換終了時、インジェクト変換終了時、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の場合（レギュラ変換）に割り込みを生成
- シングルおよび連続変換モード
- 自動変換のスキャンモード順序がフル設定可能
- 組み込みデータコヒーレンシによりデータアライメントが設定可能
- 各 ADC チャネルサンプリング時間が個別に設定可能
- レギュラ変換とインジェクト変換で検出エッジが設定可能な外部トリガオプション
- 不連続モード
- ADC 変換時間：全速で 1 μ s (ADC クロック = 16 MHz) から低速で 4 μ s (ADC クロック = 4 MHz) まで、APB クロックと独立
- 消費電力を低減する自動パワーアップ/パワーダウン
- ADC 電源条件:
 - 全速または基準ズーミング ($V_{REF+} < V_{DDA}$) で 2.4 V から 3.6 V
 - 低速の 1.8 V まで
- ADC 入力範囲: $V_{REF-} \leq V_{IN} \leq V_{REF+}$
- 設定可能なハードウェア遅延を変換と変換の間に自動挿入
- レギュラチャネル変換中の DMA リクエストの生成

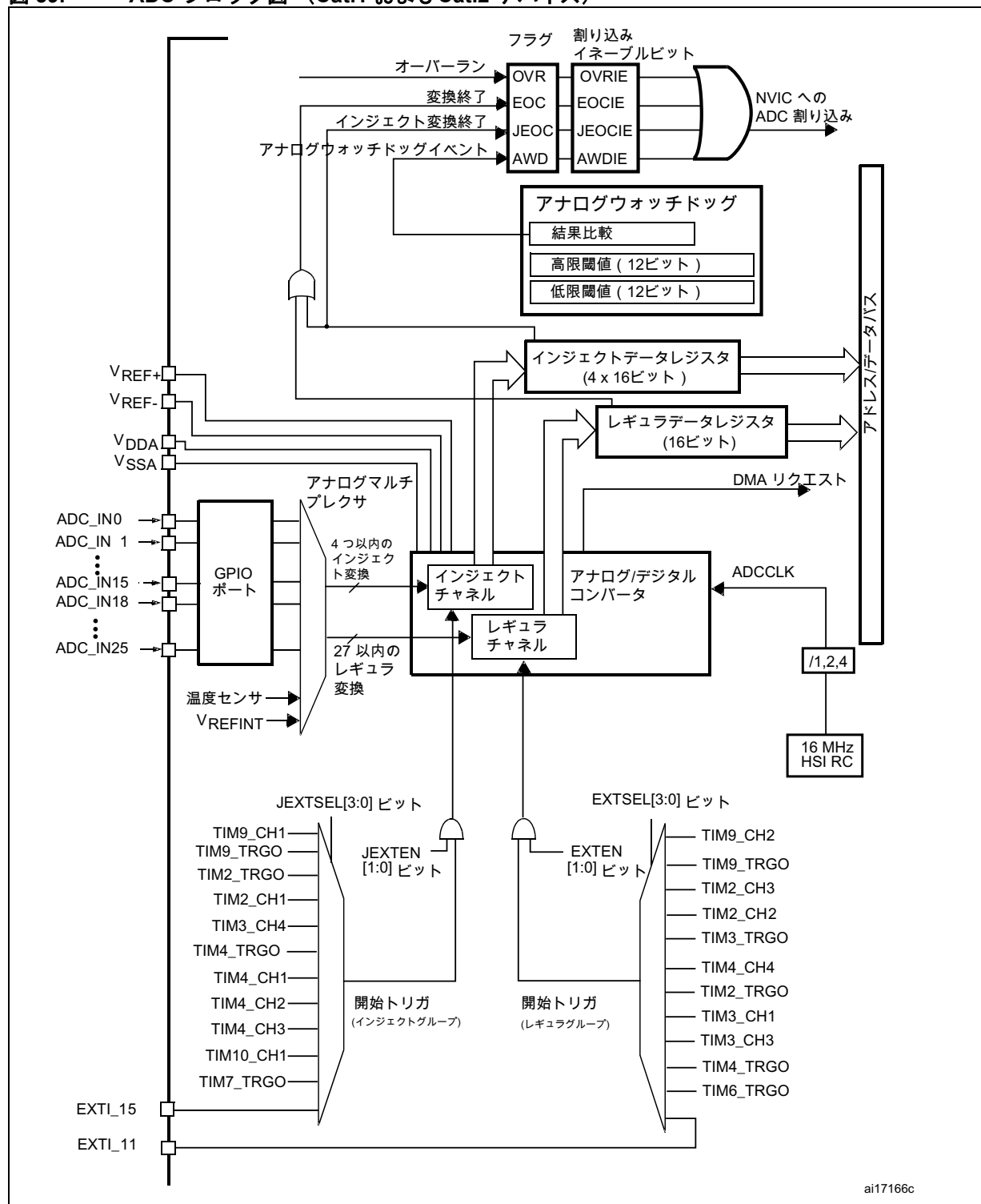
 39に、ADC のブロック図を示します。

注： V_{REF-} ピンが用意されているパッケージでは、このピンを V_{SSA} に接続する必要があります。

12.3 ADC の機能詳細

図 39 と図 40 に、ADC のブロック図を、表 57 に ピンの説明を示します。

図 39. ADC ブロック図 (Cat.1 および Cat.2 デバイス)

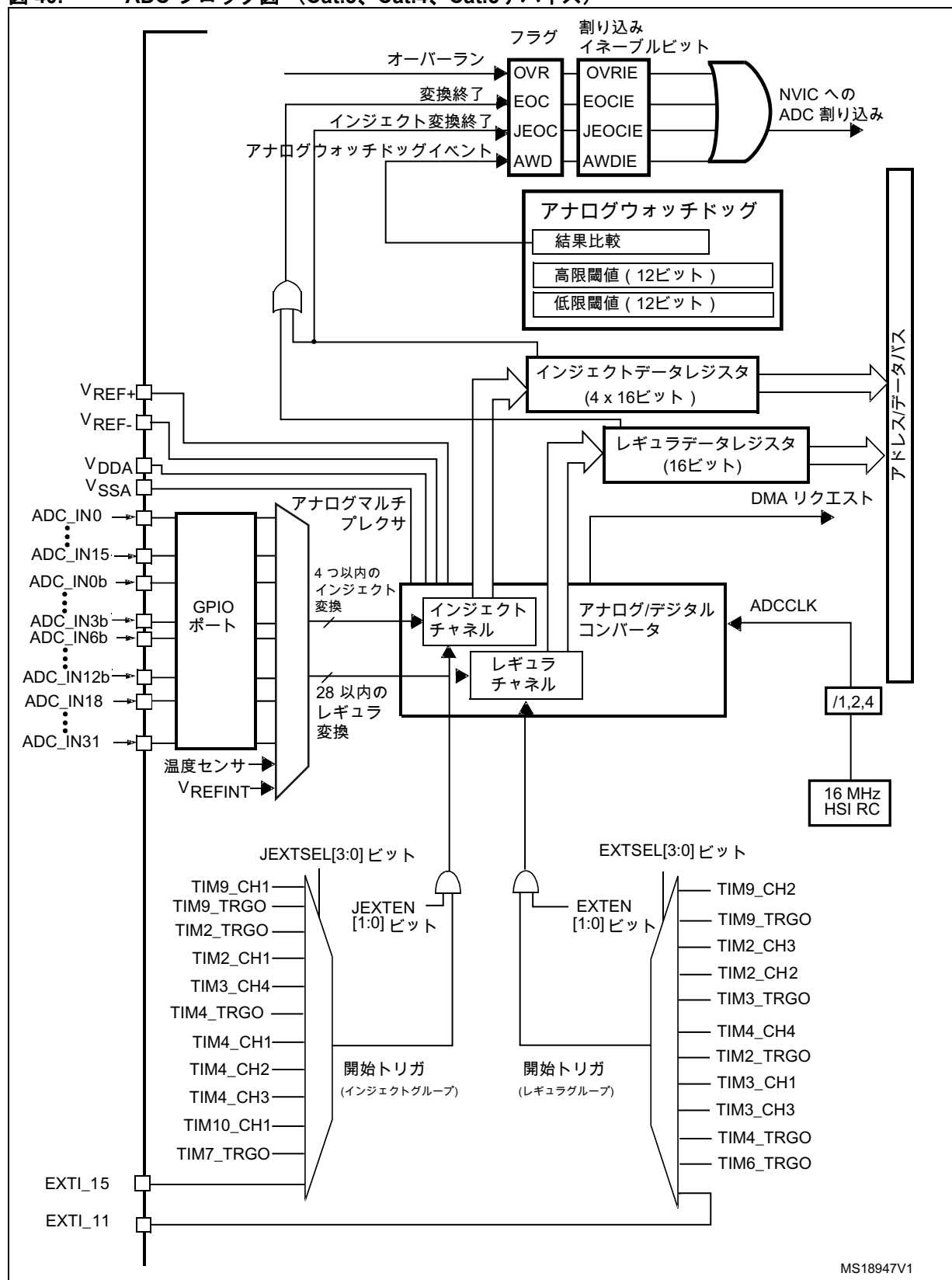


注： 内部接続 (ADC マルチプレクサスイッチ) のため、ADC チャンネル 4、5、22、23、24、25 は最大サンプリングレートのダイレクトチャンネルとなり、他のチャンネルは低速サンプリングレートでマルチプレクスされています。詳細については、製品データシートを参照してください。詳細については、[図 25: Cat.1 および Cat.2 製品のルーティングインタフェース \(RI\) ブロック図 \(ページ\) 198](#)、[図 26: Cat.3 製品のルーティングインタフェース \(RI\) ブロック図 \(ページ\) 199](#)、[図 27: Cat.4 および Cat.5 製品のルーティングインタフェース \(RI\) ブロック図 \(ページ\) 200](#)を参照してください。

表 57. ADC ピン

名称	信号種別	摘要
V_{REF+}	入力、アナログ正基準電圧	ADC の正基準電圧は、 全速で $2.4\text{ V} \leq V_{REF+} = V_{DDA}$ (ADCCLK = 16 MHz、1 Msps) 中速で $1.8\text{ V} \leq V_{REF+} = V_{DDA}$ (ADCCLK = 8 MHz、500 Ksps) 中速で $2.4\text{ V} \leq V_{REF+} \neq V_{DDA}$ (ADCCLK = 8 MHz、500 Ksps) 低速で $1.8\text{ V} \leq V_{REF+} < V_{DDA}$ (ADCCLK = 4 MHz、250 Ksps) 製品電圧範囲 3 を選択した場合 ($V_{CORE} = 1.2\text{ V}$)、ADC は低速 (ADCCLK = 4 MHz、250 Ksps)
V_{DDA}	入力、アナログ電源	アナログ電源は V_{DD} で、 全速では $2.4\text{ V} \leq V_{DDA} \leq V_{DD}$ (3.6 V) 中速と低速では $1.8\text{ V} \leq V_{DDA} \leq V_{DD}$ (3.6 V)
V_{REF-}	入力、アナログ負基準	ADC の負基準電圧、 $V_{REF-} = V_{SSA}$
V_{SSA}	入力、アナログ電源グラウンド	V_{SS} に等しいアナログ電源供給のグラウンド
ADC_IN[15:0] と ADC_IN[25:18] ADC_IN[31:27]、 ADC_IN[3:0]b および ADC_IN[12:6]b	アナログ入力信号	Cat.1 および Cat.2 デバイスでは 24 のアナログ入力チャンネル Cat.3、Cat.4、Cat.5 デバイスでは最大 40 チャンネル

図 40. ADC ブロック図 (Cat.3、Cat.4、Cat.5デバイス)



MS18947V1

12.3.1 ADC 電源のオン/オフ制御

ADC_CR2 レジスタの ADON ビットをセットすると、ADC はパワーオンします。ADON ビットが最初にセットされると、ADC はパワーダウンモードからウェイクアップします。

SWSTART ビットまたは JSWSTART ビットのいずれかがセットされるか、または外部トリガに応答すると変換が開始されます。ADC 変換の準備が整っている場合にのみ (ADONS=1)、これらのソフトウェアまたはハードウェアトリガを有効化する必要があります。

ADON ビットをリセットすると、変換が停止して、ADC はパワーダウンモードになります。このモードでは、ADC の消費電力はほぼゼロです。ADON が ADCCLK クロックドメインに同期化されると、ADONS がクリアされます。

注： 2 つのクロックドメインの間の同期のために生ずる遅延のため、ADONS=0 のときに ADON をセットし、ADC 変換がレディのとき (ADONS=1)、クリアする必要があります。

パワーダウンの設定 (PDI および PDD)

ADC 変換レディ (ADONS=1) のときに消費電力を小さくするため、ADC は非変換時に自動的にパワーオフすることができ、ADC_CR1 レジスタの PDI ビットと PDD ビットに応じて次の変換が開始するまでこのパワーオフが続きます。詳細については、[セクション 12.10：省電力 \(292 ページ\)](#) を参照してください。

PDI ビットを使って、非変換時 (ハードウェアまたはソフトウェアトリガイベント待ちの時) の ADC のパワーアップまたはパワーダウンを調べることができます。

PDD ビットを使って、遅延が挿入されたとき (DELS ビット) に、2 つの変換 (変換のシーケンス) の間での ADC のパワーアップまたはパワーダウンを調べることができます。

PDI=1 のとき、ADCCLK クロックから見たとき、ADONS は ADON と同じ値になります。

SWSTART ビットまたは JSWSTART ビットがセットされるか、または外部トリガに応答すると、ADC パワーアップ時間 (t_{STAB}) が経過した後に変換が開始されます。ADC 変換の準備が整っている場合にのみ (ADONS=1)、これらのソフトウェアまたはハードウェアトリガを有効化する必要があります。

ADON ビットをリセットすると、変換が停止して、ADC 電源非供給モードになります。

注： 2 つのクロックドメインの間の同期のために生ずる遅延のため、ADONS=0 のときに ADON をセットし、ADONS=1 のときクリアする必要があります。

12.3.2 ADC クロック

非変換時の不要な電力消費を回避するため、ADC デジタルインタフェースは、CPU 動作周波数（数 kHz から 32 MHz）に無関係に、全く独立に、内部 16 MHz クロックソース（HSI）を使う最大速度で動作するようにデザインされています。

注： Stop モードを開始するとき、HSI と PCLK2 が無効化されるため、ADC のアナログとデジタルのインタフェースは非アクティブのままになります。Stop モードから回復した後、HSI が非アクティブのままであるため、ADC アナログインタフェースのクロックソースとして HSI を有効化して、ADC 変換の使用を続ける必要があります。

ADCCLK クロックは、クロックコントローラから供給されます。このクロックは HSI 発振器で生成され、分周器の後では次のようになります。

- 全速に対しては 1 分周 ($f_{\text{ADCCLK}} = 16 \text{ MHz}$)
- 中速に対しては 2 分周、低速に対しては 4 分周 ($f_{\text{ADCCLK}} = 4 \text{ MHz}$)

APB クロック (PCLK) 周波数に応じて、ADCCLK クロック周波数は PCLK より高くまたは低くすることができます。特に、APB が低くなりすぎると、データの損失なく全速で変換結果を得ることが難しくなります（データフローが CPU または DMA で処理できる速度より高くなるため）。この問題は、次のデータ到着の前に変換したデータを読み出して保存するための十分な時間をシステムが確保できるようにするため、2 つの変換の間または 2 つの変換シーケンスの間に遅延を挿入することにより解決することができます。詳細については、[セクション 12.9：ハードウェアフリーズと低速変換用の遅延挿入モード \(289 ページ\)](#) を参照してください。

12.3.3 チャネルの選択

Cat.1 および Cat.2 デバイスには、最大 26 のマルチプレクスチャネル ADC_IN0 から 25 があります。

Cat.3、Cat.4、Cat.5 デバイスには、2 バンク構成の最大 42 のマルチプレクスチャネルがあります。チャネル ADC_IN0 から ADC_IN31 はバンク A に、チャネル ADC_IN0b から ADC_IN31b はバンク B に、それぞれ存在します。バンクの選択は、ADC_CR2 レジスタの ADC_CFG ビットを使って設定します。温度センサ入力、内部電圧基準入力、選択された外部 アナログ入力すなわち ADC_IN4 / ADC_IN4b () は、両バンクに存在します。詳細については、[表 40：I/O グループと選択](#) を参照してください。

変換は、レギュラとインジェクトの 2 つのグループに構成することができます。グループは、選択したバンクからの任意の順序で任意の入力チャネルで実行できる変換シーケンスから構成されています。例えば、変換シーケンスを ADC_IN3、ADC_IN8、ADC_IN2、ADC_IN2、ADC_IN0、ADC_IN2 の順序で実現することができます。

- レギュラグループは最大 28 の変換から構成されます。レギュラチャネルと変換シーケンス内での順序は、ADC_SQRx レジスタで選択する必要があります。レギュラグループで最大 28 まで可能な合計変換数は、ADC_SQR1 レジスタの L[4:0] ビットに書き込む必要があります。
- インジェクトグループは、最大 4 つの変換で構成されます。インジェクトされるチャネルと変換シーケンス内での順序は、ADC_JSQR レジスタで選択する必要があります。インジェクトグループで最大 4 まで可能な合計変換数は、ADC_JSQR レジスタの L[1:0] ビットに書き込む必要があります。

注： レギュラ変換で ADC_SQRx レジスタが変更された場合、またはインジェクト変換で ADC_JSQR レジスタが変更された場合、現在の変換がリセットされ、ADC は新しい開始パルスを待ちます。リセットされた変換がレギュラ変換を中断させたインジェクト変換である場合、レギュラ変換は再開されます。

最大 6 個のダイレクトチャネルを ADC の入力スイッチマトリックスに接続して、1 MSamples/s の全速でのサンプルを可能にします。残りの入力チャネルはルーティングインタフェースを介してマルチプレクスし、速度を 800 kSamples/s に低下させます。詳細については、[セクション 8.2：RI の主な特徴](#) を参照してください。

温度センサ、V_{REFINT} 内部チャネル

温度センサはチャネル ADC_IN16 に、内部 基準 電圧 V_{REFINT} は ADC_IN17 に、それぞれ接続されます。この 2 つの内部チャネルは、インジェクトまたはレギュラチャネルとして選択して変換できます。

12.3.4 シングル変換モード

シングル変換モードでは ADC は 1 回の変換を行います。このモードは、次の理由で、ADC_CR2 の CONT ビットが 0 のとき開始されます。

- ADC_CR2 レジスタの SWSTART ビットをセット（レギュラチャネルの場合）
- JSWSTART ビットをセット（インジェクトチャネルの場合）
- 外部 トリガ（レギュラまたはインジェクトチャネルの場合）

選択したチャネルの変換が完了すると、

- レギュラチャネルを変換した場合（変換したチャネルを SQR5 レジスタの SQ1[4:0] ビットで選択）：
 - － 変換したデータは 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、割り込みが発生します。
- インジェクトチャネルを変換した場合（変換したチャネルを JSQR レジスタの JSQ1[4:0] ビットで選択）：
 - － 変換したデータは 16 ビットの ADC_JDR1 レジスタに格納されます。
 - － JEOP（End Of Conversion Injected）フラグがセットされます。
 - － JEOCIE ビットがセットされている場合、割り込みが発生します。

次に ADC が停止します。

12.3.5 連続変換モード

連続変換モードでは、ADC で 1 つの変換が終了すると、直ちに新しい変換が開始されます。CONT ビットが 1 であれば、外部トリガまたは ADC_CR2 レジスタの SWSTART ビットをセットすることのいずれかによって、このモードが開始されます（レギュラチャネルのみ）。

各変換の後:

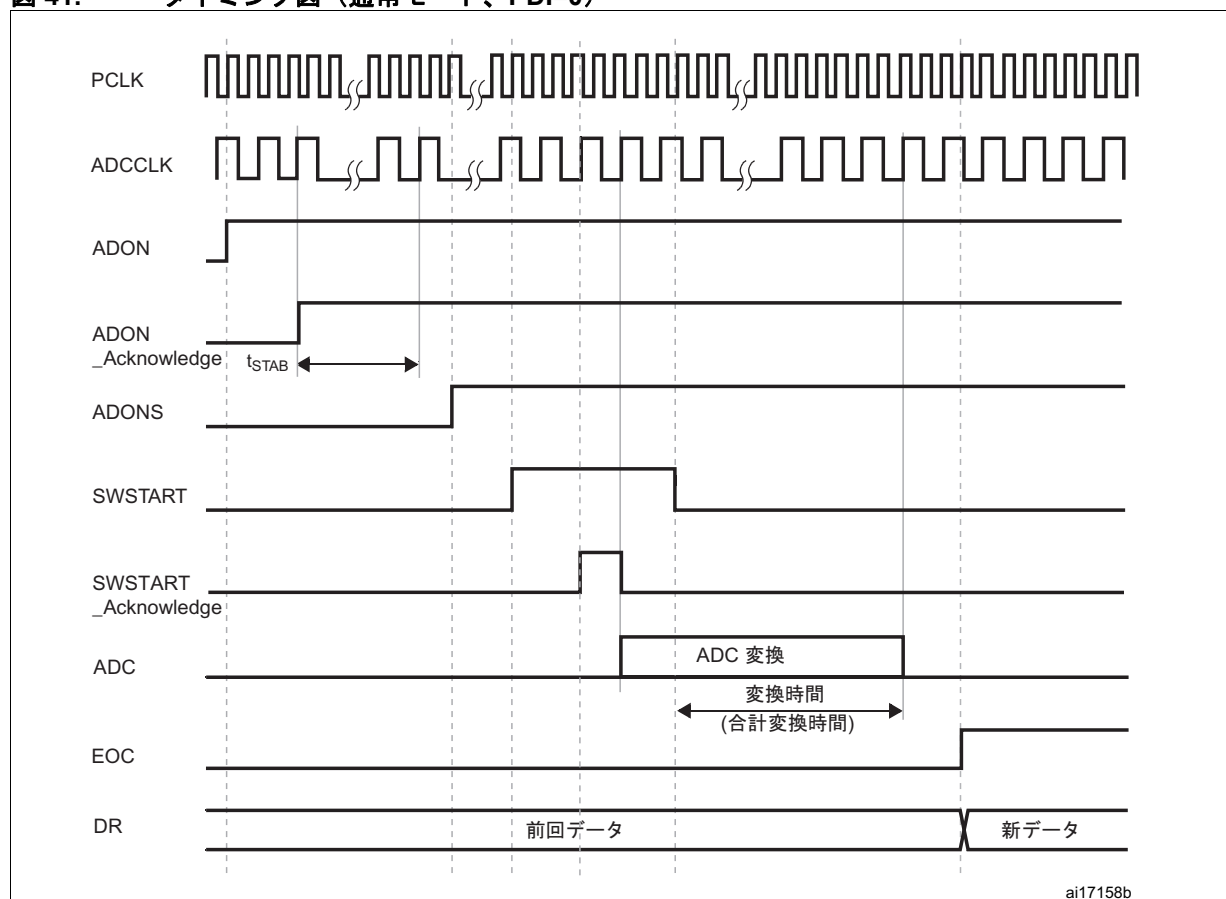
- レギュラチャネルを変換した場合（変換したチャネルを SQR5 レジスタの SQ1[4:0] ビットで選択）：
 - － 直前に変換されたデータは、16 ビットの ADC_DR レジスタに格納されます。
 - － EOC（end of conversion）フラグがセットされます。
 - － EOCIE ビットがセットされている場合、割り込みが発生します。

注： インジェクトチャネルは連続変換できません。唯一の例外は、連続モードでレギュラチャネルの後にインジェクトチャネルを自動的に変換するよう設定されている（JAUTO ビットを使用して）場合です。**自動インジェクト変換**を参照してください。

12.3.6 タイミング図

図 41 に示すように、ADC は実際に変換する前に安定化時間 (t_{STAB}) を必要とします。変換を開始できるとき ADON_{DS} ビットがセットされます。SWSTART ビットがセットされると (または外部トリガが検出されると)、変換が開始されます。変換時間経過後 (12 ビットデータの場合プログラマブルなサンプリング時間 + 12 ADCCLK クロックサイクル)、EOC フラグがセットされ、ADC データレジスタに変換結果が格納されます。異なる信号を 1 つのクロックドメインから別のドメインへ再同期させるためには、遅延が必要であることに注意してください。

図 41. タイミング図 (通常モード、PDI=0)



12.3.7 アナログウォッチドッグ

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、16 ビットの ADC_HTR レジスタおよび ADC_LTR レジスタの下位 12 ビットに設定されます。ADC_CR1 レジスタの AWDIE ビットを使用して、割り込みを有効にできます。

閾値は、ADC_CR2 レジスタの ALIGN ビットによって選択された配置とは無関係です。アナログ電圧は配置される前に低閾値および高域値と比較されます。

表 58 に、1 つまたは複数のチャンネル上でアナログウォッチドッグを有効にするためにどのようにして ADC_CR1 レジスタを設定するかを示します。

図 42. アナログウォッチドッグの保護領域

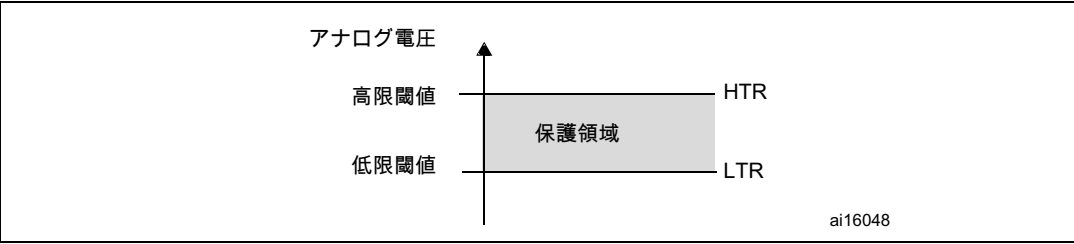


表 58. アナログウォッチドッグチャンネル選択

アナログウォッチドッグで保護されるチャンネル	ADC_CR1 レジスタ制御ビット (x = 無視)		
	AWDSGL ビット	AWDEN ビット	JAWDEN ビット
なし	x	0	0
すべてのインジェクトチャンネル	0	0	1
すべてのレギュラチャンネル	0	1	0
すべてのレギュラインジェクトチャンネル	0	1	1
シングル ⁽¹⁾ インジェクトチャンネル	1	0	1
シングル ⁽¹⁾ レギュラチャンネル	1	1	0
シングル ⁽¹⁾ レギュラまたはインジェクトチャンネル	1	1	1

1. AWDCH[4:0] ビットにより選択

12.3.8 スキャンモード

このモードは、アナログチャネルのグループをスキャンするために使用されます。

スキャンモードを選択するときは、ADC_CR1 レジスタの SCAN ビットをセットします。このビットがセットされると、ADC は ADC_SQRx レジスタ（レギュラチャネルの場合）または ADC_JSQR レジスタ（インジェクトチャネルの場合）で選択されたすべてのチャネルをスキャンします。スキャン中に ADC_CFG ビットを変化させないよう、変換対象のすべてのチャネルは、同じバンク内にある必要があります。シングル変換は、グループの各チャネルに対して行われます。各変換終了後、グループ内の次のチャネルが自動的に変換されます。ADC_CR2 レジスタの CONT ビットがセットされている場合、グループの最後の選択チャネルでレギュラチャネル変換は停止せず、選択チャネルの最初から続行します。

DMA ビットがセットされている場合、各レギュラチャネル変換後、ダイレクトメモリアクセス (DMA) コントローラを使用して、変換されたデータ (ADC_DR レジスタに格納されている) がレギュラチャネルグループからメモリに転送されます。

次の場合に、ADC_SR レジスタの EOC ビットがセットされます。

- 各レギュラグループシーケンスの終わりに、EOCS ビットが 0 クリアされる
- 各レギュラチャネル変換の終わりに、EOCS ビットが 1 にセットされる

インジェクトチャネルから変換されたデータは、常に ADC_JDRx レジスタへ格納されます。

12.3.9 インジェクトチャネルの管理

トリガインジェクト変換

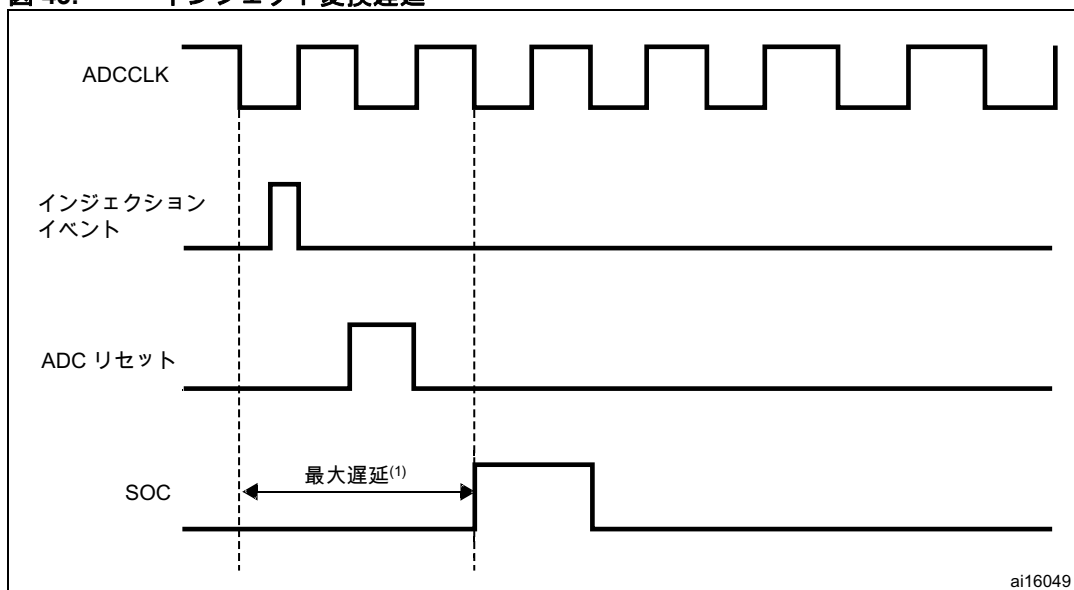
トリガインジェクションを使用するときは、ADC_CR1 レジスタの JAUTO ビットをクリアする必要があります。

1. 外部トリガによって、または、ADC_CR2 レジスタの JSWSTART ビットをセットすることによって、インジェクトチャネルグループの変換を開始します。
2. レギュラチャネルグループの変換中に外部インジェクトトリガが発生した場合または JSWSTART ビットがセットされた場合、現在の変換はリセットされ、インジェクトチャネルシーケンスはスキャンワンスモードに切り替わります。
3. その後、レギュラチャネルグループのレギュラ変換が、最後に中断されたレギュラ変換から再開されます。

インジェクト変換中にレギュライベントが発生した場合、インジェクト変換は中断されず、インジェクトシーケンスの終了時にレギュラシーケンスが実行されます。[図 43](#) に対応するタイミング図を示します。

注： トリガインジェクションを使用するときには、トリガイベントの間隔がインジェクションシーケンスより長くなるようにする必要があります。例えば、シーケンス長が 30 ADC クロックサイクルの場合 (2 クロック周期のサンプリング時間を持つ 2 つの変換)、トリガの最小間隔は 31 ADC クロックサイクルである必要があります。

図 43. インジェクト変換遅延



1. 最大遅延時間の値は、STM32L1xxxデータシートの電気特性に記載されています。

自動インジェクト変換

JAUTO ビットがセットされている場合、レギュラグループのチャネルの変換後、インジェクトグループのチャネルが自動的に変換されます。これを使用して、ADC_SQRx レジスタおよび ADC_JSQR レジスタでプログラミングされた最大 31 回の変換シーケンスを実行することができます。

このモードでは、インジェクトチャネルの外部トリガは無効である必要があります。

JAUTO ビットに加えて CONT ビットもセットされている場合、レギュラチャネルに続いてインジェクトチャネルが連続して変換されます。

注： 自動インジェクトモードと不連続モードを同時に使用することはできません。

12.3.10 不連続モード

レギュラグループ

このモードは、ADC_CR1 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。これを使用して、ADC_SQRx レジスタで選択された変換シーケンスの一部である n 変換の短いシーケンス ($n \leq 8$) を変換できます。 n の値は、ADC_CR1 レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADC_SQRx レジスタで選択された次の n 個の変換が開始され、シーケンス内のすべての変換が行われるまで続きます。合計シーケンス長は、ADC_SQR1 レジスタの L[4:0] ビットによって定義されます。

例：

$n = 3$ 、変換対象レギュラチャネル = 0、1、2、3、6、7、9、10

最初のトリガ：0、1、2 が変換されます

2 番目のトリガ：3、6、7 が変換されます

3 番目のトリガ：9、10 が変換され、EOC イベントが生成されます

4 番目のトリガ：0、1、2 が変換されます

注： レギュラグループが不連続モードで変換されるときには、ロールオーバーは発生しません。
 すべてのサブグループが変換されると、次のトリガでは、最初のサブグループの変換が開始されます。
 上の例では、4 番目のトリガが最初のサブグループ内のチャンネル 0、1、2 を変換します。

インジェクトグループ

このモードは、ADC_CR1 レジスタの JDISCEN ビットをセットすることで有効になります。これを使用して、ADC_JSQR レジスタで選択された変換シーケンスの一部である n 変換の短いシーケンス ($n \leq 3$) を変換できます。 n の値は、ADC_CR1 レジスタの DISCNUM[2:0] ビットに書き込むことによって指定されます。

外部トリガが発生すると、ADC_JSQR レジスタで選択された次のチャンネル変換が開始され、シーケンス内のすべての変換がおこなわれるまで続きます。合計シーケンス長は、ADC_JSQR レジスタの JL[1:0] ビットによって定義されます。

例：

- $n = 1$ 、変換対象インジェクトチャンネル = 1、2、3
- 最初のトリガ：チャンネル 1 が変換されます。
- 2 番目のトリガ：チャンネル 2 が変換されます。
- 3 番目のトリガ：チャンネル 3 が変換され、EOC および JEOC イベントが生成されます。
- 4 番目のトリガ：チャンネル 1 が変換されます。

注： すべてのインジェクトチャンネルが変換されると、次のトリガでは、最初のインジェクトチャンネルの変換が開始されます。上の例では、4 番目のトリガで最初のインジェクトチャンネル 1 が再変換されます。
 自動インジェクトモードと不連続モードを同時に使用することはできません。
 不連続モードは、レギュラグループとインジェクトグループに同時に設定することはできません。

12.4 データアライメント

ADC_CR2 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。データは右詰めまたは左詰めが可能です (図 44 と 図 45 参照)。

インジェクトグループチャンネルの変換後のデータ値は、ADC_JOFRx レジスタに書き込まれたユーザ定義オフセットだけ減算されるので結果は負の値になることもあります。SEXT ビットは、拡張符号値です。

レギュラグループのチャンネルの場合、オフセットは引かれられないので、12 ビットのみが有効です。

図 44. 右詰め 12 ビットデータ

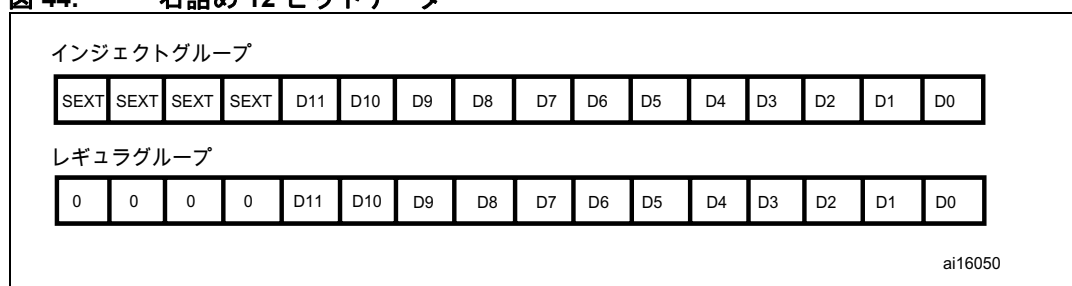
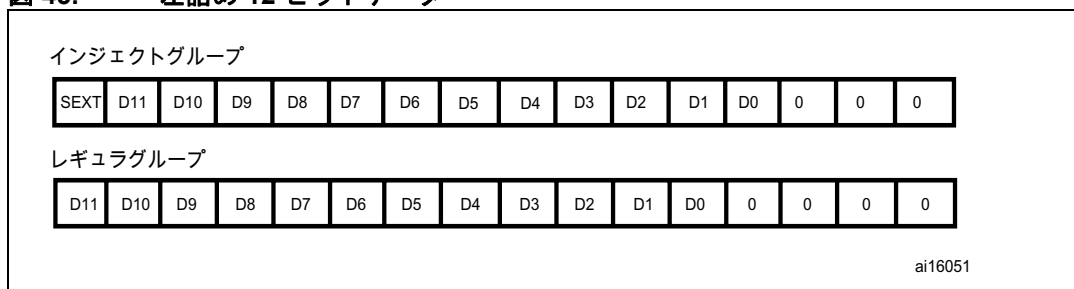
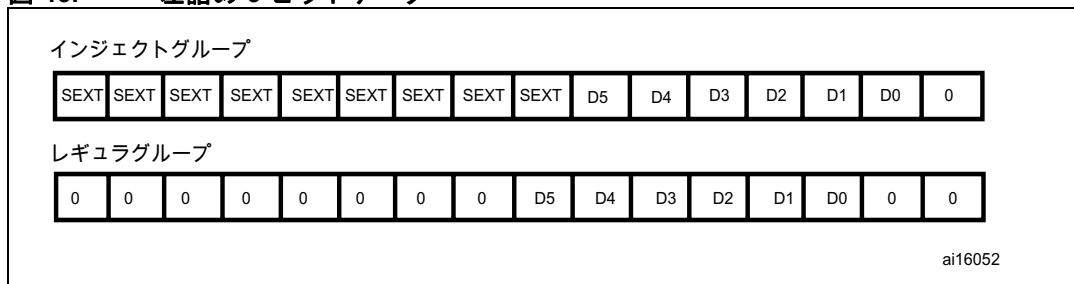


図 45. 左詰め 12 ビットデータ



特殊なケース：左詰めの場合、図 46 に示すようにバイト単位に配置される場合で分解能が 6 ビットに設定されている場合を除きデータはハーフワード単位に配置されます。

図 46. 左詰め 6 ビットデータ



12.5 チャネル単位で設定可能なサンプリングタイム

ADC は、一定の ADCCLK サイクル数間だけ入力電圧をサンプリングしますが、この時間は、ADC_SMPRx レジスタ (x=1 から 3) の SMP[2:0] ビットを使用して変更できます。ある特定のバンクにおける各チャネルは、異なるサンプリング時間でサンプリングすることができます。しかしながら、サンプリングタイムの選択は 2 つのバンク間で共用されます。

合計変換時間は、次のように計算されます。

$$T_{\text{conv}} = \text{サンプリングタイム} + \text{チャネル変換時間}$$

例：

ADCCLK = 16 MHz かつサンプリングタイム = 4 サイクルの場合：

$$T_{\text{conv}} = 4 + 12 = 16 \text{ サイクル} = 1 \mu\text{s} \text{ (12 ビット変換)}$$

$$T_{\text{conv}} = 4 + 7 = 11 \text{ サイクル} = 685 \text{ ns} \text{ (6 ビット変換)}$$

12.6 外部トリガによる変換

変換は、外部イベント（タイマキャプチャ、EXTI ラインなど）によってトリガできます。EXTEN[1:0] 制御ビット（レギュラ変換の場合）または JEXTEN[1:0] ビット（インジェクト変換の場合）が “0b00” 以外の場合、選択したエッジで外部イベントによって変換をトリガすることができます。表 59 に、EXTEN[1:0] 値、JEXTEN[1:0] 値、トリガエッジ間の対応を示します。

表 59. トリガエッジ検出の設定

ソース	EXTEN[1:0] / JEXTEN[1:0]
トリガー検出を無効化	00
立ち上がりエッジでの検出	01
立ち下がりエッジでの検出	10
立ち上がりと立ち下がりの両エッジでの検出	11

注： 外部トリガのエッジ検出は、即座に変更することができます。

EXTSEL[3:0] および JEXTSEL[3:0] 制御ビットを使用してレギュラおよびインジェクトグループの変換をトリガできるイベントを 16 のイベントから選択します。

表 60 にレギュラ変換に使用できる外部トリガを示します。

表 60. レギュラチャネルの外部 トリガ

ソース	タイプ	EXTSEL[3:0]
TIM9_CC2 イベント	オンチップタイマからの内部信号	0000
TIM9_TRGO イベント		0001
TIM2_CC3 イベント		0010
TIM2_CC2 イベント		0011
TIM3_TRGO イベント		0100
TIM4_CC4 イベント		0101
TIM2_TRGO イベント		0110
TIM3_CC1 イベント		0111
TIM3_CC3 イベント		1000
TIM4_TRGO イベント		1001
TIM6_TRGO イベント		1010
予約済み		1011
予約済み	NA	1100
予約済み		1101
予約済み		1110
EXTI ライン 11	外部ピン	1111

表 61 に、インジェクト変換に対する外部 トリガを示します。

表 61. インジェクトチャネルの外部 トリガ

ソース	タイプ	EXTSEL[3:0]
TIM9_CC1 イベント	オンチップタイマからの内部信号	0000
TIM9_TRGO イベント		0001
TIM2_TRGO イベント		0010
TIM2_CC1 イベント		0011
TIM3_CC4 イベント		0100
TIM4_TRGO イベント		0101
TIM4_CC1 イベント		0110
TIM4_CC2 イベント		0111
TIM4_CC3 イベント		1000
TIM10_CC1 イベント		1001
TIM7_TRGO イベント		1010
予約済み	NA	1011
予約済み		1100
予約済み		1101
予約済み		1110
EXTI ライン 15	外部ピン	1111

レギュラグループ変換は、インジェクトトリガによって中断することが可能です。

注： トリガの選択は動作中に変更することができます。ただし、これを行うと、トリガ検出が 2 PCLK サイクル間無効化されます。これは、変更時の誤検出を防止するためです。

トリガイベント間の間隔は、次より長い必要があります。

- レギュラ変換のシーケンス
- シーケンス + インジェクト変換用の 1 ADCCLK サイクル

例えば、シーケンス 長が 32 ADC クロックサイクル (4 クロック周期のサンプリング時間を持つ 2 つの変換) の場合、レギュラトリガの最小間隔は 32 ADC クロックサイクルより、インジェクトトリガの間隔は 33 ADC クロックサイクルより、それぞれ長い必要があります。

12.7 変換の中止

12.7.1 インジェクトチャネル

インジェクト変換または変換のシーケンスは、JSQR レジスタへの書き込みにより停止することができます (JL[1:0] ビットフィールドに現在値を書き込む必要があります)。次に、進行中のインジェクト変換が中止され、ペンディング中のトリガがリセットされます。新しいハードウェアまたはソフトウェアトリガが発生すると、新しいインジェクト変換を開始することができます。

インジェクト変換の中止後、システムは新しいインジェクト変換を開始する前に数クロックサイクルを必要とします (3 から 5 ADC クロックサイクル + 2 から 5 APB クロックサイクル)。この条件を満たすため、JCNr=0 となる前に JSWSTART をセットしないでください。

12.7.2 レギュラチャネル

レギュラ変換または変換のシーケンスは、SQR1 から SQR5 レジスタへの書き込みにより停止することができます (SQR1 へ書き込む場合、L[4:0] ビットフィールドに現在値を書き込む必要があります)。その後、ADC はインジェクト変換の場合と同じ動作をします ([セクション 12.7.2 : レギュラチャネル](#)参照)。

複数の SQRi レジスタに書き込みを行い、新しいシーケンスを設定する場合、異なる書き込みアクセスの間に変換を開始することはできません。この場合、次のシーケンスを使用する必要があります。

1. EXTEN ビットに 00 を書き込んで外部 トリガを無効化します (外部 トリガを使用の場合)
2. シーケンス設定を変更します (SQRi レジスタへ書き込みます)
3. ADC_SR レジスタで RCNR=0 になるのを待ちます。
4. 外部 トリガを有効化するか、または SWSTART ビットをセットします。

12.8 変換分解能

ADC の分解能を下げることによって、高速変換を行うことができます。RES[1:0] ビットを使って、データレジスタのビット数を選択します。各分解能に対する最小変換時間は、サンプリングタイムが 4 サイクルの場合、次のようになります。

- 12 ビット分解能の場合 : $12 + 4 = 16$ サイクル
- 10 ビット分解能の場合 : $11 + 4 = 15$ サイクル
- 8 ビット分解能の場合 : $9 + 4 = 13$ サイクル
- 6 ビット分解能の場合 : $7 + 4 = 11$ サイクル

12.9 ハードウェアフリーズと低速変換用の遅延挿入モード

APB クロック速度がデータレートを管理するために不十分な場合、変換と変換の間に遅延を挿入してこのデータレートを下げることができます。変換中は、遅延中に発生するトリガイイベント（変換の同じグループに対して）が無視されるので、各レギュラ変換の後およびインジェクト変換の各シーケンスの後に遅延が挿入されます。

異なるグループの変換の間には（インジェクト変換の後にレギュラ変換が続く場合または逆順の場合）、遅延は挿入されません：

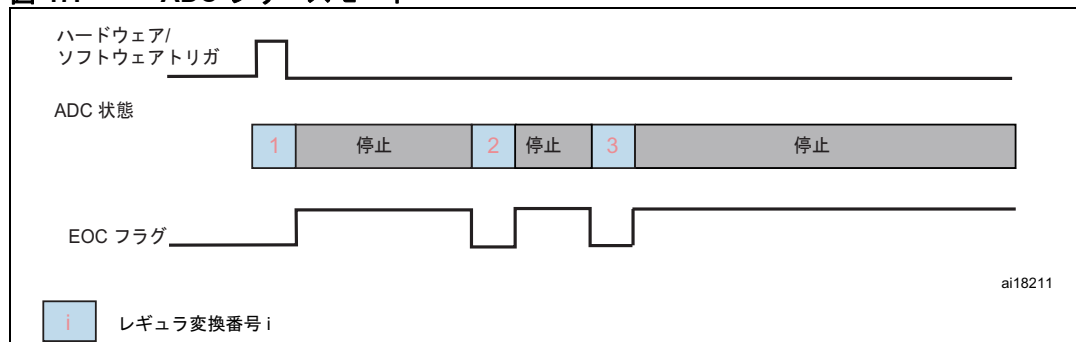
- レギュラ変換の遅延中にインジェクトトリガが発生すると、インジェクト変換が直ちに開始されます。
- インジェクトシーケンスにより中断された後にレギュラ変換が再開される場合、前のレギュラ変換の遅延が終わると、直ちに開始されます。

この動作は自動インジェクトモードの動作と少し異なります。自動インジェクトモードでは、前のインジェクト変換の遅延が終わったときにのみ新しいレギュラ変換を開始することができます。これは、ソフトウェアが与えられたシーケンスの全データを読み出した後に新しいシーケンスを確実に開始できるようにするためです。このモードでは、レギュラ変換に続く遅延中にレギュラトリガ発生すると、そのレギュラトリガは無視されます。ただし、この遅延の後に発生した場合、さらにインジェクトシーケンスまたはそれに続く遅延中に発生した場合にも、ペンディングと見なされます。その後、インジェクトシーケンスの遅延の終わりに、変換が開始されます。

遅延の長さは、ADC_CR2 レジスタの DELS[2:0] ビットを使って設定します。次の 2 つのケースを考慮する必要があります。

- ADC フリーズモード：
DELS[2:0]=001 の場合、同じグループの前の全データが処理された後にのみ新しい変換を開始することができます。
 - － レギュラ変換の場合：ADC_DR レジスタを読み出した後、または EOC ビットをクリアした場合
 - － インジェクト変換の場合：JEOC ビットをクリアしたとき
- ADC 遅延挿入モード：
DELS[2:0]>001 の場合、同じグループの前の変換が終わった後に所定数の APB クロックサイクルが経過した後にのみ新しい変換を開始することができます。

図 47. ADC フリーズモード

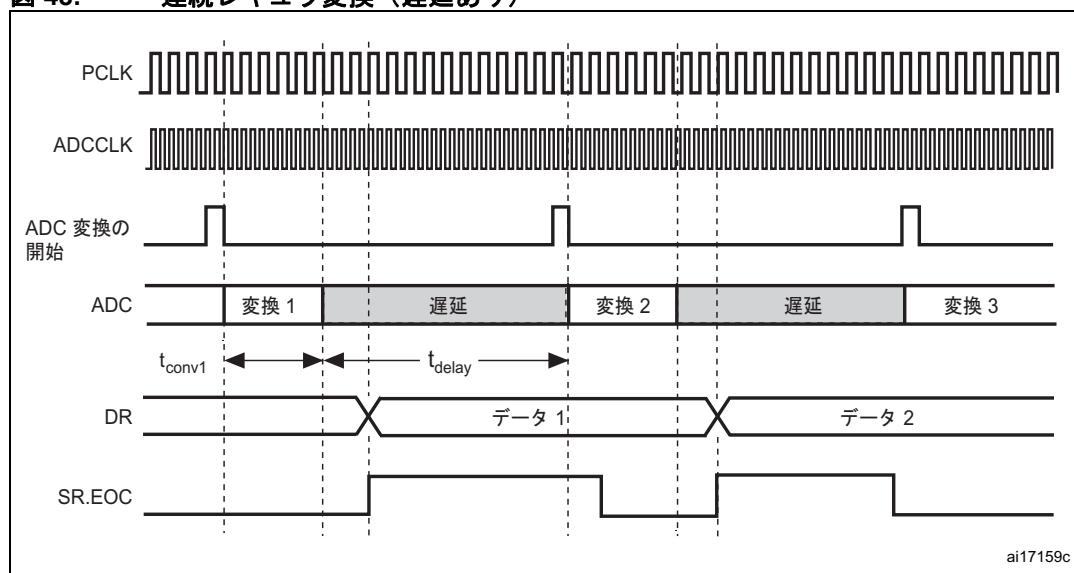


12.9.1 各レギュラ変換後の遅延挿入

有効化された場合、各レギュラ変換の終わりで、かつ新しいレギュラ変換開始前に遅延が挿入されます。これにより、新しいレギュラ変換が完了する前に変換したデータを ADC_DR レジスタへ読み込む時間が確保されます。遅延の長さは、DELS[2:0] ビットで設定します。図 48 に、連続 レギュラ変換の例を示します。ここでは、10 PCLK サイクルの遅延が各変換後に挿入されています。

注： `ADC_CR2_EOCS = 1` の場合、レギュラグループ変換の各シーケンスの後に遅延が挿入されます。

図 48. 連続レギュラ変換（遅延あり）

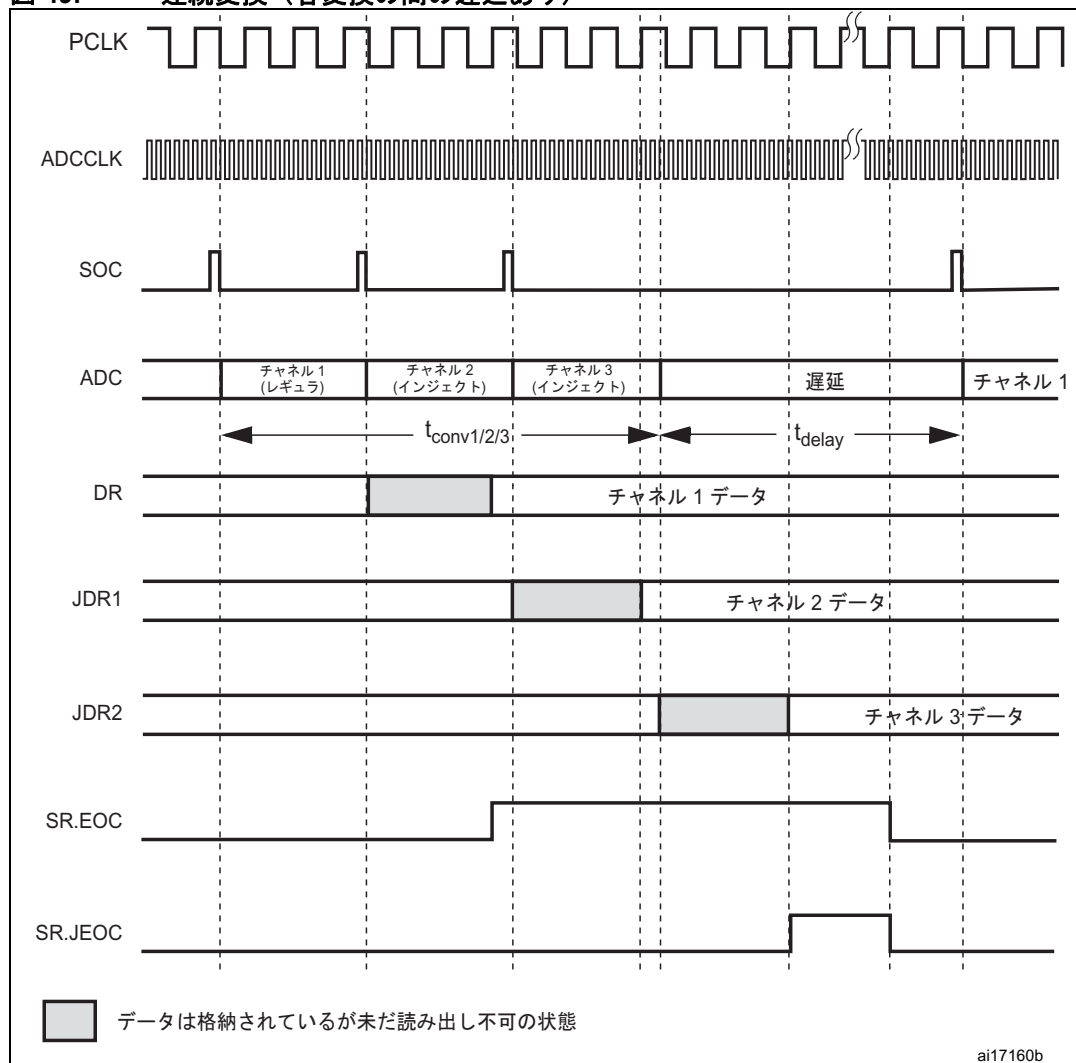


1. t_{conv1} : サンプル時間と変換時間を含みます (例えば、最小サンプリングタイムで 16 ADC クロックサイクル)
2. t_{delay} : 変換の終わってから次の変換の開始までの遅延 (DELS[2:0] ビットで設定された PCLK 周期数) + 変換終了を同期化する遅延 (0 から 1 PCLK クロックサイクル) + 遅延終了を同期化する遅延 (2 または 3 ADC クロックサイクル)。

12.9.2 自動インジェクト変換の各シーケンスの後への遅延挿入

有効化された場合、インジェクト変換の各シーケンスの終わりに遅延が挿入されます。最大 5 個の変換結果を ADC_DR と ADC_JDRx レジスタへ保存することができます。遅延の長さは、DELS[2:0] ビットで設定します。図 49 に、連続変換 (CONT ビットをセット) の例を示します。ここでは、インジェクト変換の各シーケンスの後に遅延が挿入されています。ここでは、JAUTO ビットがセットされ、シーケンスは最後のインジェクト変換の後に終わります (シーケンスは、1 レギュラ変換 + 2 インジェクト変換で構成されています)。

図 49. 連続変換 (各変換の間の遅延あり)



1. $t_{\text{conv}1/2/3}$: チャンネル 1、2、3 のサンプリング時間と変換時間を含みます。
2. t_{delay} : 前のシーケンスの終わりから新しいシーケンスの開始までの遅延 (DELS ビットで設定された PCLK 周期数) + 変換終了を同期化する遅延 (0 から 1 PCLK クロックサイクル) + 遅延終了を同期化する遅延 (2 または 3 ADC クロックサイクル)。

12.10 省電力

ADC が変換していない場合の消費電力を削減するため、ADC のパワーオンとパワーオフをハードウェアで管理することができます。次の場合 ADC をパワーダウンさせることができます。

- 前述の遅延中 (PDD ビットのセット時) この場合、ADC は遅延の終わりにパワーアップします。および/または
- ADC がトリガイイベント待ちのとき (PDI ビットのセット時) この場合、ADC は次のトリガイイベントでパワーアップします。

ADC は、変換が実際に開始される前に所定のスタートアップ時間を必要とします。自動電源制御モードの選択時または遅延の設定時に、このスタートアップ時間を考慮する必要があります。このため、各変換の後に遅延を挿入して 1 つずつ各変換を開始するより、複数の変換のシーケンスを開始するために複数のチャンネルをスキャンし、シーケンスの後に電力消費を停止する方が効率良くなります (消費電力の点から、また可能な場合)。

一定の変換のシーケンスに対して、最初の変換の開始前に ADCCLK クロックを有効化し、EOC ビット (またはインジェクトチャンネルの場合 JEOC ビット) がセットされるまでこのクロックを入力し続ける必要があります。

図 50、図 51、図 52 に、様々な構成での電源管理の例を示します。これらすべての例で ADON=1。

図 50. 自動パワーダウン制御：例 1

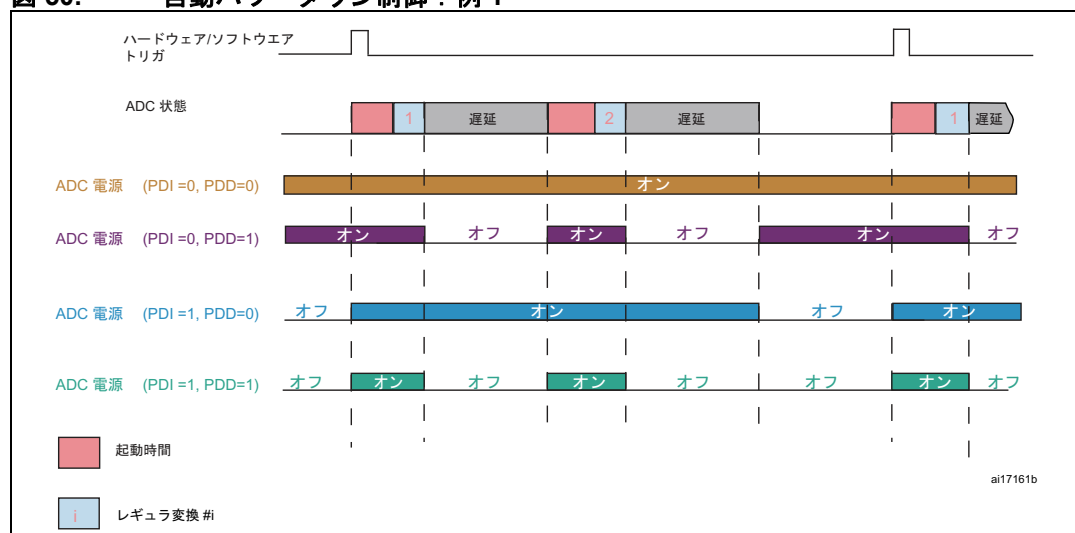


図 51. 自動パワーダウン制御：例 2

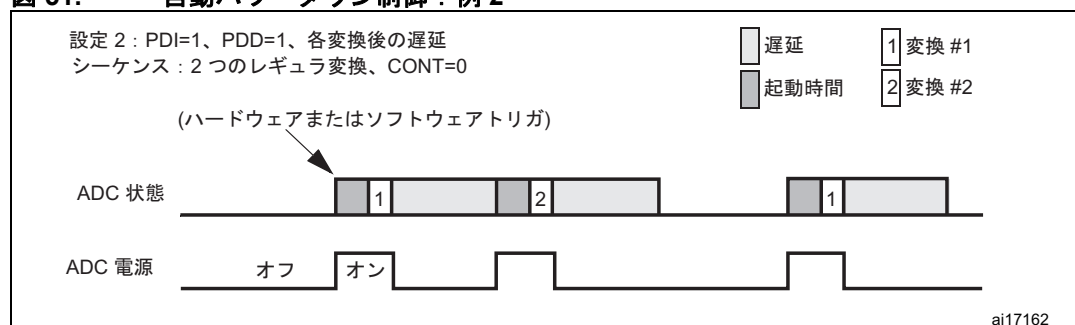
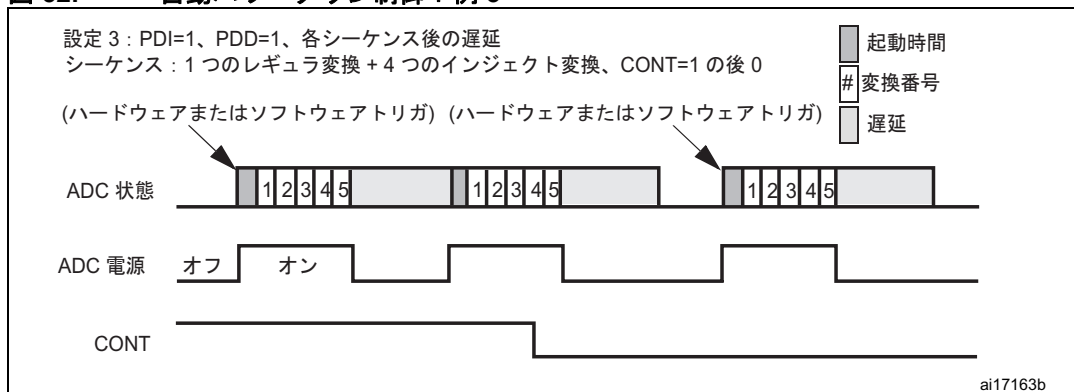


図 52. 自動パワーダウン制御：例 3



12.11 データ管理とオーバーラン検出

12.11.1 DMA の使用

変換されたレギュラチャネルの値は一意なデータレジスタに格納されるので、複数のレギュラチャネルの変換では DMA を使用すると便利です。これによって、ADC_DR レジスタにすでに格納されているデータの損失を防ぐことができます。

DMA モードが有効 (ADC_CR2 レジスタの DMA ビットが 1 に設定されている) な場合、レギュラチャネルが変換されるごとに 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。

これにもかかわらず、データが失われる場合 (オーバーラン)、ADC_SR レジスタの OVR ビットがセットされ割り込みが生成されます (OVRIE イネーブルビットがセットされている場合)。続いて DMA 転送が無効になり、その後の DMA リクエストは受け付けられません。この場合、DMA リクエストが行われると、実行中のレギュラ変換は中断され、その後のレギュラトリガは無視されます。そのときは、必要な変換済みチャネルデータを正しいメモリ位置へ転送するために OVR フラグおよび使用している DMA ストリームの DMAEN ビットをクリアして DMA と ADC を再初期化する必要があります。その後初めて、変換を再開することができ、データ転送を再び有効にすることができます。インジェクトチャネル変換はオーバーランエラーの影響を受けません。

DMA モードで OVR = 1 の場合、最後の有効なデータが転送されたあとに DMA リクエストがブロックされます。つまり、RAM へ転送されたデータはすべて有効とみなすことができます。

最後の DMA 転送 (DMA コントローラの DMA_SxRTR レジスタに設定されている転送数) が終了すると、

- ADC_CR2 レジスタの DDS ビットが 0 にクリアされている場合、DMA コントローラへの新規の DMA リクエストは発生しません (これによりオーバーランエラーの発生を防ぎます)。ただし、DMA ビットはハードウェアによってクリアされることはありません。新たな転送を開始するにはそこに 0 を書き込み、その後 1 を書き込む必要があります。
- DDS ビットが 1 にセットされている場合、リクエストを生成は継続することができます。これにより DMA をダブルバッファサーキュラモードに設定することができます。

12.11.2 DMA を使用しない変換シーケンスの管理

変換が十分遅い場合、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、変換シーケンスの終わりだけではなく各変換が終わるたびに EOC ステータスビットをセットするため、ADC_CR2 レジスタの EOCS ビットをセットする必要があります。EOCS = 1 のとき、オーバーラン検出は自動的に有効になります。したがって、変換が終了するごとに EOC がセットされ ADC_DR レジスタを読み出すことができます。オーバーラン 管理は、DMA を使用する場合と同じです。

12.11.3 全データを読み出さない変換

毎回データの読み出しをせずに ADC に 1 つまたは複数のチャネルを変換させると便利な場合があります (たとえば、アナログウォッチドッグがある場合)。そのためには、DMA を無効 (DMA = 0) にし、シーケンスの終了時にのみ EOC ビットをセットする必要があります (EOCS = 0)。この構成では、前の変換結果が読み出されずに変換が終了しても、オーバーランエラーは報告されません。

12.11.4 オーバーラン検出

オーバーラン検出は常に有効です。データが APB クロックに同期する前に行われます。

注： **オーバーランエラーは、レギュラチャネル変換のみで発生します。**

変換の終わりに、変換結果はデータレジスタへ転送 (APB クロックドメイン内の ADC_DR) されるまで、中間バッファに保存されます (ADC クロックドメイン内)。前のデータの転送前に新しいデータが到着すると、その新しいデータが失われて、オーバーランエラーが検出されます。ADC_SR レジスタの OVR ビットがセットされ、OVRIE 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。

これは次の 2 つのケースで発生します。

- APB クロック周波数に対して遅延が不適切に設定されているか (データの同期化にとって遅延が短過ぎる)、または
- ADC_DR レジスタが空でないため (DMA=1 または EOCS=1)、前のデータが APB クロックに同期化できなかった。実際、これらのモードでは、ADC_DR レジスタ値を上書きできなかったため、レジスタは常に最後の有効データを保持しています。ADC_DR を読み出すことにより、または ADC_SR レジスタの EOC ビットをクリアすることにより、ADC_DR を空にすることができます。

注： **DMA により転送された最後のデータが、非常に遅れて読み出されたとき、次のデータが失われるので、DMA (または EOCS) をクリアした直後にオーバーランが検出されることがあります。**

OVR ビットをクリアした後、ADC_SR レジスタで RCNR=0 になるまで、ソフトウェアから新しいレギュラ変換を開始することはできません。

12.12 温度センサと内部基準電圧

温度センサを使って、デバイス接合温度 (T_J) を測定することができます。温度センサは、内部で ADC_IN16 入力チャンネルに接続されていて、センサの出力電圧をデジタル値に変換することができます。使用しないときには、このセンサをパワーダウンモードにすることができます。

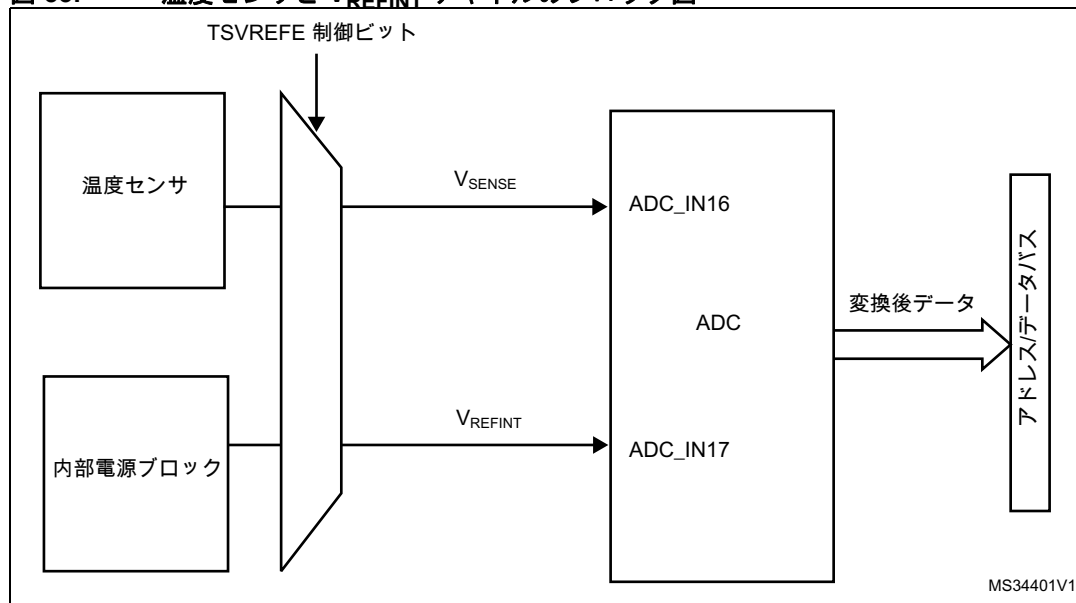
温度センサ出力電圧は温度に比例して変化しますが、特性は製造プロセス変動のためにチップごとに大幅に変動します。温度センサ精度を改善するため（特に絶対温度測定の場合）、製造検査時に ST によって較正值が各デバイスごとに測定され、システムメモリ領域に格納されます。詳細については、特定のデバイスデータシートを参照してください。

内部電圧基準 (V_{REFINT}) は、ADC とコンパレータに対して安定な（バンドギャップ）電圧出力を供給します。 V_{REFINT} は内部で ADC_IN17 入力チャンネルに接続されています。 V_{REFINT} の電圧は、ST により製造テストで、各デバイスごとに高精度に測定され、システムメモリ領域に格納されます。

図 53 に、温度センサ、内部電圧基準、ADC との間の接続のブロック図を示します。

ADC_IN16（温度センサ）の変換と ADC_IN17（ V_{REFINT} ）の変換を有効化するには、TSVREFE ビットをセットする必要があります。

図 53. 温度センサと V_{REFINT} チャンネルのブロック図



温度の読み出し

1. ADC_IN16 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 (T_{S_temp}) で規定されたサンプリングタイムを選択します。
3. ADC_CCR レジスタの TSVREFE ビットをセットして、パワーダウンモードから温度センサをウェイクアップさせて、安定化時間 (t_{START}) の間待ちます。
4. ADC 変換を開始します。
5. ADC_DR レジスタから結果の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。

$$\text{Temperature} = \frac{110^{\circ}\text{C} - 30^{\circ}\text{C}}{TS\_CAL2 - TS\_CAL1} \times (TS\_DATA - TS\_CAL1) + 30^{\circ}\text{C}$$

ここで:

- TS_CAL2 は、110°C で取得した温度センサ較正值。
- TS_CAL1 は、30°C で取得した温度センサ較正值。
- TS_DATA は、ADC で変換した実際の温度センサ出力値。

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、ADON ビットと TSVREFE 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へ加えられる V_{DDA} 電源電圧は変動することがあり、正確な値は不明です。内蔵電圧基準 (V_{REFINT}) と、製造プロセス時に $V_{DDA} = 3\text{ V}$ で ADC により取得したキャリブレーションデータを使って、実際の V_{DDA} 電圧レベルを評価することができます。

次式によりデバイスに供給する実際の V_{DDA} 電圧を求めることができます。

$$V_{DDA} = 3\text{ V} \times V_{REFINT\_CAL} / V_{REFINT\_DATA}$$

ここで、

- VREFINT_CAL は、 V_{REFINT} 較正值
- VREFINT_DATA は、ADC で変換した実際の V_{REFINT} 出力値

電源基準 ADC 変換値から絶対電圧値への変換

ADC は、アナログ電源電圧と変換したチャネルに加えられる電圧との間の比に対応するデジタル値を出力するようにデザインされています。大部分のアプリケーションでは、この比を V_{DDA} に無関係な電圧へ変換することが必要です。 V_{DDA} が既知で、ADC 変換値が右詰め of アプリケーションでは、次式からこの絶対値を求めることができます。

$$V_{CHANNELx} = \frac{V_{DDA}}{FULL\_SCALE} \times ADC\_DATA_x$$

V_{DDA} 値が未知 of アプリケーションでは、内部電圧基準を使う必要があり、 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](#) に示す式で置き換えると、次式が得られます。

$$V_{CHANNELx} = \frac{3\text{ V} \times V_{REFINT\_CAL} \times ADC\_DATA_x}{V_{REFINT\_DATA} \times FULL\_SCALE}$$

ここで:

- VREFINT_CAL は、 V_{REFINT} 較正值
- ADC_DATA_x は、チャネル x の ADC で測定した値 (右詰め)
- VREFINT_DATA は、ADC で変換した実際の V_{REFINT} 出力値
- FULL_SCALE は、ADC 出力の最大デジタル値。例えば、12 ビット分解能の場合、 $2^{12} - 1 = 4095$ で、8 ビット分解能の場合、 $2^8 - 1 = 255$ 。

注: ADC 測定が 12 ビット右詰め以外の出力フォーマットを使って行われた場合、すべてのパラメータを互換フォーマットへ変換した後に、計算を行います。

12.13 内部基準電圧 (V_{REFINT}) の変換

内部基準電圧は、内部で VREFINT チャンネルに接続されています。このアナログ入力チャンネルは、内部基準電圧をデジタル値に変換するために使用されます。

内部基準電圧を有効にするには、ADC_CCR レジスタの TSVREFE ビットをセットする必要があります。この基準電圧は、変換が必要な場合にのみ有効にされなければなりません。

内部基準電圧は出荷時に測定され、ADC 変換結果が特定のデータアドレス (VREFINT_Factory_CONV バイト) に格納されています。

12.14 ADC 割り込み

アナログウォッチドッグステータスビットがセットされ、かつオーバーランステータスビットがセットされていれば、レギュラおよびインジェクトグループの変換終了時、割り込みを生成することができます。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。

ADC_SR レジスタには他に 5 つのフラグがありますが、これらは割り込みには無関係です。

- JCNR (インジェクトチャンネル非レディ)
- RCNR (レギュラチャンネル非レディ)
- ADONS (ADON ステータス)
- JSTRT (インジェクトグループチャンネルの変換開始)
- STRT (レギュラグループチャンネルの変換開始)

図 54. ADC フラグと割り込み

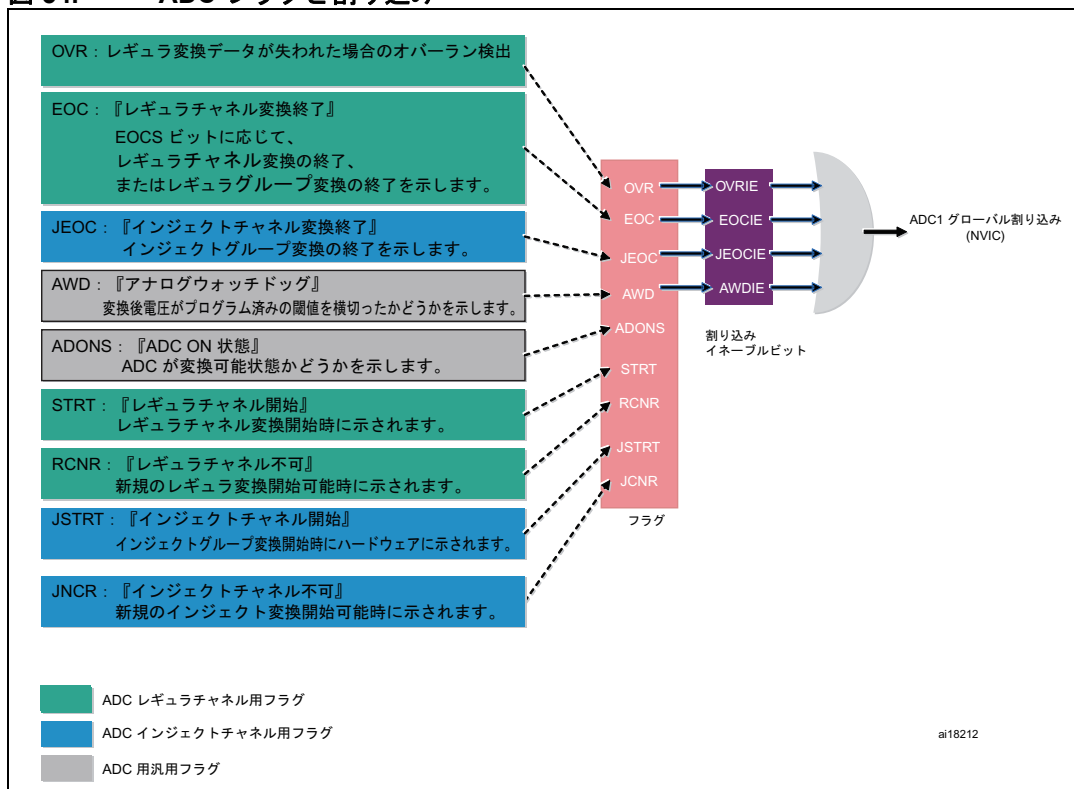


表 62. ADC 割り込み

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
レギュラグループの変換終了	EOC	EOCIE
インジェクトグループの変換終了	JEOC	JEOCIE
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
オーバーラン	OVR	OVRIE

12.15 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。

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。

12.15.1 ADC ステータスレジスタ (ADC_SR)

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						JCNR	RCNR	予約済み	ADONS	OVR	STRT	JSTRT	JEOC	EOC	AWD
						r	r		r	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

ビット 31:10 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 **JCNR** : インジェクトチャンネル 非レディ

JSQR レジスタの書き込み後、このビットはハードウェアからセット/クリアされます。新しいインジェクト変換開始の可否を示します (JSWSTART ビットをセット)。

0 : インジェクトチャンネルレディ

1 : インジェクトチャンネル非レディ、JSWSTART はセットしない

ビット 8 **RCNR** : レギュラチャンネル非レディ

SQRx レジスタの 1 つに書き込んだ後または OVR ビットをクリアした後に、このビットはハードウェアからセット/クリアされます。新しいレギュラ変換開始の可否を示します (SWSTART ビットをセット)。

0 : レギュラチャンネルレディ

1 : インジェクトチャンネル非レディ、SWSTART はセットしない

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 **ADONS** : ADC ON ステータス

このビットはハードウェアからセット/クリアされて、ADC の変換レディかどうかを示します。

0 : ADC 非レディ

1 : ADC 変換レディ外部トリガを有効化でき、SWSTART ビットと JSWSTART ビットをセットすることができます。

ビット 5 OVR : オーバーラン

レギュラ変換データが失われたとき、このビットはハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます。DMA = 1 または EOCS = 1 の場合にのみ、オーバーラン検出が有効化されます。

- 0 : オーバーランなし
- 1 : オーバーラン発生

ビット 4 STRT : レギュラチャネル開始フラグ

レギュラチャネル変換が開始されると、このビットはハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます。

- 0 : レギュラチャネル変換の開始なし
- 1 : レギュラチャネル変換が開始された

ビット 3 JSTRT : インジェクトチャネル開始フラグ

このビットは、インジェクトグループの変換が開始すると、ハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます。

- 0 : インジェクトグループ変換の開始なし
- 1 : インジェクトグループ変換が開始された

ビット 2 JEOC : インジェクトチャネル変換終了

このビットは、グループ内のインジェクトチャネル変換がすべて終了したときに、ハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます。

- 0 : 変換は未完了
- 1 : 変換が完了

ビット 1 EOC : レギュラチャネル変換終了

このビットは、レギュラグループチャネル変換の終わりにハードウェアによってセットされます。このビットは、ソフトウェアでクリアされるか、または ADC_DR レジスタの読み出しによりクリアされます。

- 0 : 変換は未完了 (EOCS=0)、または変換シーケンスは未完了 (EOCS=1)
- 1 : 変換が完了 (EOCS=0)、または変換シーケンスが完了 (EOCS=1)

ビット 0 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 レジスタに設定された値を超えたときに、ハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます。

- 0 : アナログウォッチドッグイベントなし
- 1 : アナログウォッチドッグイベント発生

12.15.2 ADC 制御レジスタ 1 (ADC_CR1)

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					OVR1E	RES[1:0]		AWDEN	JAWDEN	予約済み					PDI	PDD
					rw	rw	rw	rw	rw						rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
DISCNUM[2:0]			JDISCEN	DISC EN	JAUTO	AWDSGL	SCAN	JEOCIE	AWDIE	EOCIE	AWDCH[4:0]					
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

ビット 31:27 予約済みであり、リセット値に保持する必要があります。

ビット 26 **OVR1E** : 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、ソフトウェアでセット/クリアします。
 0 : オーバーラン割り込みを無効化
 1 : オーバーラン割り込みを有効化OVR 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 25:24 **RES[1:0]** : 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 ($T_{CONV} = 12 \text{ ADCCLK サイクル}$)
 01 : 10 ビット ($T_{CONV} = 11 \text{ ADCCLK サイクル}$)
 10 : 8 ビット ($T_{CONV} = 9 \text{ ADCCLK サイクル}$)
 11 : 6 ビット ($T_{CONV} = 7 \text{ ADCCLK サイクル}$)
 このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 23 **AWDEN** : レギュラチャネルのアナログウォッチドッグイネーブル

このビットは、ソフトウェアによってセット/クリアします。
 0 : レギュラチャネルのアナログウォッチドッグを無効化
 1 : レギュラチャネルのアナログウォッチドッグを有効化

ビット 22 **JAWDEN** : インジェクトチャネルのアナログウォッチドッグイネーブル

このビットは、ソフトウェアによってセット/クリアします。
 0 : インジェクトチャネルのアナログウォッチドッグを無効化
 1 : インジェクトチャネルのアナログウォッチドッグを有効化

ビット 21:18 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 **PDI** : アイドルフェーズ時のパワーダウン

このビットはソフトウェアによって書き込まれ、クリアされます。ADON=1 のとき、ADC 非変換時 (ハードウェアまたはソフトウェアトリガイベント待ち) のパワーアップまたはパワーダウンを指定します。
 0 : 開始イベント待ちのとき、ADC はパワーアップ
 1 : 開始イベント待ちのとき、ADC はパワーダウン

注 : このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 16 **PDD** : 遅延フェーズ時のパワーダウン

このビットはソフトウェアから書き込まれ、クリアされます。ADON=1 の場合、遅延が挿入されたとき (DELS ビット)、2 つの変換 (変換のシーケンス) の間での ADC のパワーアップまたはパワーダウンを指定します。
 0 : 遅延時に ADC はパワーアップ
 1 : 遅延時に ADC はパワーダウン

注 : このビットは、ADON=0 のときにのみ書き込む必要があります。



ビット 15:13 DISCNUM[2:0] : 不連続モードチャンネルカウント

これらのビットは、外部トリガの受信後に、不連続モードで変換されるチャンネル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。

000 : 1 チャンネル

001 : 2 チャンネル

...

111 : 8 チャンネル

注： このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 12 JDISCEN:インジェクトチャンネルの不連続モード

このビットは、インジェクトチャンネルグループに対する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 : インジェクトチャンネルの不連続モードを無効化

1 : インジェクトチャンネルの不連続モードを有効化

注： このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 11 DISCEN:レギュラチャンネルの不連続モード

このビットは、レギュラチャンネルに対する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 : レギュラチャンネルの不連続モードを無効化

1 : レギュラチャンネルの不連続モードを有効化

注： このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 10 JAUTO : 自動インジェクトグループ変換

このビットは、レギュラグループ変換の後、自動インジェクトグループ変換を有効/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 : 自動インジェクトグループ変換を無効化

1 : 自動インジェクトグループ変換を有効化

注： このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 9 AWDSGL : スキャンモードでの新しいチャンネルのウォッチドッグのイネーブル

このビットは、AWDCH[4:0] ビットによって識別されたチャンネルに対するアナログウォッチドッグを有効/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 : 全チャンネルのアナログウォッチドッグを有効化

1 : シングルチャンネルのアナログウォッチドッグを有効化

ビット 8 スキャン: スキャンモード

このビットは、スキャンモードを有効/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。スキャンモードでは、ADC_SQRx および ADC_JSEQR レジスタを通じて選択された入力に変換されます。

0 : スキャンモードを無効化

1 : スキャンモードを有効化

注： このビットは、ADON=0 のときにのみ書き込む必要があります。

ビット 7 JEOCIE : インジェクトチャンネルの割り込みイネーブル

このビットは、インジェクトチャンネルの変換割り込み終了を有効/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 : JEOC 割り込みを無効化

1 : JEOC 割り込みを有効化 JEOC 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 6 **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。スキャンモードでは、このビットが有効な場合のみ、ウォッチドッグの閾値を超えるとスキャンがアボートされます。

0 : アナログウォッチドッグ割り込みを無効化

1 : アナログウォッチドッグ割り込みを有効化

ビット 5 **EOCIE**:EOC の割り込みイネーブル

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 割り込みを無効化

1 : EOC 割り込みを有効化 EOC 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 4:0 **AWDCH[4:0]** : アナログウォッチドッグチャンネル選択ビット

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。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 ADC_IN0

00001 : ADC アナログ入力 ADC_IN1

...

11000 : ADC アナログ入力 ADC_IN24

11001 : ADC アナログ入力 ADC_IN25

11010 : ADC アナログ入力 ADC_IN26

その他の値は予約済みです。

注 : **ADC アナログ入力 ADC_IN16、ADC_IN17、ADC_IN26 は、内部でそれぞれ温度センサ、 V_{REFINT} 、 V_{COMP} に接続されています。**

12.15.3 ADC 制御レジスタ 2 (ADC_CR2)

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
予約済 み	SWSTA RT	EXTEN		EXTSEL[3:0]				予約済 み	JSWSTA RT	JEXTEN		JEXTSEL[3:0]			
	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
予約済 み				ALIGN	EOCS	DDS	DMA	予約済 み	DELS			予約済 み	ADC_CF G	CONT	ADON
				rW	rW	rW	rW		rW	rW	rW		rW	rW	

ビット 31 予約済みであり、リセット値に保持する必要があります。

ビット 30 **SWSTART** : レギュラチャネルの変換開始

このビットは、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、変換開始直後にハードウェアによってクリアされます。

0 : リセット状態

1 : レギュラチャネルの変換開始

注 : このビットは、ADONS=1 かつ RCNR=0 の場合にのみセットする必要があります。

ビット 29:28 **EXTEN** : レギュラチャネルの外部トリガイネーブル

これらのビットは、外部トリガ極性を選択しレギュラグループの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 : トリガー検出を無効化

01 : 立ち上がりエッジでトリガー検出

10 : 立ち下がりエッジでトリガー検出

11 : 立ち上がりと立ち下がりの両エッジでトリガー検出

注 : ADONS=1 の場合にのみ、外部 トリガを有効化する必要があります。

ビット 27:24 **EXTSEL[3:0]**:レギュラグループの外部イベント選択

これらのビットは、レギュラグループの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。

0000 : TIM9_CC2 イベント

0001 : TIM9_TRGO イベント

0010 : TIM2_CC3 イベント

0011 : TIM2_CC2 イベント

0100 : TIM3_TRGO イベント

0101 : TIM4_CC4 イベント

0110 : TIM2_TRGO イベント

0111 : TIM3_CC1 イベント

1000 : TIM3_CC3 イベント

1001 : TIM4_TRGO イベント

1010 : TIM6_TRGO イベント

1011 : 予約済み

1100 : 予約済み

1101 : 予約済み

1110 : 予約済み

1111 : EXTI ライン 11

ビット 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 **JSWSTART** : インジェクトチャネルの変換開始

このビットは、ソフトウェアによってセットされ、変換開始直後にハードウェアによってクリアされます。

0 : リセット状態

1 : インジェクトチャネルの変換開始

注 : このビットは、**ADONS=1** かつ **JCNR=0** の場合にのみセットする必要があります。

ビット 21:20 **JEXTEN**:インジェクトチャネルの外部トリガイネーブル

これらのビットは、外部トリガ極性を選択しインジェクトグループの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 : トリガー検出を無効化

01 : 立ち上がりエッジでトリガー検出

10 : 立ち下がりエッジでトリガー検出

11 : 立ち上がりと立ち下がりの両エッジでトリガー検出

注 : **ADONS=1** の場合にのみ、外部トリガを有効化する必要があります。

ビット 19:16 **JEXTSEL[3:0]**:インジェクトグループの外部イベント選択

これらのビットは、インジェクトグループの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。

0000 : TIM9_CC1 イベント

0001 : TIM9_TRGO イベント

0010 : TIM2_TRGO イベント

0011 : TIM2_CC1 イベント

0100 : TIM3_CC4 イベント

0101 : TIM4_TRGO イベント

0110 : TIM4_CC1 イベント

0111 : TIM4_CC2 イベント

1000 : TIM4_CC3 イベント

1001 : TIM10_CC1 イベント

1010 : TIM7_TRGO イベント

1011 : 予約済み

1100 : 予約済み

1101 : 予約済み

1110 : 予約済み

1111 : EXTI ライン 15

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 **ALIGN** : データアライメント

このビットは、ソフトウェアによってセット/クリアします。図 44 および 図 45 を参照してください。

0 : 右詰め

1 : 左詰め

ビット 10 **EOCS**:変換選択の終了

このビットは、ソフトウェアによってセット/クリアします。

0 : レギュラ変換の各シーケンスの終わりで EOC ビットがセットされます。

1 : 各レギュラ変換の終わりで EOC ビットがセットされます。

ビット 9 **DDS** : DMA ディスエーブル選択

このビットは、ソフトウェアによってセット/クリアします。

0 : 最後の転送後新たな DMA リクエストは発行されません (DMA コントローラの設定どおり)

1 : データが変換され DMA=1 である限り DMA リクエストは発行されます。

ビット 8 **DMA** : ダイレクトメモリアクセスモード

このビットは、ソフトウェアによってセット/クリアします。詳細については、DMA コントローラの章を参照してください。

0 : DMA モードを無効化

1 : DMA モードを有効化

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 **DELS** : 遅延選択

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。これらのビットは、変換または変換のシーケンスの後に使用される遅延長を指定します。

000 : 遅延なし

001 : 変換したデータが読み出されるまで (DR 読み出しまたはレギュラ変換では EOC=0、インジェクト変換では JEOP=0)

010 : 変換終了後 7 APB クロックサイクル

011 : 変換終了後 15 APB クロックサイクル

100 : 変換終了後 31 APB クロックサイクル

101 : 変換終了後 63 APB クロックサイクル

110 : 変換終了後 127 APB クロックサイクル

111 : 変換終了後 255 APB クロックサイクル

注 : 1- このビットは、 $ADON=0$ のときにのみ書き込む必要があります。

2- クロックドメインを跨ぐため、2 または 3 ADC クロックサイクルの遅延が新しい変換の開始前に加わります。

3- APB クロックと ADC クロックとの間の与えられた周波数比に対して必要とされる遅延は、AHB バスと APB バスの動作に依存します。ADC がデータ転送を必要とする唯一のペリフェラルである場合、次の最小遅延を設定する必要があります。

$f_{APB} < f_{ADCCLK}/2$ の場合は 15 APB クロックサイクル、 $f_{APB} < f_{ADCCLK}$ の場合は 7 APB クロックサイクル、その他の場合は遅延不要。

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 **ADC_CFG** : ADC 設定

このビットは、ソフトウェアによってセット/クリアします。変換するチャンネルバンクを選択します。

0 : チャンネル ADC_IN0 から 31 に対してバンク A を選択

1 : チャンネル ADC_IN0 から 31b に対してバンク B を選択

注 : このビットは、変換を行っていないときにのみ変更する必要があります。

このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 1 **CONT** : 連続変換

このビットは、ソフトウェアによってセット/クリアします。セットされると、変換が連続して実行され、クリアされるまで続きます。

0 : シングル変換モード

1 : 連続変換モード

ビット 0 **ADON** : A/D コンバータオン/オフ

このビットは、ソフトウェアによってセット/クリアします。

0 : ADC 変換を無効化し、パワーダウンモードになります

1 : ADC イネーブル : 開始イベント (ハードウェアまたはソフトウェア) を受信すると直ちに変換を開始できます。変換がない場合、ADC は PDI ビットと PDD ビットに応じてパワーアップまたはパワーダウンモードに移行します。

注 : このビットは、 $ADONS=0$ かつ $ADONS=1$ の場合にのみセットする必要があります。

12.15.4 ADC サンプルタイムレジスタ 1 (ADC_SMPR1)

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
予約済み	SMP29[2:0]			SMP28[2:0]			SMP27[2:0]			SMP26[2:0]			SMP25[2:1]		
	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	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP25[0]	SMP24[2:0]			SMP23[2:0]			SMP22[2:0]			SMP21[2:0]			SMP20[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	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 29:0 **SMPx[2:0]** : チャネル x のサンプリングタイム選択

これらのビットは、各チャネルのサンプルタイムを個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

- 000 : 4 サイクル
- 001 : 9 サイクル
- 010 : 16 サイクル
- 011 : 24 サイクル
- 100 : 48 サイクル
- 101 : 96 サイクル
- 110 : 192 サイクル
- 111 : 384 サイクル

注 : これらのビットは、ADON=0 のときにのみ書き込む必要があります。

12.15.5 ADC サンプルタイムレジスタ 2 (ADC_SMPR2)

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
予約済み		SMP19[2:0]			SMP18[2:0]			SMP17[2:0]			SMP16[2:0]			SMP15[2:1]	
		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
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMP15[0]	SMP14[2:0]			SMP13[2:0]			SMP12[2:0]			SMP11[2:0]			SMP10[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	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 29:0 **SMPx[2:0]** : チャネル x のサンプリングタイム選択

これらのビットは、各チャネルのサンプルタイムを個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000 : 4 サイクル

001 : 9 サイクル

010 : 16 サイクル

011 : 24 サイクル

100 : 48 サイクル

101 : 96 サイクル

110 : 192 サイクル

111 : 384 サイクル

注 : これらのビットは、ADON=0 のときにのみ書き込む必要があります。

12.15.6 ADC サンプルタイムレジスタ 3 (ADC_SMPR3)

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
予約済み	SMP9[2:0]			SMP8[2:0]			SMP7[2:0]			SMP6[2:0]			SMP5[2:1]		
	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
SMP5[0]	SMP4[2:0]			SMP3[2:0]			SMP2[2:0]			SMP1[2:0]			SMP0[2:0]		
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 29:0 **SMPx[2:0]** : チャネル *x* のサンプルタイム選択

これらのビットは、各チャネルのサンプルタイムを個別に選択するために、ソフトウェアによって書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

000 : 4 サイクル

001 : 9 サイクル

010 : 16 サイクル

011 : 24 サイクル

100 : 48 サイクル

101 : 96 サイクル

110 : 192 サイクル

111 : 384 サイクル

注 : これらのビットは、ADON=0 のときにのみ書き込む必要があります。

12.15.7 ADC インジェクトチャネルデータオフセットレジスタ x (ADC_JOFRx) (x=1..4)

アドレスオフセット : 0x18 から 0x24

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				JOFFSETx[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 JOFFSETx[11:0] : インジェクトチャネル x のデータオフセット

これらのビットは、インジェクトチャネルを変換するときに変換データから引かれるオフセットを定義するために、ソフトウェアによって書き込まれます。変換結果は、ADC_JDRx レジスタで読み出すことができます。

12.15.8 ADC ウォッチドッグ高閾値レジスタ (ADC_HTR)

アドレスオフセット : 0x28

リセット値 : 0x0000 0FFF

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				HT[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 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。

12.15.9 ADC ウォッチドッグ低閾値レジスタ (ADC_LTR)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				LT[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 **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。

12.15.10 ADC レギュラシーケンスレジスタ 1 (ADC_SQR1)

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
予約済み								L[4:0]				SQ28[4:1]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SQ28[0]		SQ27[4:0]					SQ26[4:0]				SQ25[4:0]				
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 24:20 **L[4:0]** : レギュラチャンネルシーケンス長

これらのビットは、レギュラチャンネル変換シーケンスの合計変換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。

00000 : 1 変換

00001 : 2 変換

...

11010 : 27 変換

11011 : 28 変換 (Cat.3、Cat.4、Cat.5 デバイスのみに適用)

ビット 19:15 **SQ28[4:0]** : レギュラシーケンス内の 28 番目の変換

これらのビットには、変換シーケンス内の 28 番目として割り当てられたチャンネル番号 (0..31) がソフトウェアによって書き込まれます。ADC_CR2 レジスタの ADC_CFG ビットに応じてバンク A またはバンク B でチャンネルが選択されます。

注: このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 14:10 **SQ27[4:0]** : レギュラシーケンス内の 27 番目の変換

Cat.1 Cat.2 デバイス: これらのビットには、変換シーケンス内の 27 番目として割り当てられたチャンネル番号 (0..26) がソフトウェアによって書き込まれます。

Cat.3、Cat.4、Cat.5 デバイス: レギュラシーケンス内の 27 番目の変換

ビット 9:5 **SQ26[4:0]** : レギュラシーケンス内の 26 番目の変換

ビット 4:0 **SQ25[4:0]** : レギュラシーケンス内の 25 番目の変換

12.15.11 ADC レギュラシーケンスレジスタ 2 (ADC_SQR2)

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
予約済み		SQ24[4:0]					SQ23[4:0]					SQ22[4:1]			
		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
SQ22[0]	SQ21[4:0]					SQ20[4:0]					SQ19[4:0]				
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 29:26 **SQ24[4:0]** : レギュラシーケンス内の 24 番目の変換
これらのビットには、変換シーケンス内の 24 番目として割り当てられたチャンネル番号 (0.31) がソフトウェアによって書き込まれます。
- ビット 24:20 **SQ23[4:0]** : レギュラシーケンス内の 23 番目の変換
- ビット 19:15 **SQ22[4:0]** : レギュラシーケンス内の 22 番目の変換
- ビット 14:10 **SQ21[4:0]** : レギュラシーケンス内の 21 番目の変換
- ビット 9:5 **SQ20[4:0]** : レギュラシーケンス内の 20 番目の変換
- ビット 4:0 **SQ19[4:0]** : レギュラシーケンス内の 19 番目の変換

12.15.12 ADC レギュラシーケンスレジスタ 3 (ADC_SQR3)

アドレスオフセット : 0x38

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		SQ18[4:0]					SQ17[4:0]					SQ16[4:1]			
		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
SQ16[0]	SQ15[4:0]					SQ14[4:0]					SQ13[4:0]				
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 29:25 **SQ18[4:0]** : レギュラシーケンス内の 18 番目の変換
これらのビットには、シーケンス内の 18 番目の変換として割り当てられたチャンネル番号 (0.31) がソフトウェアによって書き込まれます。
- ビット 24:20 **SQ17[4:0]** : レギュラシーケンス内の 17 番目の変換
- ビット 19:15 **SQ16[4:0]** : レギュラシーケンス内の 16 番目の変換
- ビット 14:10 **SQ15[4:0]** : レギュラシーケンス内の 15 番目の変換
- ビット 9:5 **SQ14[4:0]** : レギュラシーケンス内の 14 番目の変換
- ビット 4:0 **SQ13[4:0]** : レギュラシーケンス内の 13 番目の変換

12.15.13 ADC レギュラシーケンスレジスタ 4 (ADC_SQR4)

アドレスオフセット : 0x3C

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	SQ12[4:0]					SQ11[4:0]					SQ10[4:1]				
	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
SQ10[0]	SQ9[4:0]					SQ8[4:0]					SQ7[4:0]				
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 29:26 **SQ12[4:0]** : レギュラシーケンス内の 12 番目の変換

これらのビットには、シーケンス内の 12 番目の変換として割り当てられたチャネル番号 (0..31) がソフトウェアによって書き込まれます。

ビット 24:20 **SQ11[4:0]** : レギュラシーケンス内の 11 番目の変換ビット 19:15 **SQ10[4:0]** : レギュラシーケンス内の 10 番目の変換ビット 14:10 **SQ9[4:0]** : レギュラシーケンス内の 9 番目の変換ビット 9:5 **SQ8[4:0]** : レギュラシーケンス内の 8 番目の変換ビット 4:0 **SQ7[4:0]** : レギュラシーケンス内の 7 番目の変換

12.15.14 ADC レギュラシーケンスレジスタ 5 (ADC_SQR5)

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
予約済み	SQ6[4:0]					SQ5[4:0]					SQ4[4:1]				
	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
SQ4_0	SQ3[4:0]					SQ2[4:0]					SQ1[4:0]				
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 29:25 **SQ6[4:0]** : レギュラシーケンス内の 6 番目の変換

これらのビットには、変換シーケンスで変換される 6 番目として割り当てられたチャネル番号 (0..31) がソフトウェアによって書き込まれます。

ビット 24:20 **SQ5[4:0]** : レギュラシーケンス内の 5 番目の変換ビット 19:15 **SQ4[4:0]** : レギュラシーケンス内の 4 番目の変換ビット 14:10 **SQ3[4:0]** : レギュラシーケンス内の 3 番目の変換ビット 9:5 **SQ2[4:0]** : レギュラシーケンス内の 2 番目の変換ビット 4:0 **SQ1[4:0]** : レギュラシーケンス内の 1 番目の変換

12.15.15 ADC インジェクトシーケンスレジスタ (ADC_JSQR)

アドレスオフセット : 0x44

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										JL[1:0]		JSQ4[4:1]			
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ4[0]		JSQ3[4:0]				JSQ2[4:0]					JSQ1[4:0]				
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:22 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 21:20 JL[1:0] : インジェクトシーケンス長

これらのビットは、インジェクトチャネル変換シーケンスの合計変換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。

- 00 : 1 変換
- 01 : 2 変換
- 10 : 3 変換
- 11 : 4 変換

ビット 19:15 JSQ4[4:0]:インジェクトシーケンス内の 4 番目の変換 (JL[1:0]=3 の場合は、下の注を参照)

これらのビットには、変換シーケンス内の 4 番目として割り当てられたチャネル番号 (0..31) がソフトウェアによって書き込まれます。ADC_CR2 レジスタの ADC_CFG ビットに応じてバンク A またはバンク B でチャネルが選択されます。

ビット 14:10 JSQ3[4:0]: インジェクトシーケンス内の 3 番目の変換 (JL[1:0]=3 の場合は、下の注を参照)

ビット 9:5 JSQ2[4:0]:インジェクトシーケンス内の 2 番目の変換 (JL[1:0]=3 の場合は、下の注を参照)

ビット 4:0 JSQ1[4:0]:インジェクトシーケンス内の 1 番目の変換 (JL[1:0]=3 の場合は、下の注を参照)

注 :

JL[1:0]=3 (シーケンサ内に 4 つのインジェクト変換) の場合、ADC は JSQ1[4:0]、JSQ2[4:0]、JSQ3[4:0]、JSQ4[4:0] の順でチャネルを変換します。

JL=2 (シーケンサ内に 3 つのインジェクト変換) の場合、ADC は JSQ2[4:0]、JSQ3[4:0]、JSQ4[4:0] の順でチャネルを変換します。

JL=1 (シーケンサ内に 2 つのインジェクト変換) の場合、ADC は JSQ3[4:0]、JSQ4[4:0] の順でチャネルを変換します。

JL=0 (シーケンサ内に 1 つのインジェクト変換) の場合、ADC は JSQ4[4:0] チャネルだけを変換します。



12.15.16 ADC インジェクトデータレジスタ x (ADC_JDRx) (x= 1..4)

アドレスオフセット : 0x48 から 0x54

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
JDATA[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 **JDATA[15:0]** : インジェクトデータ

これらのビットは読み出し専用です。インジェクトチャンネル x からの変換結果を含みます。データは、[図 44](#)と [図 45](#)に示すように、左詰めまたは右詰めで格納されます。

12.15.17 ADC レギュラデータレジスタ (ADC_DR)

アドレスオフセット : 0x58

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
データ [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 **DATA[15:0]** : レギュラデータ

これらのビットは読み出し専用です。レギュラチャンネルからの変換結果を含みます。データは、[図 44](#)と [図 45](#)に示すように、左詰めまたは右詰めで格納されます。

12.15.18 ADC サンプルタイムレジスタ 0 (ADC_SMPR0)

アドレスオフセット : 0x5C

リセット値 : 0x0000 0000

注 : このレジスタは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

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										SMP31[2:0]			SMP30[2:0]		
										rw	rw	rw	rw	rw	rw

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 5:0 **SMPx[2:0]** : チャネル x のサンプルタイム選択

これらのビットは、各チャネルのサンプルタイムを個別に選択するために、ソフトウェアから書き込まれます。サンプルサイクル中は、チャネル選択ビットを変更することはできません。

- 000 : 4 サイクル
- 001 : 9 サイクル
- 010 : 16 サイクル
- 011 : 24 サイクル
- 100 : 48 サイクル
- 101 : 96 サイクル
- 110 : 192 サイクル
- 111 : 384 サイクル

注 : これらのビットは、ADON=0 のときにのみ書き込む必要があります。



12.15.19 ADC 共通 ステータスレジスタ (ADC_CSR)

アドレスオフセット : 0x00 (このオフセットアドレスは、ADC 共通 レジスタ 0x300 のベースアドレスを基準とします)

リセット値 : 0x0000 0000

このレジスタは、様々な ADC のステータスビットのイメージを提供します。ただし、これは読み出し専用で、別の ADC のステータスビットをクリアすることはできません。その代わりに、対応する ADC_SR レジスタに 0 を書き込むことによって各ステータスビットをクリア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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み									ADONS1	OVR1	STRT1	JSTRT1	JEOC 1	EOC1	AWD1
									r	r	r	r	r	r	r

ビット 31:7 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 **ADONS1** : ADC1 の ADON ステータス
このビットは、ADC_SR レジスタの ADONS ビットのコピーです。
- ビット 5 **OVR1** : ADC のオーバーランフラグ
このビットは、ADC_SR レジスタの OVR ビットのコピーです。
- ビット 4 **STRT1** : ADC のレギュラチャネル開始フラグ
このビットは、ADC_SR レジスタの STRT ビットのコピーです。
- ビット 3 **JSTRT1** : ADC のインジェクトチャネル開始フラグ
このビットは、ADC_SR レジスタの JSTRT ビットのコピーです。
- ビット 2 **JEOC1** : ADC のインジェクトチャネル変換終了
このビットは、ADC_SR レジスタの JEOC ビットのコピーです。
- ビット 1 **JEOC1** : ADC の変換終了
このビットは、ADC_SR レジスタの EOC ビットのコピーです。
- ビット 0 **AWD1** : ADC のアナログウォッチドッグフラグ
このビットは、ADC_SR レジスタの AWD ビットのコピーです。



12.15.20 ADC 共通制御レジスタ (ADC_CCR)

アドレスオフセット : 0x04 (このオフセットアドレスは、ADC 共通 レジスタ 0x300 のベースアドレスを基準とします)

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								TSVREFE	予約済み						ADCPRE[1:0]	
								rw							rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
予約済み																

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 **TSVREFE** : 温度センサおよびV_{REFINT} イネーブル

このビットは、温度センサおよび V_{REFINT} チャネルを有効化/無効化にするために、ソフトウェアによってセット/クリアされます。

0 : 温度センサおよび V_{REFINT} チャネルを無効化

1 : 温度センサおよび V_{REFINT} チャネルを有効化

ビット 22:18 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 17:16 **ADCPRE** : ADC プリスケアラ

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。

00 : HSI を 1 分周

01 : HSI を 2 分周

10 : HSI を 4 分周

11 : 予約済み

ビット 15:0 予約済みであり、リセット値に保持する必要があります。

12.15.21 ADC レジスタマップ

次の表は ADC レジスタの一覧です。

表 63. ADC グローバルレジスタマップ

オフセット	レジスタ
0x000 から 0x058	ADC
0x05C から 0x2FC	予約済み
0x300 から 0x304	一般レジスタ

表 64. ADC のレジスタマップおよびリセット値

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_SR	予約済み																						JCNR	RCNR	予約済み	ADONS	OVR	STRT	JSTRT	JEOC	EOC	AWD																
	リセット値																							0	0		0	0	0	0	0	0	0																
0x04	ADC_CR1	予約済み					OVRIE	RES[1:0]		AWDEN	JAWDEN	予約済み			PDI	PDD	DISC NUM [2:0]		JDISCEN	DISCEN	JAUTO	AWD SGL	SCAN	JEOCIE	AWDIE	EOCIE	AWDCH[4:0]																						
	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																	
0x08	ADC_CR2	予約済み	SWSTART	EXTEN[1:0]		EXTSEL[3:0]				予約済み	JSWSTART	JEXTEN[1:0]		JEXTSEL [3:0]			予約済み					ALIGN	EOCS	DDS	DMA	予約済み	DELS[2:0]			予約済み	ADC_CFG		CONT	ADON															
	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															
0x0C	ADC_SMPR1	サンプル時間ビット SMPx_x																																															
	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															
0x10	ADC_SMPR2	サンプル時間ビット SMPx_x																																															
	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															
0x14	ADC_SMPR3	サンプル時間ビット SMPx_x																																															
	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															
0x18	ADC_JOFR1	予約済み																				JOFFSET1[11:0]																											
	リセット値																					0	0	0	0	0	0	0	0	0	0	0																	
0x1C	ADC_JOFR2	予約済み																				JOFFSET2[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	0	
0x20	ADC_JOFR3	予約済み																				JOFFSET3[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	0	0
0x24	ADC_JOFR4	予約済み																				JOFFSET4[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	0	0



表 64. ADC のレジスタマップおよびリセット値 (続き)

[illegible]

表 65. ADC のレジスタマップおよびリセット値 (共通レジスタ)

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_CSR	予約済み																									ADONS	OVR	STRT	JSTRT	JEOC	EOC	AWD	
	リセット値																										0	0	0	0	0	0	0	
																												ADC1						
0x04	ADC_CCR	予約済み									TSVREFE	予約済み					ADCPRE		予約済み															
	リセット値										0						0	0																

レジスタ境界アドレステーブルについては、表 5 (47 ページ) 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。

13 D/A コンバータ (DAC)

13.1 DAC の概要

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、データを左詰めまたは右詰めに設定できます。DAC には 2 つの出力チャンネルがあり、それぞれがコンバータを内蔵しています。デュアル DAC チャンネルモードでは、変換は独立して行われるか、両方のチャンネルが同期更新操作のためにグループ化されているときには同時に行うことができます。分解能を高めるため、入力基準ピン V_{REF+} (ADC と共用) を使用することができます。

13.2 DAC の主な機能

- 2 つの DAC コンバータ：それぞれ 1 つの出力チャンネル
- 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
- デュアル DAC チャンネルの独立または同時変換
- 各チャンネルでの DMA 利用
- DMA アンダーランエラー検出
- 変換外部トリガ
- 入力基準電圧 V_{REF+}

DAC チャンネルのブロック図を [図 55](#) に、ピンの概要を [表 66](#) に示します。

図 55. DAC チャンネルブロック図

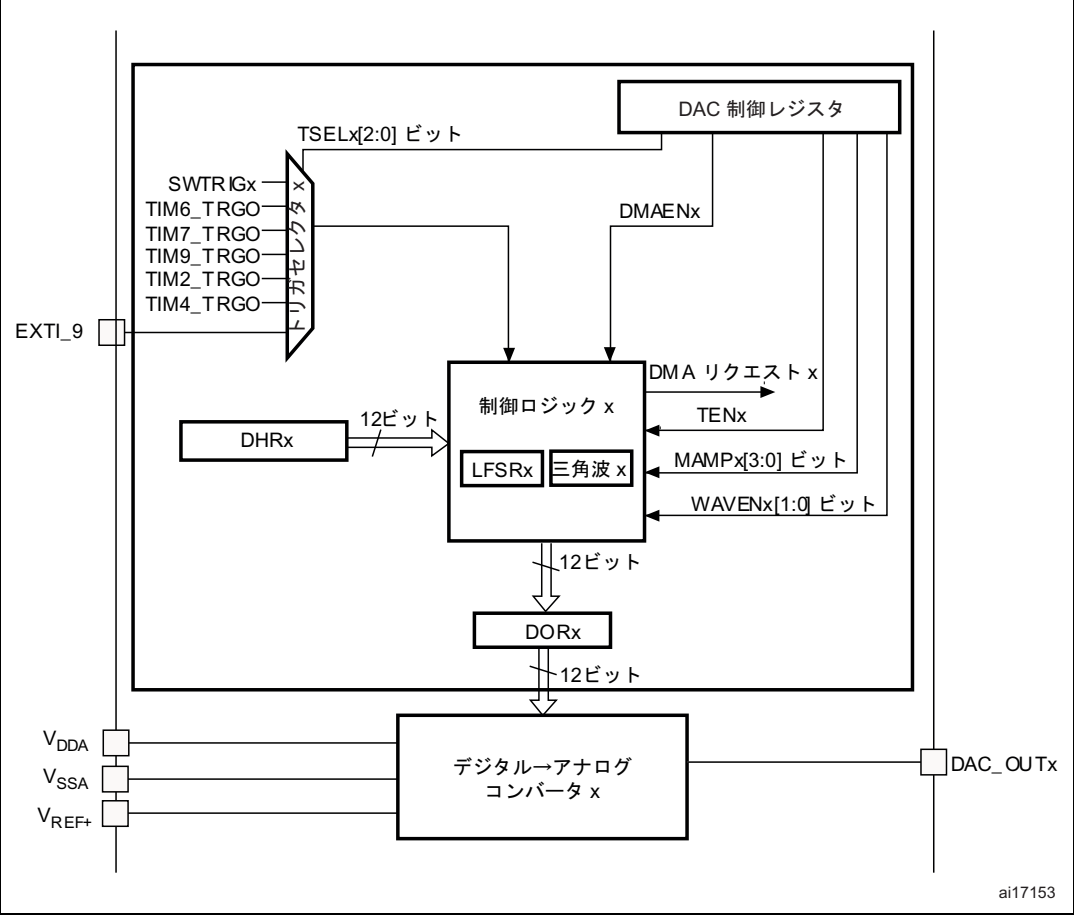


表 66. DAC のピン

名称	信号種別	摘要
VREF+	入力、アナログ基準電圧 正	DAC のハイレベル／正基準電圧、 1.8 V ≤ VREF+ ≤ VDDA
VDDA	入力、アナログ電源	アナログ電源
VSSA	入力、アナログ電源グラウンド	アナログ電源供給のグラウンド
DAC_OUTx	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 または PA5) が自動的にアナログコンバータ出力 (DAC_OUTx) に接続されます。寄生消費を防ぐために、PA4 または PA5 ピンはまずアナログ (AIN) として設定してください。

13.3 DAC 機能の説明

13.3.1 DAC チャンネルイネーブル

各 DAC チャンネルは、DAC_CR レジスタの対応する ENx ビットをセットすることによって起動できます。DAC チャンネルはスタートアップ時間 t_{WAKEUP} 経過後に有効になります。

注： ENx ビットは、アナログ DAC チャンネルマクロセルのみを有効にします。DAC チャンネル 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。

13.3.2 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駆動できる 2 つの出力バッファを内蔵しています。各 DAC チャンネル出力バッファは、DAC_CR レジスタの対応する BOFFx ビットを使用して有効/無効にできます。

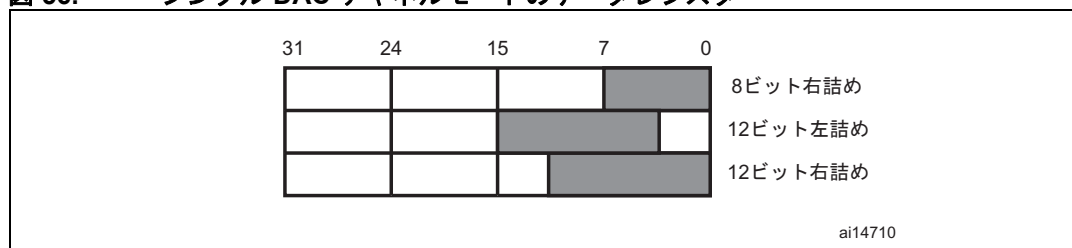
13.3.3 DAC データフォーマット

次のように、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。

- 1 つの DAC チャンネル x では、3 つの設定が可能です。
 - 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する必要があります。
 - 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する必要があります。
 - 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する必要があります。

ロードされた DAC_DHRyyyx レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHRx (Data Holding Register x、メモリマップされない内部レジスタ) 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。

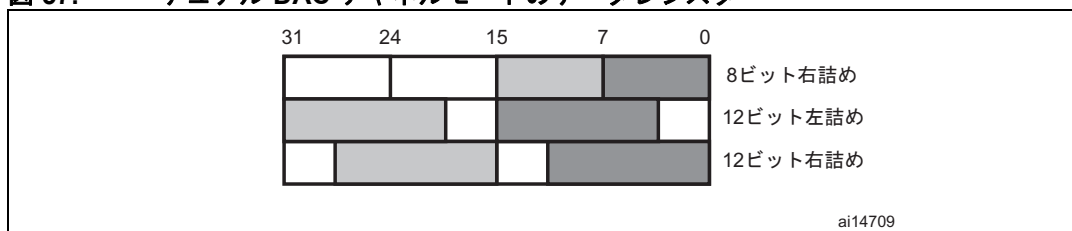
図 56. シングル DAC チャンネルモードのデータレジスタ



- デュアル DAC チャンネルでは、3 つの設定が可能です。
 - － 8 ビット右詰め：DAC チャンネル 1 のデータは、DAC_DHR8RD[7:0] ビット (DHR1[11:4] ビットに格納) に、DAC チャンネル 2 のデータは、DAC_DHR8RD[15:8] ビット (DHR2[11:4] ビットに格納) にロードされます。
 - － 12 ビット左詰め：DAC チャンネル 1 のデータは、DAC_DHR12LD[15:4] ビット (DHR1[11:0] ビットに格納) に、DAC チャンネル 2 のデータは、DAC_DHR12LD[31:20] ビット (DHR2[11:0] ビットに格納) にロードされます。
 - － 12 ビット右詰め：DAC チャンネル 1 のデータは、DAC_DHR12RD[11:0] ビット (DHR1[11:0] ビットに格納) に、DAC チャンネル 2 のデータは、DAC_DHR12LD[27:16] ビット (DHR2[11:0] ビットに格納) にロードされます。

ロードされた DAC_DHRyyD レジスタに応じて、ユーザによって書き込まれたデータは、シフトされてから、DHR1 および DHR2 (Data Holding Register、メモリマップされない内部レジスタ) に格納されます。その後、DHR1 および DHR2 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、それぞれ DOR1 および DOR2 レジスタにロードされます。

図 57. デュアル DAC チャンネルモードのデータレジスタ



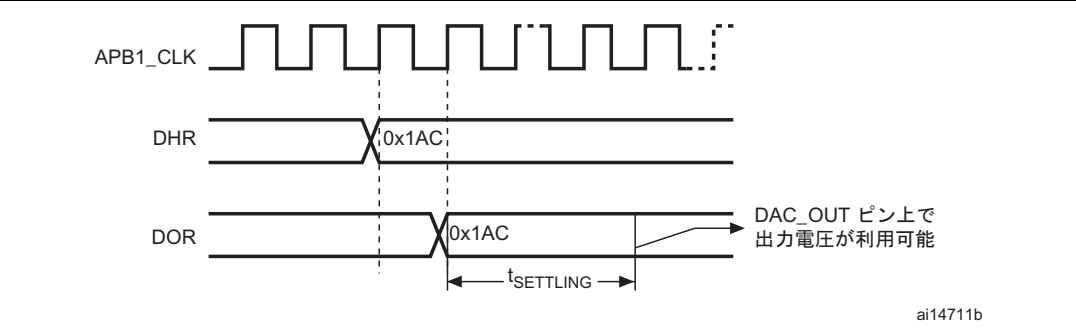
13.3.4 DAC 変換

DAC_DORx に直接書き込むことはできず、DAC_DHRx レジスタをロードすることによって (DAC_DHR8Rx、DAC_DHR12Lx、DAC_DHR12Rx、DAC_DHR8RD、DAC_DHR12LD、または DAC_DHR12LD への書き込み)、DAC チャンネル 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 TENx 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 APB1 クロックサイクル後に行われます。

DAC_DORx に DAC_DHRx 値がロードされると、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 58. トリガ無効 (TEN = 0) 時の変換タイミング図



13.3.5 DAC 出力電圧

デジタル入力は、0 から V_{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、次式で求められます。

$$DAC_{output} = V_{REF} \times \frac{DOR}{4095}$$

13.3.6 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 67 に示すように、TSELx[2:0] 制御ビットは、8 つのイベントから変換をトリガするイベントを決定します。

表 67. 外部トリガ

ソース	タイプ	TSEL[2:0]
タイマ 6 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
タイマ 7 TRGO イベント		010
タイマ 9 TRGO イベント		011
タイマ 2 TRGO イベント		100
タイマ 4 TRGO イベント		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。



13.3.7 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。

デュアルモードでは、両方の DMAENx ビットがセットされている場合、2 つの DMA リクエストが生成されます。1 つの DMA リクエストしか必要ない場合には、対応する DMAENx ビットだけをセットしてください。これにより、アプリケーションは 1 つの DMA リクエストと一意な DMA チャンネルを使用して、両方の DAC チャンネルをデュアルモードで管理することができます。

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 つ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条件を報告します。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。

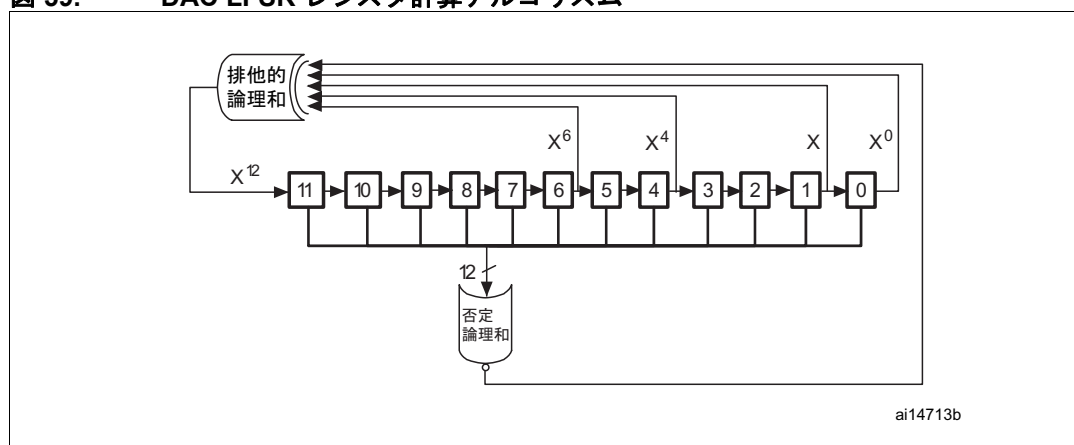
ソフトウェアでは、「1」を書き込むことによって DMAUDRx フラグをクリアし、使用された DMA ストリームの DMAEN ビットをクリアし、DMA と DAC のチャンネルx を再初期化して転送を正しくリスタートさせてください。また、DAC トリガ変換周波数を変更するか DMA の負荷を軽減して、新しい DMA アンダーランを回避してください。最後に、DMA データ転送と変換トリガを有効にすることによって DAC 変換を再開することができます。

各 DAC チャンネル x 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。

13.3.8 ノイズ発生

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 ノイズ生成は、WAVEx[1:0] を「01」にセットすることによって選択されます。LFSR にプリロードされる値は 0xAAA です。このレジスタは、各トリガイベントの 3 APB1 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 59. DAC LFSR レジスタ計算アルゴリズム

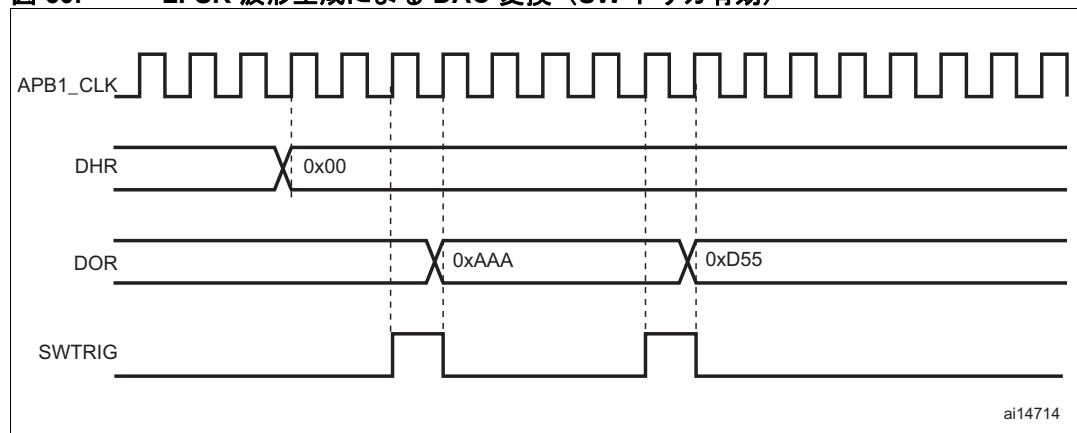


LFSR 値は、DAC_CR レジスタの MAMPx[3:0] ビットによって部分的または全体的にマスクでき、オーバーフローなしに DAC_DHRx の内容に加算されて、DAC_DORx レジスタに格納されます。

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 60. 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。

13.3.9 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 三角波生成は、WAVEx[1:0] を「10」にセットすることによって選択されます。振幅は、DAC_CR レジスタの MAMPx[3:0] ビットによって設定されます。内部三角波カウンタは、各トリガイベントの 3 APB1 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より小さい場合に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 61. DAC 三角波生成

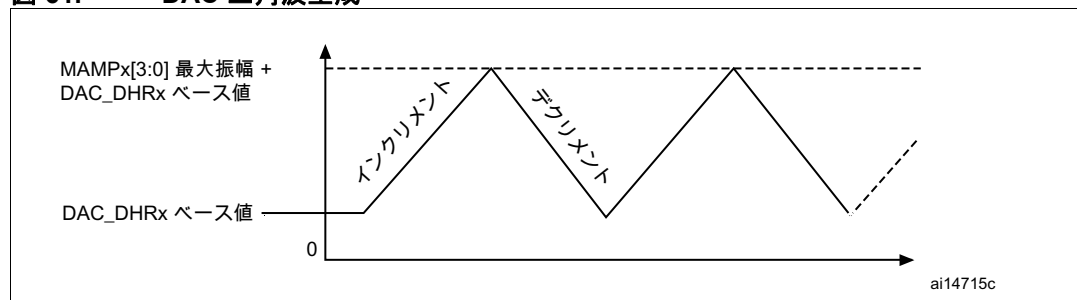
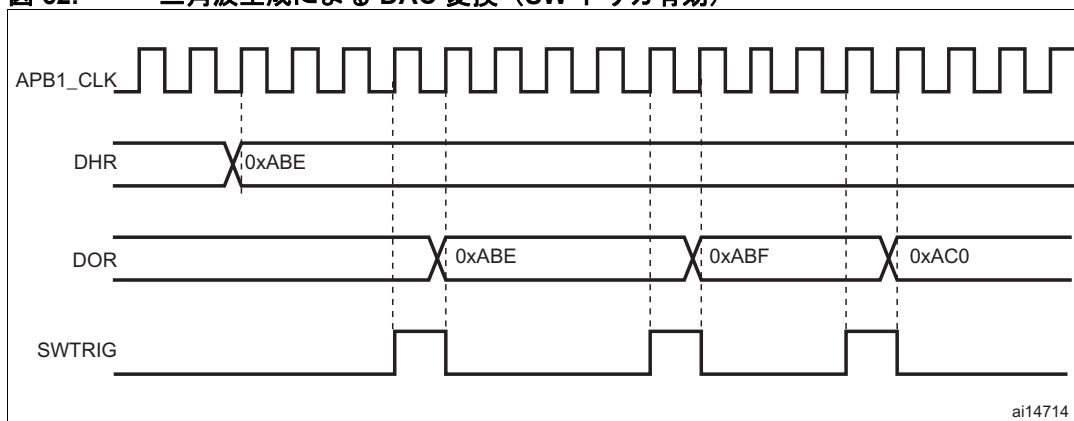


図 62. 三角波生成による DAC 変換 (SW トリガ有効)



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。

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。それ以外の場合、これらのビットは変更できません。

13.4 デュアル DAC チャンネル変換

同時に 2 つの DAC チャンネルを必要とするアプリケーションで、バスのバンド幅を有効に使用するために、DHR8RD、DHR12RD、および DHR12LD の 3 つのデュアルレジスタが内蔵されています。両方の DAC チャンネルを同時に駆動するには、一意なレジスタアクセスが必要です。

2 つの DAC チャンネルとこれらのデュアルレジスタを使用することで、11 の変換モードが使用可能です。すべての変換モードが、必要に応じて、個別の DHRx レジスタを使用して利用することもできます。

すべてのモードについて、以下の節で説明します。

13.4.1 波形生成なしの独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

DAC チャンネル 1 トリガが発生すると、DHR1 レジスタが DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。

DAC チャンネル 2 トリガが発生すると、DHR2 レジスタが DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。

13.4.2 1 つの LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの 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。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) に、デュアル DAC チャンネルデータをロードします。

DAC チャンネル 1 トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR1 カウンタが更新されます。

DAC チャンネル 2 トリガが発生すると、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR2 カウンタが更新されます。

13.4.3 異なる LFSR 生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャンネル WAVEx[1:0] ビットを "01" に設定し、MAMP1[3:0] ビットと MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

DAC チャンネル 1 トリガが発生すると、MAMP1[3:0] によって設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR1 カウンタが更新されます。

DAC チャンネル 2 トリガが発生すると、MAMP2[3:0] によって設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR2 カウンタが更新されます。

13.4.4 1 つの三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャンネルの 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。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

DAC チャンネル 1 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 1 の三角波カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 1 の三角波カウンタが更新されます。



DAC チャンネル 2 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 2 の三角波カウンタが DHR2 レジスタに加算され、和が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 2 の三角波カウンタが更新されます。

13.4.5 異なる三角波生成による独立トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに異なる値を設定することによって、異なるトリガソースを設定します。
- 2 つの DAC チャンネルの WAVEx[1:0] ビットを "1x" に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる最大振幅値を設定します。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

DAC チャンネル 1 トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャンネル 1 の三角波カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 1 の三角波カウンタが更新されます。

DAC チャンネル 2 トリガが発生すると、MAMP2[3:0] によって設定された三角波振幅を持つ DAC チャンネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 2 の三角波カウンタが更新されます。

13.4.6 同時ソフトウェア開始

DAC をこの変換モードに設定するには、次の手順が必要です。

- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) に、デュアル DAC チャンネルデータをロードします。

この設定では、1 APB1 クロックサイクル後に、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます。

13.4.7 波形生成なしの同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- 2 つの DAC チャンネルのトリガイネーブルビット TEN1 および TEN2 をセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャンネルに同じトリガソースを設定します。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) に、デュアル DAC チャンネルデータをロードします。

トリガが発生すると、DHR1 および DHR2 レジスタが DAC_DOR1 と DAC_DOR2 にそれぞれ転送されます (3 APB1 クロックサイクル後)。

13.4.8 1 つの LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャンネルに同じトリガソースを設定します。
- 2 つの 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。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) に、デュアル DAC チャンネルデータをロードします。

トリガが発生すると、同じマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR1 カウンタが更新されます。同時に、同じマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR2 カウンタが更新されます。

13.4.9 異なる LFSR 生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャンネルに同じトリガソースを設定します。
- 2 つの DAC チャンネルの WAVEx[1:0] ビットを "01" に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる LFSR マスク値を設定します。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

トリガが発生すると、MAMP1[3:0] で設定されたマスクを持つ LFSR1 カウンタが DHR1 レジスタに加算され、合計が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR1 カウンタが更新されます。

同時に、MAMP2[3:0] で設定されたマスクを持つ LFSR2 カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、LFSR2 カウンタが更新されます。

13.4.10 1 つの三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャンネルに同じトリガソースを設定します。
- 2 つの DAC チャンネルの 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。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

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 1 の三角波カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 1 の三角波カウンタが更新されます。

同時に、同じ三角波振幅を持つ DAC チャンネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 2 の三角波カウンタが更新されます。



13.4.11 異なる三角波生成による同時トリガ

DAC をこの変換モードに設定するには、次の手順が必要です。

- TEN1 と TEN2 の 2 つの DAC チャンネルトリガイネーブルビットをセットします。
- TSEL1[2:0] および TSEL2[2:0] ビットに同じ値をセットすることによって、両方の DAC チャンネルに同じトリガソースを設定します。
- 2 つの DAC チャンネルの WAVEx[1:0] ビットを "1x" に設定し、MAMP1[3:0] と MAMP2[3:0] ビットで異なる最大振幅値をセットします。
- 目的の DHR レジスタ (DAC_DHR12RD、DAC_DHR12LD、または DAC_DHR8RD) にデュアル DAC チャンネルデータをロードします。

トリガが発生すると、MAMP1[3:0] によって設定された三角波振幅を持つ DAC チャンネル 1 の三角波カウンタが DHR1 レジスタに加算され、和が DAC_DOR1 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 1 の三角波カウンタが更新されます。

同時に、MAMP2[3:0] で設定された三角波振幅を持つ DAC チャンネル 2 の三角波カウンタが DHR2 レジスタに加算され、合計が DAC_DOR2 に転送されます (3 APB1 クロックサイクル後)。その後、DAC チャンネル 2 の三角波カウンタが更新されます。

13.5 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のアクセスは、ワード (32 ビット) 単位で行う必要があります。

13.5.1 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
予約済み		DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2[1:0]		TSEL2[2:0]			TEN2	BOFF2	EN2
		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
予約済み		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 31:30 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 **DMAUDRIE2**: DAC チャンネル 2 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 2 DMA アンダーラン割り込みは無効です。

1 : DAC チャンネル 2 DMA アンダーラン割り込みは有効です。

ビット 28 **DMAEN2**: DAC チャンネル 2 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 2 DMA モードは無効です。

1 : DAC チャンネル 2 DMA モードは有効です。

ビット 27:24 **MAMP2[3:0]**: DAC チャンネル 2 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 = 4095 のアンマスクビット [11:0]

ビット 23:22 **WAVE2[1:0]**: DAC チャンネル 2 ノイズ/三角波発生イネーブル

これらのビットは、ソフトウェアによってセット/リセットされます。

00 : 波形発生は無効です。

01 : ノイズ波発生は有効です。

1x : 三角波発生は有効です。

注: ビット **TEN2=1** (DAC チャンネル 2 トリガ有効) の場合のみ使用されます。



ビット 21:19 **TSEL2[2:0]** : DAC チャンネル 2 トリガ選択

これらのビットは、DAC チャンネル 2 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 : タイマ 7 TRGO イベント

011 : タイマ 9 TRGO イベント

100 : タイマ 2 TRGO イベント

101 : タイマ 4 TRGO イベント

110 : 外部ライン 9

111 : ソフトウェアトリガ

注 : ビット **TEN2=1 (DAC チャンネル2 トリガ有効)** の場合のみ使用されます。

ビット 18 **TEN2** : DAC チャンネル 2 トリガイネーブル

このビットは、DAC チャンネル 2 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 2 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_DOR2 レジスタに転送されます。

1 : DAC チャンネル 2 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_DOR2 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は、DAC_DHRx から DAC_DOR2 レジスタへの転送には、1 APB1 クロックサイクルしか必要ありません。

ビット 17 **BOFF2** : DAC チャンネル 2 出力バッファディセーブル

このビットは、DAC チャンネル 2 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 2 出力バッファは有効です。

1 : DAC チャンネル 2 出力バッファは無効です。

ビット 16 **EN2** : DAC チャンネル 2 イネーブル

このビットは、DAC チャンネル 2 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 2 は無効です。

1 : DAC チャンネル 2 は有効です。

ビット 15:14 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 = 4095 のアンマスクビット [11:0]

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 : タイマ 7 TRGO イベント
011 : タイマ 9 TRGO イベント
100 : タイマ 2 TRGO イベント
101 : タイマ 4 TRGO イベント
110 : 外部ライン 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 から DAC_DOR1 レジスタへの転送には、1 APB1 クロックサイクルしか必要ありません。

ビット 1 **BOFF1** : DAC チャンネル 1 出力バッファディスエーブル

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。

13.5.2 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み													SWTRIG2	SWTRIG1	
													w	w	

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 **SWTRIG2** : DAC チャンネル 2 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_DHR2 レジスタの値が DAC_DOR2 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)。

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)。

13.5.3 DAC チャンネル 1 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				DACC1DHR[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 **DACC1DHR[11:0]** : DAC チャンネル 1 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。

13.5.4 DAC チャンネル 1 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC1DHR[11:0]												予約済み			
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 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。

13.5.5 DAC チャンネル 1 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								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 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。



13.5.6 DAC チャンネル 2 12 ビット右詰めデータ保持レジスタ (DAC_DHR12R2)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				DACC2DHR[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 **DACC2DHR[11:0]** : DAC チャンネル 2 12 ビット右詰めデータ

これらのビットは、DAC チャンネル 2 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。

13.5.7 DAC チャンネル 2 12 ビット左詰めデータ保持レジスタ (DAC_DHR12L2)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[11:0]												予約済み			
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 **DACC2DHR[11:0]** : DAC チャンネル 2 12 ビット左詰めデータ

これらのビットは、DAC チャンネル 2 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。



13.5.8 DAC チャンネル 2 8 ビット右詰めデータ保持レジスタ (DAC_DHR8R2)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								DACC2DHR[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 **DACC2DHR[7:0]** : DAC チャンネル 2 8 ビット右詰めデータ

これらのビットは、DAC チャンネル 2 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。

13.5.9 デュアル DAC 12 ビット右詰めデータ保持レジスタ (DAC_DHR12RD)

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
予約済み				DACC2DHR[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
予約済み				DACC1DHR[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、リセット値に保持する必要があります。

ビット 27:16 **DACC2DHR[11:0]** : DAC チャンネル 2 12 ビット右詰めデータ

これらのビットは、DAC チャンネル 2 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 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 11:0 **DACC1DHR[11:0]** : DAC チャンネル 1 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。



13.5.10 デュアル DAC 12 ビット左詰めデータ保持レジスタ (DAC_DHR12LD)

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
DACC2DHR[11:0]												予約済み			
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
DACC1DHR[11:0]												予約済み			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

- ビット 31:20 **DACC2DHR[11:0]** : DAC チャンネル 2 12 ビット左詰めデータ
 これらのビットは、DAC チャンネル 2 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 19:16 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 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。

13.5.11 デュアル DAC 8 ビット右詰めデータ保持レジスタ (DAC_DHR8RD)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DACC2DHR[7:0]								DACC1DHR[7:0]							
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 15:8 **DACC2DHR[7:0]** : DAC チャンネル 2 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 2 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。
- 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。

13.5.12 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				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。

13.5.13 DAC チャンネル 2 データ出力レジスタ(DAC_DOR2)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				DACC2DOR[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 **DACC2DOR[11:0]** : DAC チャンネル 2 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 2 のデータ出力を含みます。

13.5.14 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
予約済み		DMAUDR2	予約済み												
		rc_w1													
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み		DMAUDR1	予約済み												
		rc_w1													

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 **DMAUDR2** : DAC チャンネル 2 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 2 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 2 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 2 変換を駆動しています)。

ビット 28:14 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 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。

13.5.15 DAC レジスタマップ

表 68 に DAC レジスタの一覧を示します。

表 68. DAC レジスタマップ

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	予約済み		DMAUDRIE2	DMAEN2	MAMP2[3:0]				WAVE2 [2:0]		TSEL2[2:0]				TEN2	BOFF2	EN2	予約済み	DMAUDRIE1	DMAEN1	MAMP1[3:0]				WAVE 1[2:0]	TSEL1[2: 0]			TEN1	BOFF1	EN1	
0x04	DAC_ SWTRIGR	予約済み																													SWTRIG2	SWTRIG1	
0x08	DAC_ DHR12R1	予約済み																			DACC1DHR[11:0]												
0x0C	DAC_ DHR12L1	予約済み														DACC1DHR[11:0]										予約済み							
0x10	DAC_ DHR8R1	予約済み																						DACC1DHR[7:0]									
0x14	DAC_ DHR12R2	予約済み																			DACC2DHR[11:0]												
0x18	DAC_ DHR12L2	予約済み														DACC2DHR[11:0]										予約済み							
0x1C	DAC_ DHR8R2	予約済み																						DACC2DHR[7:0]									
0x20	DAC_ DHR12RD	予約済み			DACC2DHR[11:0]											予約済み			DACC1DHR[11:0]														
0x24	DAC_ DHR12LD	DACC2DHR[11:0]											予約済み			DACC1DHR[11:0]										予約済み							
0x28	DAC_ DHR8RD	予約済み														DACC2DHR[7:0]							DACC1DHR[7:0]										
0x2C	DAC_ DOR1	予約済み																			DACC1DOR[11:0]												
0x30	DAC_ DOR2	予約済み																			DACC2DOR[11:0]												
0x34	DAC_SR	予約済み		DMAUDR2	予約済み														DMAUDR1	予約済み													

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。



14 コンパレータ (COMP)

このセクションは、特に指定がない限り、STM32L1xxxxファミリ全体に適用されます。

14.1 概要

STM32L1xxxx は、同じ電流バイアスを共有する 2 つのゼロクロスコンパレータ COMP1 と COMP2 を内蔵しています。

注： コンパレータ入力として使用するすべての入出力では、GPIO レジスタをアナログモードに設定する必要があります。

ルーティングインタフェース ([セクション 8：システム設定コントローラ \(SYSCFG\) とルーティングインタフェース \(RI\) 参照](#)) を使用するときは、コンパレータ入力を外部入出力に接続することができます。

14.2 主な機能

- 固定の閾値 (内部基準電圧) を持つ 1 つのコンパレータ (COMP1)。非反転入力は、24 個の入出力から選択できます。
- 選択可能な閾値を持つ 1 つのレールツーレールコンパレータ (COMP2)。非反転入力は、2 個の入出力から選択できます。閾値は、次から選択できます。
 - 内部基準電圧 (V_{REFINT})
 - 内部基準電圧の約数 (1/4、1/2、3/4)
 - DAC1 出力
 - DAC2 出力
 - 外部 I/O (PB3)
- 2 つのコンパレータを組み合わせてウィンドウコンパレータにすることができます。
- トリガの設定に応じ、ゼロ交差によって、コンパレータ出力の立ち上がりエッジまたは立ち下がりエッジを生成することができます。
- 各コンパレータは、スリープおよびストップからのウェイクアップで割り込みを生成する機能を持ちます。
- COMP2 出力は、TIM2/TIM3/TIM4 の入力キャプチャ 4 (IC4) または OCREF_CLR 入力、または TIM10s 入力キャプチャ 1 (IC1) へリダイレクトすることができます。
- COMP2 の速度は、最適な速度/消費電力比率が得られるように設定できます。

コンパレータ、ルーティングインタフェース、ADC インタフェースの全ブロック図をセクションの [セクション 8：システム設定コントローラ \(SYSCFG\) とルーティングインタフェース \(RI\)](#) に示します。

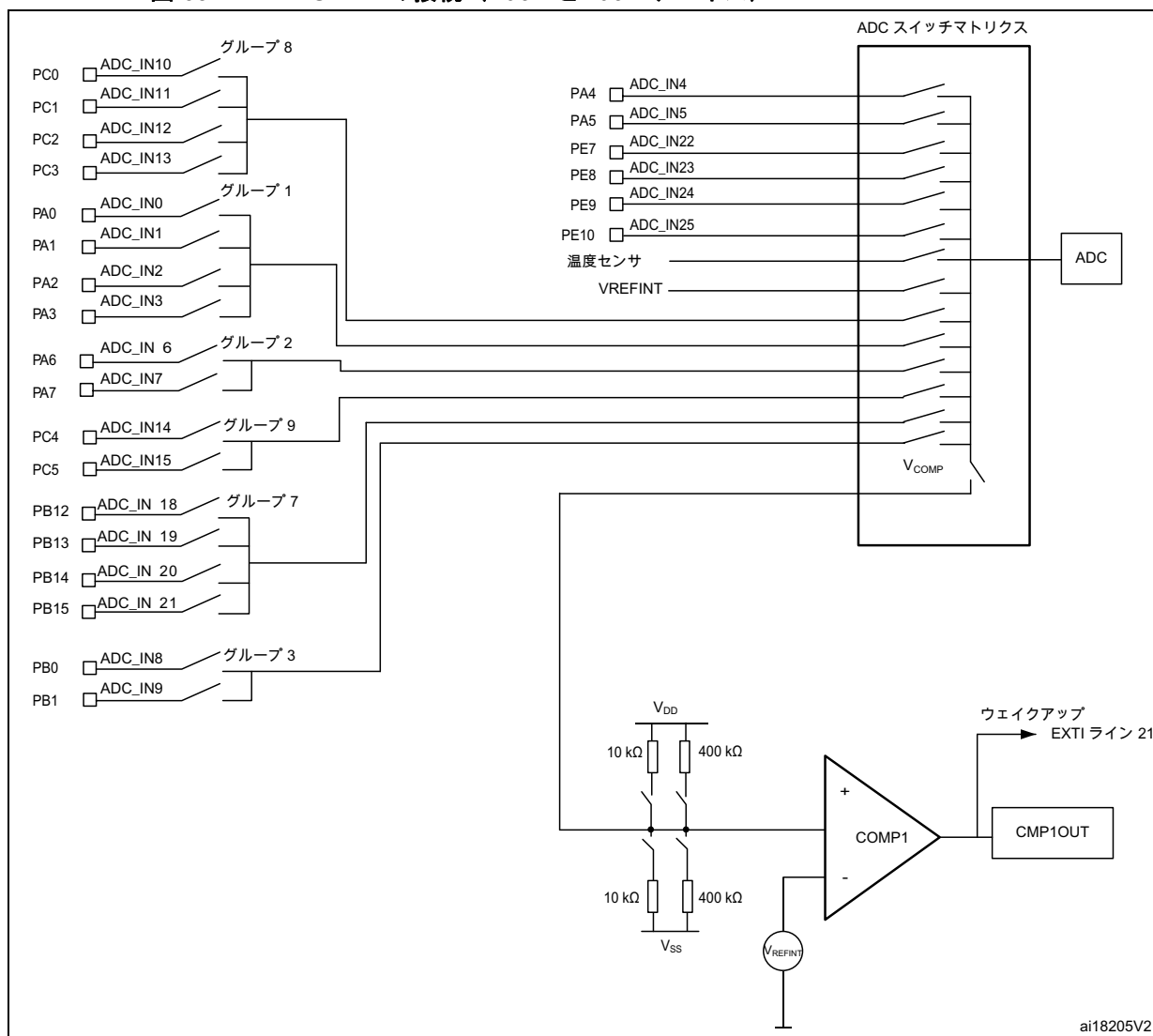
14.3 COMP クロック

クロックコントローラから供給される COMP クロックは、PCLK1 (APB1 クロック) に同期しています。

14.4 コンパレータ 1 (COMP1)

図 63 と 図 64 に、コンパレータ 1 の接続を示します。

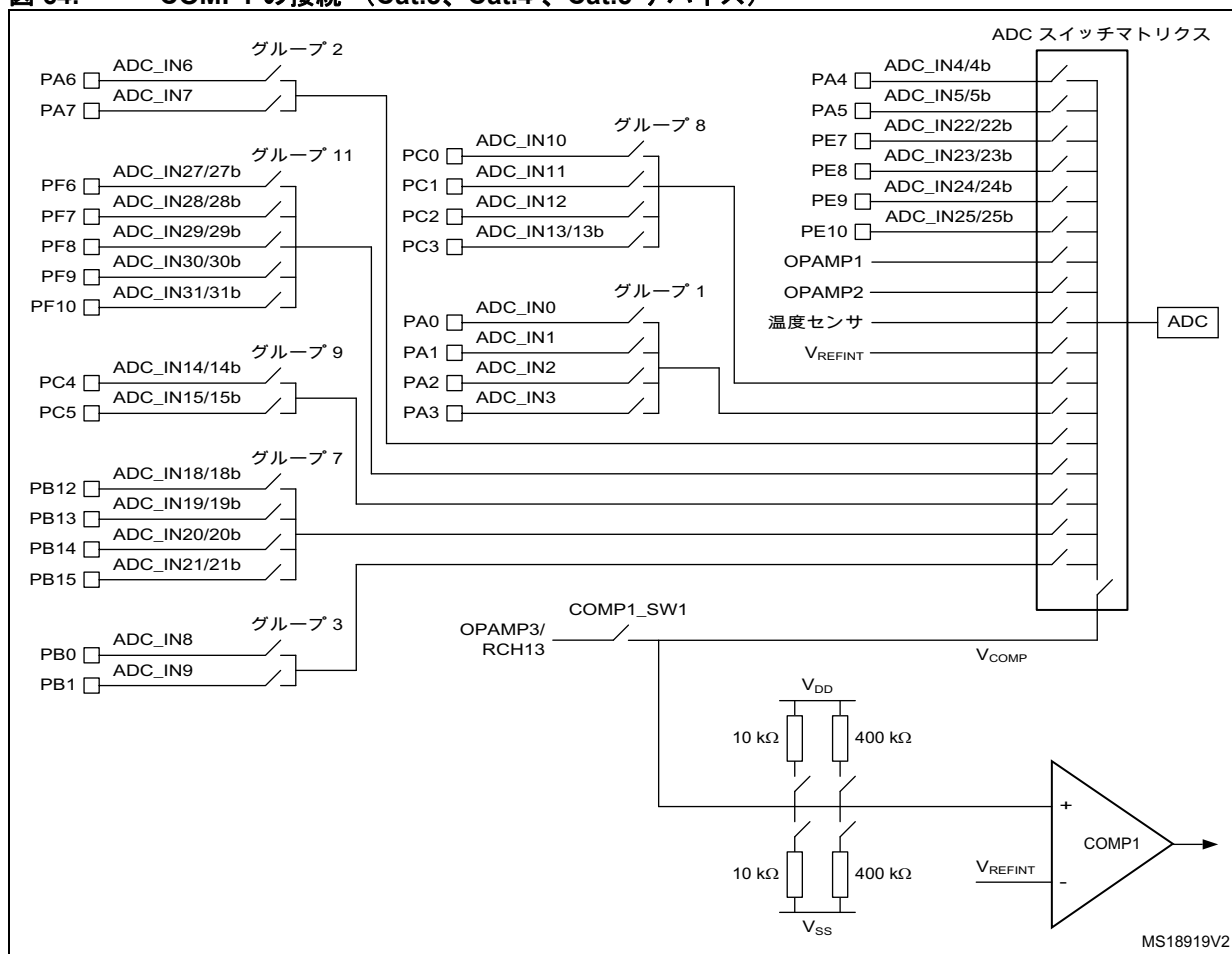
図 63. COMP1 の接続 (Cat.1 と Cat.2 デバイス)



注： 内部基準電圧と温度センサは、COMP1 非反転入力として使用することはできません。

COMP1 コンパレータと ADC は、ADC スイッチマトリクスを共用するため同時に使用することはできません。

図 64. COMP1 の接続 (Cat.3、Cat.4、Cat.5 デバイス)



注： 内部基準電圧と温度センサは、COMP1 非反転入力として使用することはできません。

COMP1 コンパレータと ADC は、ADC スイッチマトリックスを共用するため同時に使用できません。

COMP1 コンパレータを使用するときは、アプリケーションは次の手順を実行する必要があります。

1. COMP_CSR レジスタの CMP1EN ビットをセットして、コンパレータ 1 を有効化します。
2. コンパレータがレディになるまで (スタートアップ時間が過ぎるのを) 待ちます。詳細については、STM32L1xxx データシートの電気的特性を参照してください。
3. 対応する I/O スイッチが閉じている場合、ADC スイッチも閉じるように、RI_ASCR1 レジスタの SCM ビットをセットします。
4. ADC スイッチを閉じて、選択した入出力から非反転入力までのパスを作成します。入力としては最大 29 個の I/O が可能で、グループに分けることも、分けなくても可能です (図 40 : I/O グループと選択 (ページ) 202 参照) :
 - a) RI_ASCR1 レジスタの VCOMP ビットをセットして、VCOMP ADC アナログスイッチを閉じます。
 - b) RI_ASCR1 の CHn ビットをセットして、COMP1 非反転入力に接続する I/O グループに対応する I/O アナログスイッチ 番号 n を閉じます。
5. 必要に応じて、EXTI ライン 21 を割り込みモードに設定および有効化し、さらに所望のトリガイベント (立ち上がりエッジ、立ち下がりエッジまたは両方) を選択して、COMP1 割り込みを有効化します。

14.5 コンパレータ 2 (COMP2)

 65と 66 に、コンパレータ 2 の接続を示します。

図 65. COMP2 の接続 (Cat.1 と Cat.2 デバイス)

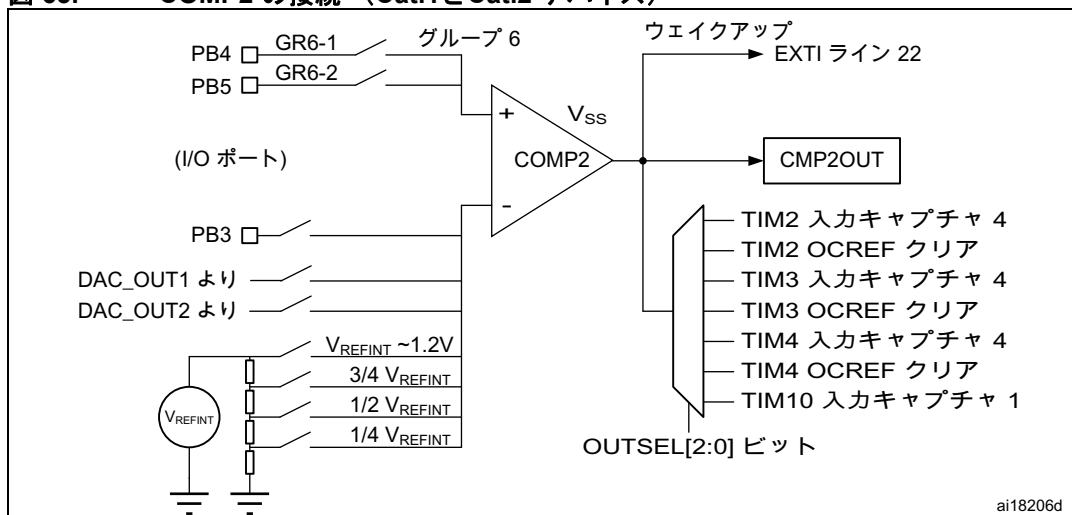
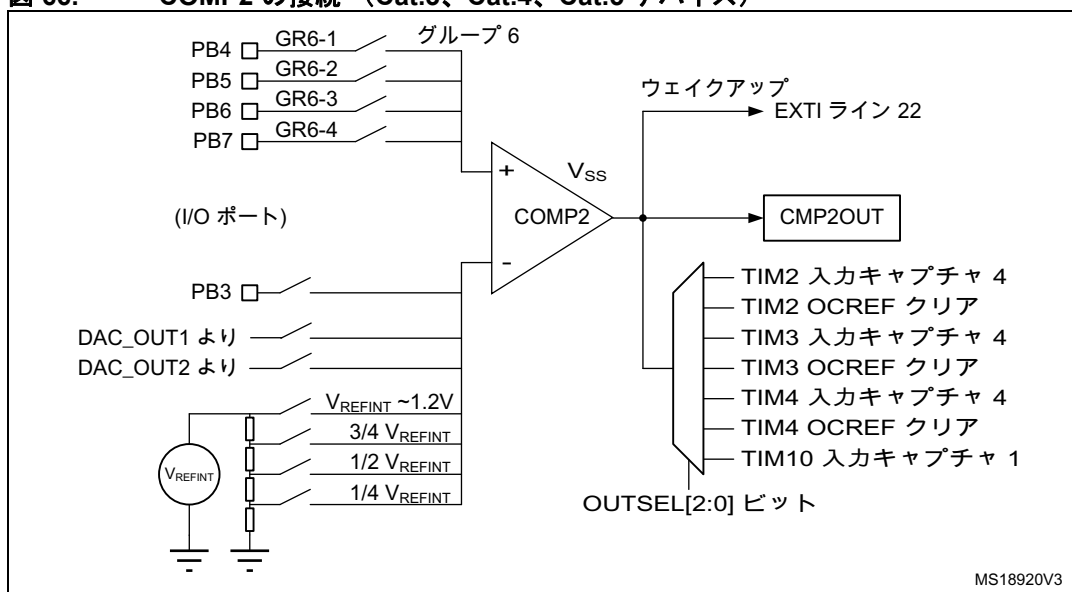


図 66. COMP2 の接続 (Cat.3、Cat.4、Cat.5 デバイス)



COMP2 コンパレータを使用するときは、アプリケーションは次の手順を実行する必要があります。

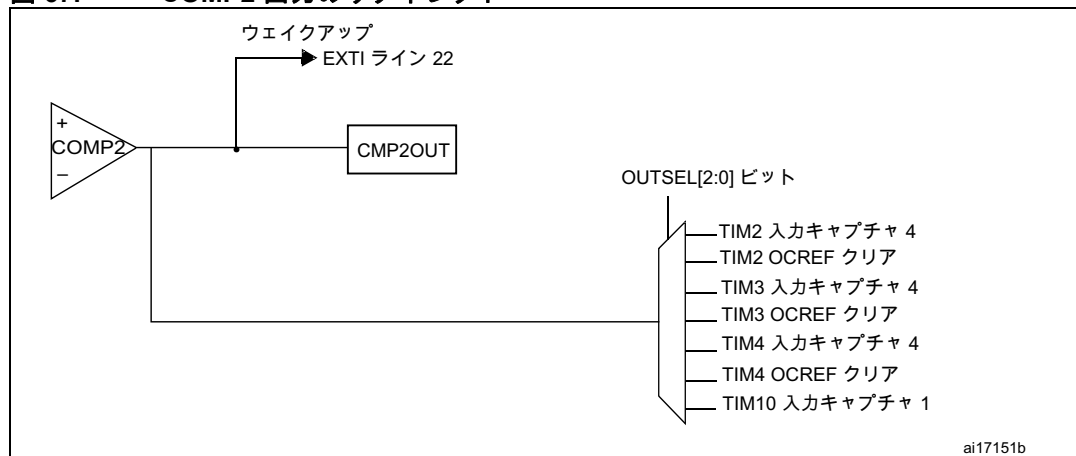
- COMP_CSR の INSEL[2:0] ビットを使って COMP2 の反転入力を選択します。
 - 外部 I/O 選択 (PB3 I/O) の場合、I/O はアナログ入力モードに設定する必要があります。
- I/O のアナログスイッチを閉じて、COMP2 非反転入力へ接続します。入力としては、グループ 6 内の I/O が可能です (表 40 : I/O グループと選択 (202 ページ) 参照)。GR6-1 スイッチまたは GR6-2 スイッチは、対応する I/O がアナログモードに設定されると直ちに閉じます。
- コンパレータがレディになるまで (スタートアップ時間が過ぎるのを) 待ちます。詳細については、STM32L1xxx データシートの電気的特性を参照してください。
- 必要に応じて、次の手順を実行します。
 - COMP_CSR の SPEED ビットを使って速度を選択します。
 - COMP_CSR の OUTSEL[2:0] ビットを設定して、COMP2 出力を TIM2、TIM3、TIM4 または TIM10 へリダイレクトします (図 67 参照)。
 - EXTI ライン 22 を割り込みモードに設定および有効化し、さらに所望の感度レベルを選択して、COMP2 割り込みを有効化します。

注 : 対応する I/O (PB4 または PB5) をアナログモードに設定するか (シュミットトリガを無効化)、または I/O を入力フローティングモードに設定し、さらに RI_ASCR2 の GR6-1 または GR6-2 をセットして (シュミットトリガを有効化)、GR6-1 と GR6-2 の I/O スイッチを閉じることができます。PB4 または PB5 をコンパレータ入力として使用する場合、 $V_{DD}/2$ での消費電力増加を防止するアナログ設定の使用が推奨されます。

注 : 反転入力を選択すると直ちに、COMP2 コンパレータが有効化されます。
コンパレータを有効化すると、チャンネルを変更することができます。

次の図に、COMP2 出力の可能な出力リダイレクトを示します。

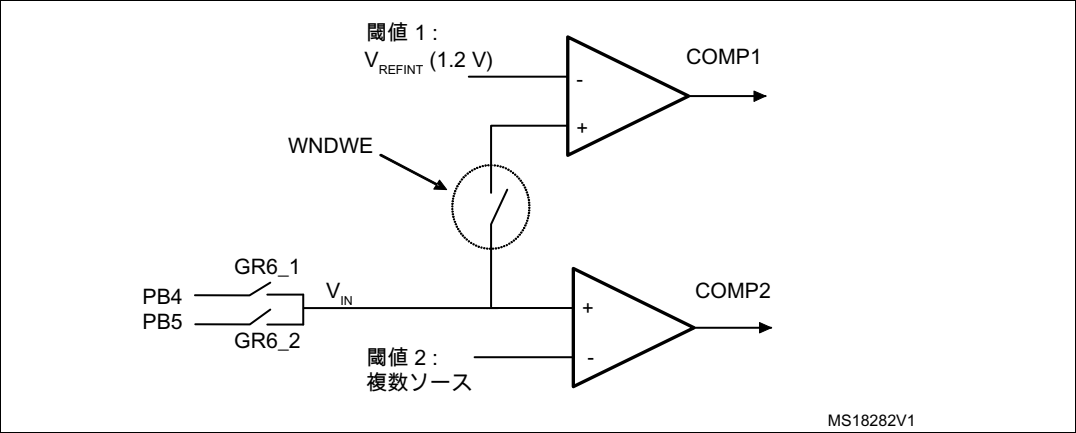
図 67. COMP2 出力のリダイレクト



注 : 「TIMx OCREF のクリア」の詳細については、セクション 17.3.11 : 外部イベントによる OCxREF 信号のクリア (419 ページ) を参照してください。

14.6 ウィンドウモードのコンパレータ

図 68. ウィンドウモードのコンパレータ



ウィンドウモードで COMP1 および COMP2 コンパレータを使用するときは、次の手順を実行する必要があります。

1. [セクション 14.5 : コンパレータ 2 \(COMP2\)](#) の説明に従い COMP2 の反転入力を選択します。
2. COMP_CSR レジスタの WNDWE をセットしてウィンドウモードを有効化します。
3. 非反転入力を選択します。
 - COMP1 : [セクション 14.4 : コンパレータ 1 \(COMP1\)](#) のステップ 2 と 3 に従います
 - COMP2 : [セクション 14.5 : コンパレータ 2 \(COMP2\)](#) のステップ 3 と 4 に従います
4. COMP_CSR レジスタの CMP1EN をセットして、COMP1 を有効化します。

注： ウィンドウモードでは、非反転入力としてグループ 6 (PB4 と PB5) のみを選択することができます。

14.7 低消費電力モード

表 69. 低消費電力モードでのコンパレータ動作

モード	説明
スリープ	コンパレータには影響ありません。 コンパレータ割り込みにより、デバイスはスリープモードから抜けます。
STOP	コンパレータには影響ありません。 コンパレータ割り込みにより、デバイスはストップモードから抜けます。

注： PWR_CR レジスタの ULP ビットを使って内部基準電圧を切り離した場合、コンパレータを使ってデバイスをスリープモードまたはストップモードから抜けさせることはできません。

14.8 割り込み

コンパレータ割り込みは EXTI コントローラに接続されます (ライン 21 と 22)。

COMP 割り込みを有効化するときは、次のシーケンスが必要です。

- EXTI ライン 21 (COMP1) または EXTI ライン 22 (COMP2) を割り込みモードに設定および有効化し、さらに所望のトリガイベント (立ち上がりエッジ、立ち下がりエッジ、または両方) を設定し、
- NVIC の COMP_IRQ チャンネルを設定し有効化します。

14.9 COMP 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。

14.9.1 COMP コンパレータ制御およびステータスレジスタ (COMP_CSR)

COMP_CSR レジスタは、コンパレータの制御/ステータスレジスタです。両コンパレータに関係する全ビットが配置されています。

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
TSUS P	CAIF	CAIE	RCH13	FCH8	FCH3	予約済み		OUTSEL[2:0]			INSEL[2:0]			WNDWE	VREFOUT EN
rw	r	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
予約済み		CMP2O UT	SPEED	予約済み				CMP1OUT	予約済み	COMP1 _SW1	CMP1E N	400KPD	10KPD	400KPU	10KPU
		r	rw					r		rw	rw	rw	rw	rw	

- ビット 31 **TSUSP** : タイマモード一時停止
0 : OC TIM9 の生成を一時停止するために TIM9 ITR を有効化
1 : OC TIM9 の生成を一時停止するために TIM9 ITR を使用しない
注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。
- ビット 30 **CAIF** : チャンネルアキュイジション割り込みフラグ
0 : チャンネルアキュイジション進行中または未起動
1 : チャンネルアキュイジション完了
注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。
- ビット 29 **CAIE** : チャンネルアキュイジション割り込みイネーブル/クリア
チャンネルアキュイジション割り込みを有効化/無効化するときに、このビットがセット/クリアされます。
Caif ビットがセットされた場合、CAIE ビットに 0 を書き込んでクリアする必要があります。
0 : チャンネルアキュイジション割り込みを無効化
1 : チャンネルアキュイジション割り込みを有効化
このビットは Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 28 **RCH13** : GPIO ポート PC3 をルート変更される ADC 入力チャンネル CH13 として選択

OPAMP3 がパワーダウンモードにある場合 (OPAMP_CSR レジスタで OPA1PD ビット = 0 (Cat.4 と Cat.5 デバイスの場合))、PC3 をルート変更されるチャンネル CH13 (ADC インタフェースから選択) として使用するために、このビットをソフトウェアからセット/クリアします。図 71 : OPAMP3 信号ルーティング (Cat.4 デバイスの場合) (ページ) 357 参照。

0 : PC3 を低速 ADC チャンネルとして使用可能

1 : PC3 をルート変更される ADC チャンネルとして使用可能

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 27 **FCH8**: GPIO ポート PB0 を高速 ADC 入力チャンネル CH8 として選択。

OPAMP2 がパワーダウンモードにある場合 (OPAMP_CSR レジスタで OPA2PD ビット = 0)、PB0 をダイレクトチャンネル CH13 (ADC インタフェースから選択) として使用するために、このビットをソフトウェアからセット/クリアします。図 70 : OPAMP2 の信号ルーティング (ページ) 356 参照。

0 : PB0 を低速 ADC チャンネルとして使用可能

1 : PB0 を高速 ADC チャンネルとして使用可能

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 26 **FCH3** : GPIO ポート PA3 を高速 ADC 入力チャンネル CH3 として選択

OPAMP1 がパワーダウンモードにある場合 (OPAMP_CSR レジスタで OPA1PD ビット = 0)、PA3 をダイレクトチャンネル CH13 (ADC インタフェースから選択) として使用するために、このビットをソフトウェアからセット/クリアします。図 69 : OPAMP1 の信号ルーティング (ページ) 356 参照。

0 : PA3 を低速 ADC チャンネルとして使用可能

1 : PA3 を高速 ADC チャンネルとして使用可能

注 : このビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

ビット 25:24 予約済み、クリア状態を保つ必要があります。

ビット 23:21 **OUTSEL** : コンパレータ 2 出力選択

これらのビットをソフトウェアから書き込んで、COMP2 出力を選択したタイマ入力に接続します。

000 = TIM2 入力キャプチャ 4

001 = TIM2 OCREF_CLR

010 = TIM3 入力キャプチャ 4

011 = TIM3 OCREF_CLR

100 = TIM4 入力キャプチャ 4

101 = TIM4 OCREF_CLR

110 = TIM10 入力キャプチャ 1

111 = リダイレクトなし

ビット 20:18 **INSEL** : 反転入力選択

000 = 選択なし

001 = 外部 I/O : PB3 (COMP2_INM)

010 = V_{REFINT}

011 = $3/4 V_{REFINT}$

100 = $1/2 V_{REFINT}$

101 = $1/4 V_{REFINT}$

110 = DAC_OUT1

111 = DAC_OUT2

注 : INSEL ビット値が「000」と異なる場合、COMP2 コンパレータが有効化されます。

ビット 17 **WNDWE** : ウィンドウモードドイネーブル

0 : 無効

1 : 有効

ビット 16 **VREFOUTEN** : V_{REFINT} 出力イネーブル

このビットを使って V_{REFINT} をグループ 3 へ出力します (図 28 : 内部基準電圧の出力参照)。

0 : 無効

1 : 有効

ビット 15:14 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 **CMP2OUT** : コンパレータ 2 出力

このビットは、コンパレータ 2 出力のハイレベルまたはローレベルを示します。

0 : 非反転入力電圧が反転入力電圧より低いとき、コンパレータ 2 出力がローレベルになります。

1 : 非反転入力電圧が反転入力電圧より高いとき、コンパレータ 2 出力がハイレベルになります。

ビット 12 **SPEED** : コンパレータ 2 速度モード

0 : 低速

1 : 高速

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 **CMP1OUT** : コンパレータ 1 出力

このビットは、コンパレータ 1 出力のハイレベルまたはローレベルを示します。

0 : 非反転入力電圧が反転入力電圧より低いとき、コンパレータ 1 出力がローレベルになります。

1 : 非反転入力電圧が反転入力電圧より高いとき、コンパレータ 1 出力がハイレベルになります。

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 **SW1** : COMP1_SW1 アナログスイッチイネーブル

ソフトウェアからこのビットをセット/クリアして、OPAMP3 出力または PC3 を ADC スイッチマトリックスおよび/または COMP1 の反転入力ヘリダイレクトするため COMP1_SW1 アナログスイッチを制御します。

0 : COMP1_SW1 アナログスイッチを開きます。

1 : COMP1_SW1 アナログスイッチを閉じます。

注 : このビットは、Cat.4 と Cat.5 デバイスでのみ使用可能です。

ビット 4 **CMP1EN** : コンパレータ 1 イネーブル

0 : コンパレータ 1 を無効化

1 : コンパレータ 1 を有効化

ビット 3 **400KPD** : 400 k Ω プルダウン抵抗

このビットで 400 k Ω プルダウン抵抗を有効化します。

0 : 400 k Ω プルダウン抵抗を無効化

1 : 400 k Ω プルダウン抵抗を有効化

ビット 2 **10KPD** : 10 k Ω プルダウン抵抗

このビットで 10 k Ω プルダウン抵抗を有効化します。

0 : 10 k Ω プルダウン抵抗を無効化

1 : 10 k Ω プルダウン抵抗を有効化

ビット 1 **400KPU** : 400 k Ω プルアップ抵抗

このビットで 400 k Ω プルアップ抵抗を有効化します。

0 : 400 k Ω プルアップ抵抗を無効化

1 : 400 k Ω プルアップ抵抗を有効化

ビット 0 **10KPU** : 10 k Ω プルアップ抵抗

このビットで 10 k Ω プルアップ抵抗を有効化します。

0 : 10 k Ω プルアップ抵抗を無効化

1 : 10 k Ω プルアップ抵抗を有効化

注 : 余分な電力消費を避けるには、一度に 1 つの抵抗だけを有効にしてください。



14.9.2 COMP レジスタマップ

表 70 : COMP レジスタマップとリセット値に COMP レジスタの一覧を示します。

表 70. 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			
0x00	COMP_CSR	TSUSP	CAIF	CAIE	RCH13	FCH8	FCH3	予約 済み		OUTSEL [2:0]			INSEL [2:0]			WINDWE		VREFOUTEN	予約済み		CMP2OUT	SPEED	予約済み						CMP1OUT	予約済み	COMP1_SW1	CMP1EN	400KPD	10KPD	400KPU	10KPU
	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

レジスタ境界アドレステーブルについては、表 5 (47 ページ) 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。

15 オペアンプ (OPAMP)

このセクションは、Cat.3、Cat.4、Cat.5 デバイスにのみ適用されます。使用可能な OPAMP については、デバイスデータシートを参照してください。

15.1 OPAMP の概要

この MCU は、外部または内部フォロアルーティング機能（または外付け部品によるアンプおよびフィルタ機能）付きの 3 個のオペアンプを内蔵しています。オペアンプが 1 つ選択されると、1 つの外部 ADC チャンネルを使って、出力の測定が可能になります。

15.2 OPAMP の主な機能

- レールツーレールの入力および出力の電圧範囲
- 低入力バイアス電流
- 低入力オフセット電圧
- 低電力モード

15.3 OPAMP の機能詳細

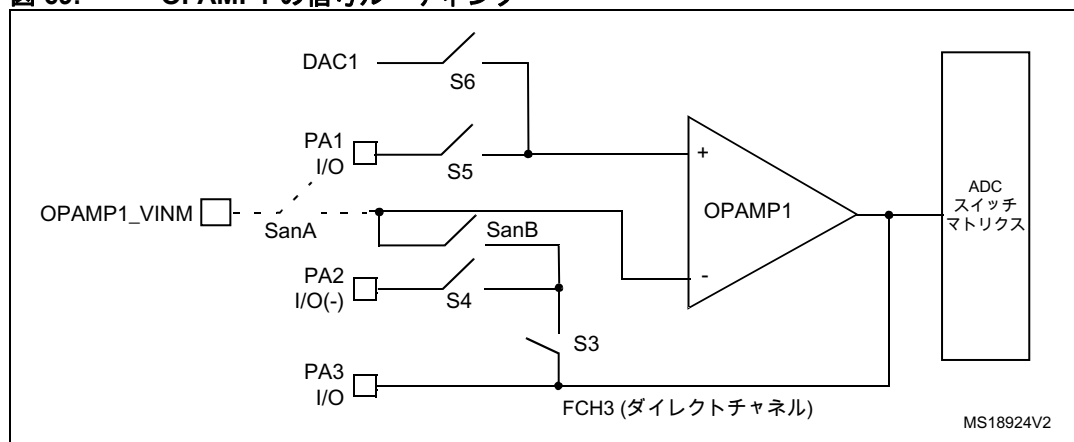
Cat.4 デバイスでは 3 個のオペアンプ（OPAMP1、OPAMP2、OPAMP3）が、Cat.3 と Cat.5 デバイスでは 2 個のオペアンプ（OPAMP1 と OPAMP2）が、それぞれ使用可能です。専用 I/O を使った接続を次に示します。

- OPAMP1_VINP --> PA1
- OPAMP1_VINM --> PA2^(a)
- OPAMP1_VOUT --> PA3 (ADC 入力 CH3)
- OPAMP2_VINP --> PA6
- OPAMP2_VINM --> PA7^(a)
- OPAMP2_VOUT --> PB0 (ADC 入力 CH8)
- OPAMP3_VINP --> PC1
- OPAMP3_VINM --> PC2^(a)
- OPAMP3_VOUT --> PC3 (ADC 入力 CH13)

a. またはパッケージによっては専用 OPAMPx_VINM ピンがある場合があります。

15.3.1 信号ルーティング

図 69. OPAMP1 の信号ルーティング



3 個のオペアンプのルーティングは、OPAMP_CSR レジスタによって選択することができます。

OPA1_PD ビット、OPA2_PD ビットおよび/または OPA3_PD ビットによりアンプが有効化されているか否かによらず、対応する OPAMP_CSR レジスタビットを設定して、アナログスイッチ S3 から S6 と SanA を開閉することができます。

アナログスイッチ SanB は、S3 スイッチまたは S4 スイッチの選択に自動的に従います。これは個別に制御されません。

OPAMP1 の場合、S6 を使って DAC1 を正入力に接続します。

OPAMP2 の場合、I/O または DAC1 または DAC2 として正入力ソースを選択するため、S7 スイッチが S6 と並列に追加されます。

OPAMP3 の場合、S6 を使って DAC2 を正入力に接続します。

OPAx_PD ビットをセットして、すべてのオペアンプをパワーダウンさせることができます。対応する入力と出力は高インピーダンスになります。

図 70. OPAMP2 の信号ルーティング

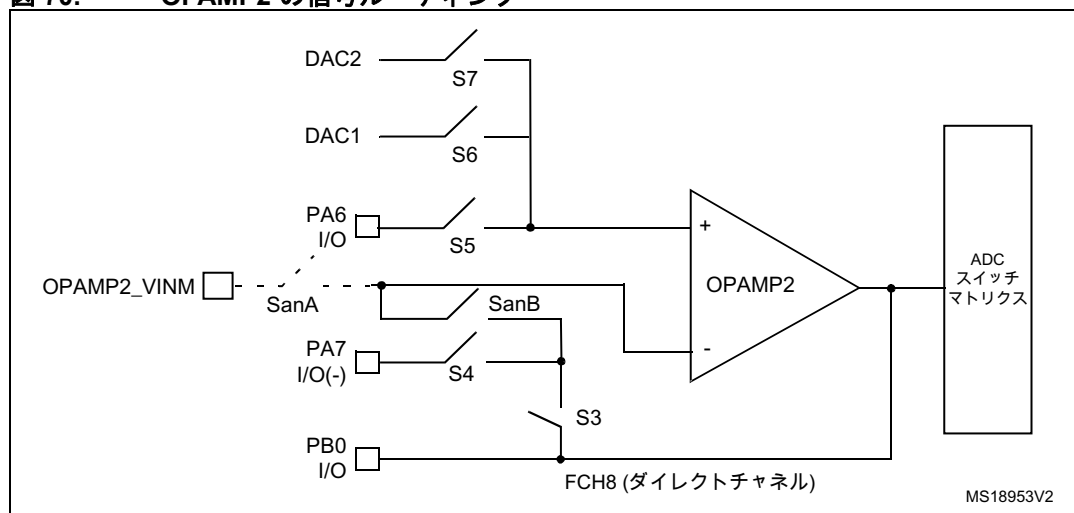
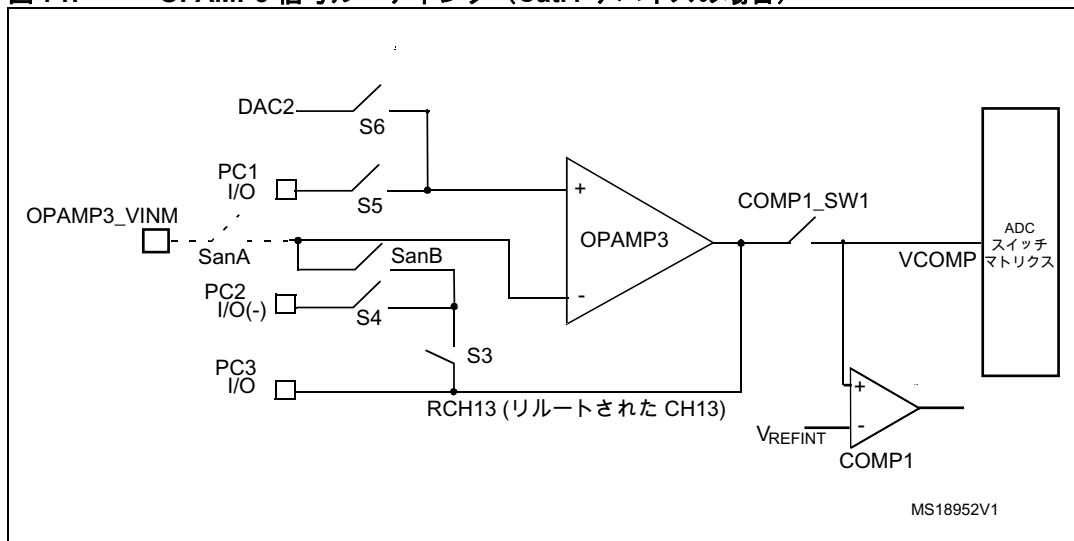


図 71. OPAMP3 信号ルーティング (Cat.4 デバイスの場合)



15.3.2 ADC 入力として OPAMP 出力を使用

OPAMP 出力を ADC 入力として使用するためには、オペアンプを有効化し、ADC では OPAMP 出力チャンネル番号を使用する必要があります。(OPA1 : CH3 ; OPA2 : CH8 ; OPA3 : CH13)。

OPA3 または FCH13 の他に、アキュイジションを行うためには COMP1_SW1 アナログスイッチを閉じる必要があります ([セクション 14.9.1 : COMP コンパレータ制御およびステータスレジスタ \(COMP_CSR\) \(351 ページ\)](#) 参照)。

15.3.3 較正

スタートアップ時、トリミング値は工場出荷時のプリセットトリミング値で初期化されます。

さらに、ユーザは各オペアンプのオフセットをトリミングすることができます。トリミング動作中、各オペアンプ入力に関係するすべてのスイッチは開いている必要があります (SanA、S3、S4、S5、S6)。

通常モードと低消費電力モードに対して、3 個のオペアンプのオフセットをトリミングする 2 個のレジスタがあります。OPAMP_OTR レジスタと OPAMP_LPOTR レジスタには、1 ワードが標準モードに、他の 1 ワードが低消費電力モードに、計 2 ワード (各 30 ビット) があります。これは"ユーザ"値です。

OPAMP_OTR レジスタの OT_USER ビットを使って、"工場出荷時"値から"ユーザ"トリミング値へ切り替えることができます。このビットは、"工場出荷時"値を OPAMP へ送信するためスタートアップ時にリセットされます。この値は、3 個の OPAMP に共通です。

[通常モードの OPAMP オフセットトリミングレジスタ \(OPAMP_OTR\)](#) を設定して、各オペアンプのオフセットをトリミングすることができます。トリミング値は不揮発性メモリに格納されます。

通常、OPAx_CAL ビットから初期化される較正動作の後オフセットトリミングレジスタに書き込みを行うことができます。

- OPAxCAL_L ビットをセットすると、P 差動対のオフセット較正が初期化されます (低電圧基準を使用)。
- OPAxCAL_H ビットをセットすると、N 差動対のオフセット較正が初期化されます (高電圧基準を使用)。

OPA_OTR または OPA_LP_OTR の有効な 30 ビットは、3 個の 10 ビットワードで構成されています (各オペアンプに 1 ワード)。各 10 ビットは 2 つの較正值から構成され、下位 5 ビットは PMOS 差動対のオフセットトリミングです。上位 5 ビットは NMOS 差動対のオフセットトリミングです。

表 71 に示すように制御ビットをセットして、オフセット較正を初期化した後、OPAxCALOUT フラグが反転して、較正の正常終了が示されるまで、新しいトリミング値を **通常モードの OPAMP オフセットトリミングレジスタ (OPAMP_OTR)** レジスタ値に書き込みます。

表 71. 動作モードと較正

モード	制御ビット				出力	
	OPAxPD	OPAxLPM	OPAxCAL_H	OPAxCAL_L	V _{OUT}	CALout フラグ
通常動作モード	0	0	0	0	アナログ	0
			1	1		
低電力モード	0	1	0	0	アナログ	0
			1	1		
パワーダウン	1	X	X	X	Z	0
オフセット較正ハイ	0	X	1	0	アナログ	X
オフセット較正ロー	0	X	0	1	アナログ	X

較正手順

オペアンプの 1 つをフル較正するときは次のステップに従ってください。

1. OPAMP_CSR レジスタを設定して、オペアンプに接続されているすべてのスイッチを開きます。
2. OPAMP_OTR レジスタの OT_USER ビットに 1 をセットします。
3. 較正モードを選択します (表 71: 動作モードと較正参照)。次から開始することができます。
 - 通常モード、オフセット較正ハイ
これを実行するときは、OPAMP_CSR レジスタで OPAxPD=0、OPAxLPM=0、OPAxCAL_H=1、OPAxCAL_L=0 を設定します。
4. OPAMP_OTR[OPAxOPT_OFFSET_TRIM_High] のコードは、00000b から OPAxCALOUT 出力レベルを 0 から 1 へ変化させる先頭値コードヘインクリメントされます。

注: OPAMP_OTR レジスタへの書き込みと OPAxCALOUT 値の読み出しとの間で、データシートの電気的特性セクションで規定される $t_{OFFTRIM}$ 最大遅延を待つように注意して、正しい OPAxCALOUT 値を取得してください。

コミュテーションとは、オフセットが正しく補償され、かつ対応するトリミングコードが OPAMP_OTR レジスタに保持されている必要があることを意味します。

値 11111b は、OPAMP_OTR[OPAxOPT_OFFSET_TRIM_High] に対して禁止されています。

次に対してステップ 3 から 4 を繰り返します。

- Normal_mode およびオフセット較正ロー
- 低消費電力モードおよびオフセット較正ハイ
- 低消費電力モードおよびオフセット較正ロー

使用しないモードがある場合、その較正には意味がありません。

注: 較正フェーズ中、オペアンプ出力の外部接続は、500 μ A より大きい電流をプルアップまたはプルダウンすることはできません。

15.4 OPAMP レジスタ

15.4.1 OPAMP 制御／ステータスレジスタ (OPAMP_CSR)

アドレスオフセット : 0x00

リセット値 : 0x0001 0101

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OPA3CALOUT	OPA2CALOUT	OPA1CALOUT	OPA_RANGE	S7SEL2	ANAWSEL3	ANAWSEL2	ANAWSEL1	OPA3LPM	OPA3CAL_H	OPA3CAL_L	S6SEL3	S5SEL3	S4SEL3	S3SEL3	OPA3PD
r	r	r		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
OPA2LPM	OPA2CAL_H	OPA2CAL_L	S6SEL2	S5SEL2	S4SEL2	S3SEL2	OPA2PD	OPA1LPM	OPA1CAL_H	OPA1CAL_L	S6SEL1	S5SEL1	S4SEL1	S3SEL1	OPA1PD
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31 **OPA3CALOUT** : OPAMP3 較正出力

較正モード時、この信号が反転するとオフセットがトリミングされます。

ビット 30 **OPA2CALOUT** : OPAMP2 較正出力

較正モード時、この信号が反転するとオフセットがトリミングされます。

ビット 29 **OPA1CALOUT** : OPAMP1 較正出力

較正モード時、この信号が反転するとオフセットがトリミングされます。

ビット 28 **OPA_RANGE** : 電源レンジの選択

オペアンプがパワーダウンしているとき、ソフトウェアによってこのビットをセット/クリアすることができます。安定なオペアンプ電源レンジを選択します。

0 : 低レンジ ($V_{DDA} < 2.4\text{ V}$)

1 : 高レンジ ($V_{DDA} > 2.4\text{ V}$)

ビット 27 **S7SEL2** : OPAMP2 用スイッチ 7 イネーブル

0 : S7 を開く

1 : S7 を閉じる

ビット 26 **ANAWSEL3**:OPAMP3 用スイッチ SanA イネーブル

0 : SanA スイッチを開く

1 : SanA スイッチを閉じる

ビット 25 **ANAWSEL2**:OPAMP2 用スイッチ SanA イネーブル

0 : SanA スイッチを開く

1 : SanA スイッチを閉じる

ビット 24 **ANAWSEL1**:OPAMP1 用スイッチ SanA イネーブル

0 : SanA スイッチを開く

1 : SanA スイッチを閉じる

ビット 23 **OPA3LPM** : OPAMP3 低消費電力モード

0 : OPAMP3 低消費電力モードをオフ

1 : OPAMP3 低消費電力モードをオン

ビット 22 **OPA3CAL_H** : N 差動対の OPAMP3 オフセット較正

0 : N 差動対の OPAMP3 オフセット較正をオフ

1 : OPA3CAL_L = 0 の場合 N 差動対の OPAMP3 オフセット較正をオン

ビット 21 **OPA3CAL_L** : P 差動対の OPAMP3 オフセット較正

0 : P 差動対の OPAMP3 オフセット較正をオフ

1 : OPA3CAL_H = 0 の場合 P 差動対の OPAMP3 オフセット較正をオン



- ビット 20 **S6SEL3** : OPAMP3 用スイッチ 6 イネーブル
0 : S6 スイッチを開く
1 : S6 スイッチを閉じる
- ビット 19 **S5SEL3** : OPAMP3 用スイッチ 5 イネーブル
0 : S5 スイッチを開く
1 : S5 スイッチを閉じる
- ビット 18 **S4SEL3** : OPAMP3 用スイッチ 4 イネーブル
0 : S4 スイッチを開く
1 : S4 スイッチを閉じる
- ビット 17 **S3SEL3** : OPAMP3 用スイッチ 3 イネーブル
0 : S3 スイッチを開く
1 : S3 スイッチを閉じる
- ビット 16 **OPA3PD** : OPAMP3 パワーダウン
0 : OPAMP3 を有効化
1 : OPAMP3 を無効化
- ビット 15 **OPA2LPM** : OPAMP2 低消費電力モード
0 : OPAMP2 低消費電力モードをオフ
1 : OPAMP2 低消費電力モードをオン
- ビット 14 **OPA2CAL_H** : N 差動対の OPAMP2 オフセット較正
0 : N 差動対の OPAMP2 オフセット較正をオフ
1 : OPA2CAL_L = 0 の場合 N 差動対の OPAMP2 オフセット較正をオン
- ビット 13 **OPA2CAL_L** : P 差動対の OPAMP2 オフセット較正
0 : P 差動対の OPAMP2 オフセット較正をオフ
1 : OPA2CAL_H = 0 の場合 P 差動対の OPAMP2 オフセット較正をオン
- ビット 12 **S7SEL2** : OPAMP2 用スイッチ 6 イネーブル
0 : S6 スイッチを開く
1 : S6 スイッチを閉じる
- ビット 11 **S7SEL2** : OPAMP2 用スイッチ 5 イネーブル
0 : S5 スイッチを開く
1 : S5 スイッチを閉じる
- ビット 10 **S7SEL2** : OPAMP2 用スイッチ 4 イネーブル
0 : S4 スイッチを開く
1 : S4 スイッチを閉じる
- ビット 9 **S7SEL2** : OPAMP2 用スイッチ 3 イネーブル
0 : S3 スイッチを開く
1 : S3 スイッチを閉じる
- ビット 8 **OPA2PD** : OPAMP2 パワーダウン
0 : OPAMP2 を有効化
1 : OPAMP2 を無効化
- ビット 7 **OPA1LPM** : OPAMP1 低消費電力モード
0 : OPAMP1 低消費電力モードをオフ
1 : OPAMP1 低消費電力モードをオン
- ビット 6 **OPA1CAL_H** : N 差動対の OPAMP1 オフセット較正
0 : N 差動対の OPAMP1 オフセット較正をオフ
1 : OPA1CAL_L = 0 の場合 N 差動対の OPAMP1 オフセット較正をオン

- ビット 5 **OPA1CAL_L** : P 差動対の OPAMP1 オフセット較正
- 0 : P 差動対の OPAMP1 オフセット較正をオフ
 - 1 : OPA1CAL_H = 0 の場合 P 差動対の OPAMP1 オフセット較正をオン
- ビット 4 **S6SEL1** : OPAMP1 用スイッチ 6 イネーブル
- 0 : S6 スイッチを開く
 - 1 : S6 スイッチを閉じる
- ビット 3 **S5SEL1** : OPAMP1 用スイッチ 5 イネーブル
- 0 : S5 スイッチを開く
 - 1 : S5 スイッチを閉じる
- ビット 2 **S4SEL1** : OPAMP1 用スイッチ 4 イネーブル
- 0 : S4 スイッチを開く
 - 1 : S4 スイッチを閉じる
- ビット 1 **S43SEL1** : OPAMP1 用スイッチ 3 イネーブル
- 0 : S3 スイッチを開く
 - 1 : S3 スイッチを閉じる
- ビット 0 **OPA1PD** : OPAMP1 パワーダウン
- 0 : OPAMP1 を有効化
 - 1 : OPAMP1 を無効化

15.4.2 通常モードの OPAMP オフセットトリミングレジスタ (OPAMP_OTR)

アドレスオフセット : 0x04

ビット 31 リセット値 : 0

ビット 29:0 リセット値 : 工場出荷時トリミング値に戻します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OT_USER	予約済み	OA3_OPT_OFFSET_TRIM_HIGH					OA3_OPT_OFFSET_TRIM_LOW					OA2_OPT_OFFSET_TRIM_HIGH[4:1]			
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
OA2_OPT_OFFSET_TRIM_HIGH	OA2_OPT_OFFSET_TRIM_LOW					OA1_OPT_OFFSET_TRIM_HIGH					OA1_OPT_OFFSET_TRIM_LOW				
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 **OT_USER** ユーザトリミング値または工場出荷時トリミング値の選択
- このビットはソフトウェアからセット/クリアし、読み出すと常に 0 が返されます。工場出荷時設定トリミング値またはユーザ設定トリミング値のいずれで OPAMPx オフセットをトリミングするかを選択します。
- 0 : デフォルトの工場出荷時値を使って OPAMP オフセットをトリミング
- 1 : ユーザ設定値を使って OPAMP オフセットをトリミング
- ビット 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 29:25 **OA3_OPT_OFFSET_TRIM_HIGH[4:0]** : OPAMP3、NMOS 対の通常モード 5 ビットオフセットトリミング値
- ビット 24:20 **OA3_OPT_OFFSET_TRIM_LOW[4:0]** : OPAMP3、PMOS 対の通常モード 5 ビットオフセットトリミング値
- ビット 19:15 **OA2_OPT_OFFSET_TRIM_HIGH[4:0]** : OPAMP2、NMOS 対の通常モード 5 ビットオフセットトリミング値
- ビット 14:10 **OA2_OPT_OFFSET_TRIM_LOW[4:0]** : OPAMP2、PMOS 対の通常モード 5 ビットオフセットトリミング値
- ビット 9:5 **OA1_OPT_OFFSET_TRIM_HIGH[4:0]** : OPAMP1、NMOS 対の通常モード 5 ビットオフセットトリミング値
- ビット 4:0 **OA1_OPT_OFFSET_TRIM_LOW[4:0]** : OPAMP1、PMOS 対の通常モード 5 ビットオフセットトリミング値

15.4.3 低消費電力モードの OPAMP オフセットトリミングレジスタ (OPAMP_LPOTR)

アドレスオフセット : 0x08

ビット 29:0 リセット値 : 工場出荷時トリミング値に戻します。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み		OA3_OPT_OFFSET_TRIM_LP_HIGH					OA3_OPT_OFFSET_TRIM_LP_LOW					OA2_OPT_OFFSET_TRIM_LP_HIGH[4:1]			
		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
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OA2_OPT_OFFSET_TRIM_LP_HIGH0	OA2_OPT_OFFSET_TRIM_LP_LOW					OA1_OPT_OFFSET_TRIM_LP_HIGH					OA1_OPT_OFFSET_TRIM_LP_LOW				
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 29:25 **OA3_OPT_OFFSET_TRIM_LP_HIGH[4:0]** : OPAMP3、NMOS 対の低消費電力モード 5 ビットオフセットトリミング値

ビット 24:20 **OA3_OPT_OFFSET_TRIM_LP_LOW[4:0]** : OPAMP3、PMOS 対の低消費電力モード 5 ビットオフセットトリミング値

ビット 19:15 **OA2_OPT_OFFSET_TRIM_LP_HIGH[4:0]** : OPAMP2、NMOS 対の低消費電力モード 5 ビットオフセットトリミング値

ビット 14:10 **OA2_OPT_OFFSET_TRIM_LP_LOW[4:0]** : OPAMP2、PMOS 対の低消費電力モード 5 ビットオフセットトリミング値

ビット 9:5 **OA1_OPT_OFFSET_TRIM_LP_HIGH[4:0]** : OPAMP1、NMOS 対の低消費電力モード 5 ビットオフセットトリミング値

ビット 4:0 **OA1_OPT_OFFSET_TRIM_LP_LOW[4:0]** : OPAMP1、PMOS 対の低消費電力モード 5 ビットオフセットトリミング値



15.4.4 OPAMP レジスタマップ

表 72. OPAMP レジスタマップ

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	OPAMP_ CSR	OPA3CALOUT	OPA2CALOUT	OPA1CALOUT	OPA_RANGE	S7SEL2	ANAWSEL3	ANAWSEL2	ANAWSEL1	OPA3LPM	OPA3CAL_H	OPA3CAL_L	S6SEL3	S5SEL3	S4SEL3	S3SEL3	OPA3PD	OPA2LPM	OPA2CAL_H	OPA2CAL_L	S6SEL2	S5SEL2	S4SEL2	S3SEL2	OPA2PD	OPA1LPM	OPA1CAL_H	OPA1CAL_L	S6SEL1	S5SEL1	S4SEL1	S3SEL1	OPA1PD
0x04	OPAMP_ OTR	OT_USER	予約済み	OA3_OPT_OFFSET_TRIM_HIGH					OA3_OPT_OFFSET_TRIM_LOW					OA2_OPT_OFFSET_TRIM_HIGH				OA2_OPT_OFFSET_TRIM_LOW				OA1_OPT_OFFSET_TRIM_HIGH				OA1_OPT_OFFSET_TRIM_LOW							
0x08	OPAMP_ LPOTR	予約済み		OA3_OPT_OFFSET_TRIM_LP_HIGH					OA3_OPT_OFFSET_TRIM_LP_LOW					OA2_OPT_OFFSET_TRIM_LP_HIGH				OA2_OPT_OFFSET_TRIM_LP_LOW				OA1_OPT_OFFSET_TRIM_LP_HIGH				OA1_OPT_OFFSET_TRIM_LP_LOW							

レジスタ境界アドレステーブルについては、表 5 (47 ページ) を参照してください。

16 液晶ディスプレイコントローラ (LCD)

16.1 概要

この LCD コントローラはモノクロパッシブ液晶ディスプレイ (LCD) 用のデジタルコントローラ／ドライバであり、176 (44x4) または 320 (40x8) の LCD ピクチャエレメント (ピクセル) を駆動する最大 8 本のコモン端子と最大 44 本のセグメント端子を持っています。正確な端子数はデータシートに規定するデバイスのピン配置によって異なります。

LCD は、可視または不可視になる複数のセグメント (ピクセルまたは完全な記号) で構成されます。各セグメントは、2 つの電極間に配置さ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の両端に印加されると、セグメントは可視になります。セグメント電圧を交番させることで、液晶の表示に悪影響を与える電気泳動効果を避ける必要があります。さらに、直流 (DC) の発生を避けるために、セグメントにまたがる波形を生成する必要があります。

16.2 LCD の主な機能

- 柔軟なフレームレート制御。
- スタティック、1/2、1/3、1/4、1/8 デューティをサポート。
- スタティック、1/2、1/3、1/4 バイアスをサポート。
- 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から何時でも、表示データを損なうことなく更新可能。
 - － ピクセル情報（有効/無効）を含む最大 16 x 32 ビットレジスタから成る LCD データ RAM
- V_{LCDmin} から V_{LCDmax} の LCD 出力電圧（コントラスト）をソフトウェアから選択可能。
- 外部アナログコンポーネントが不要
 - － V_{DD} より高い内部 V_{LCD} 電圧を発生するステップアップコンバータを内蔵。
 - － 外部と内部の V_{LCD} 電圧源をソフトウェアから選択。外部電源の場合、内部ブースト回路を無効にして消費電力を削減。
 - － 中間 V_{LCD} 電圧 ($V_{LCDrail1}$ 、 $V_{LCDrail2}$ 、 $V_{LCDrail3}$) を発生する抵抗回路を内蔵。
 - － LCD パネルに必要な容量電荷に合わせて消費電力を適合させるために、抵抗回路網構成をソフトウェアから設定可能。
- 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する際、ソフトウェアによって V_{LCDmin} から V_{LCDmax} で V_{LCD} を調整可能。
 - － フレーム間のデッドタイム（最大 8 フェーズ周期）をプログラミング可能です。
- 低消費電力モードをフルサポート：LCD コントローラをスリープ、低消費電力動作、低消費電力スリープ、STOP モードで表示可能、または消費電力を削減するため完全に無効化可能。
- 低消費電力および EMI 用に位相反転を内蔵。（電磁干渉）。
- LCD データ RAM の更新時にソフトウェアを同期させるためのフレーム開始割り込み。
- ブリンク機能:
 - － 最大 1、2、3、4、8 または全ピクセルを設定可能な周波数でブリンク可能。
 - － 0.5 Hz、1 Hz、2 Hz または 4 Hz 付近のブリンク周波数を実現するためソフトウェアから調整可能。
- 使用する LCD のセグメントピンとコモンピンは GPIO オルタネート機能として設定し、未使用セグメントピン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として使用可能。
- V_{LCD} レール ($V_{LCDrail1}$ 、 $V_{LCDrail2}$ 、 $V_{LCDrail3}$) の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電源で動作する場合、 V_{LCD} ピンをコンデンサで V_{SS} へ接続する必要があります。代表値は 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）。

V_{LCD} ピンは次の場合 V_{DDA} に接続する必要があります。

- LCD なしのデバイスの場合

- LCD を持つデバイスで LCD ペリフェラルを使用しない場合

16.3 用語

バイアス : LCD の駆動時に使用される電圧レベル数。これは、 $1/(\text{LCD ディスプレイの駆動に使用される電圧レベル数} - 1)$ として定義されます。

ブースト回路 : 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 (44 セグメント) に接続される電氣的接続端子。

デューティ比 : $1/(\text{LCD ディスプレイのコモン端子数})$ で定義される数値。

フレーム : セグメントに書き込まれる 1 周期分の波形。

フレームレート : 1 秒あたりのフレーム数、すなわち、1 秒あたりに LCD セグメントが通電される回数。

LCD (液晶ディスプレイ) : セグメントに直接接続される端子を持つパッシブディスプレイパネル

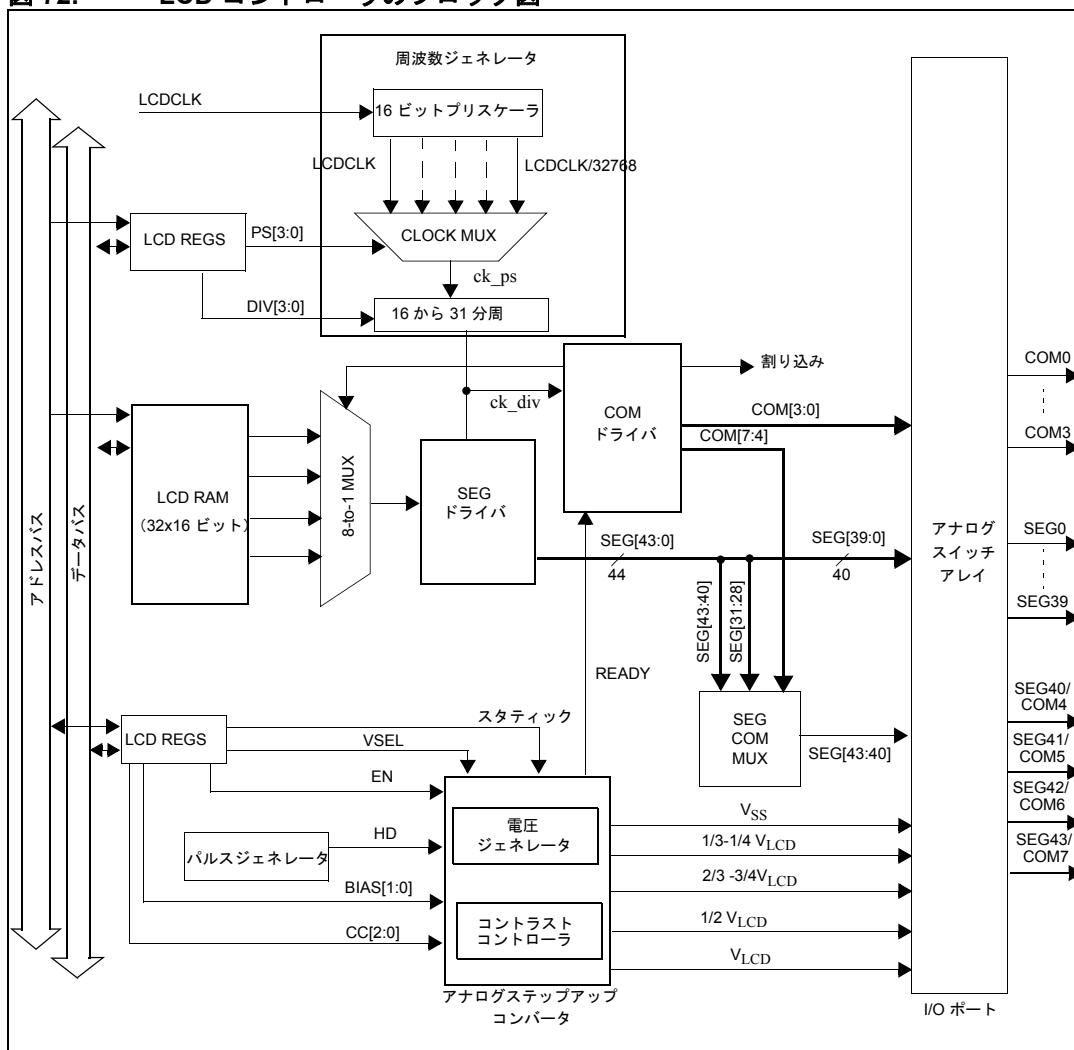
セグメント : 最小表示エレメント (LCD ディスプレイ上にキャラクタを作成するのに役立つ単一のバーまたはドット)。

16.4 LCD 機能の説明

16.4.1 概要

この LCD コントローラは 5 個のメインブロックを持っています (図 72 参照) :

図 72. 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。

16.4.2 周波数ジェネレータ

周波数ジェネレータは、32 kHz から 1 MHz の LCD 入力クロック周波数 (LCDCLK) から始まる様々な LCD フレームレートを実現することができます。

次の 3 種類のクロックソースを使って LCD クロック (LCDCLK/RTCCLK) を供給することができます:

- 32 kHz 低速外部 RC (LSE)
- 37 kHz 低速内部 RC (LSI)
- 1 から 24 MHz の高速外部水晶発振器 (HSE) を 2、4、8、16 分周して 1 MHz クロックを生成

このマニュアルの RCC セクションの RTC/LCD クロック 設定を参照してください。

正確な LCD タイミングを取得し、LCD セグメント間の DC 電圧オフセットを最小限に抑えるには、安定したクロックソースを選択する必要があります。入力クロック LCDCLK は、1 から $2^{15} \times 31$ ([セクション 16.5.2 : LCD フレーム制御レジスタ \(LCD_FCR\) \(387 ページ\)](#) 参照) の値で分周することができます。周波数ジェネレータは、プリスケアラ (16 ビットリップルカウンタ) とクロックの 16 から 31 分周器で構成されています。LCD_FCR レジスタの PS[3:0] ビットで、LCDCLK の $2^{PS[3:0]}$ 分周を選択します。より高い精度のレートが必要な場合は、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 から 31 分周することができます。この方法で周波数を粗調整した後、カウンタでクロックを微調整することができます。[周波数ジェネレータブロックの出力は \$f_{ck_div}\$ で、これが LCD コントローラ全体のタイムベースになります。ck_div 周波数は LCD 位相周波数に一致し、フレーム周波数とは異なります \(スタティックデューティの場合は一致\)。フレーム周波数 \(\$f_{frame}\$ \) は、 \$f_{ck_div}\$ をアクティブコモン端子数で除算して \(またはデューティを乗算して\) 得られます。したがって、周波数ジェネレータ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}\$ との関係は、次のようになります。](#)

$$f_{ckdiv} = \frac{f_{LCDCLK}}{2^{PS} \times \langle 16 + DIV \rangle}$$

$$f_{frame} = f_{ckdiv} \times duty$$

これにより周波数ジェネレータは非常に柔軟になります。フレームレート計算例を [表 73](#) に示します。

表 73. フレームレートの計算例

LCDCLK	PS[3:0]	DIV[3:0]	比	デューティ	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 の範囲で選択する必要があり、消費電力と許容リフレッシュレートとのトレードオフによります。さらに、専用の点滅プリスケアラで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。

$$f_{\text{BLINK}} = f_{\text{ck\_div}} / 2^{(\text{BLINKF} + 3)},$$

BLINKF[2:0] = 0, 1, 2, ..., 7

得られるブリンク周波数は、0.5 Hz、1 Hz、2 Hz、または 4 Hz の範囲内です。

16.4.3 コモンドライバ

コモン信号はコモンドライバブロックで生成されます (図 72 参照)。

COM 信号バイアス

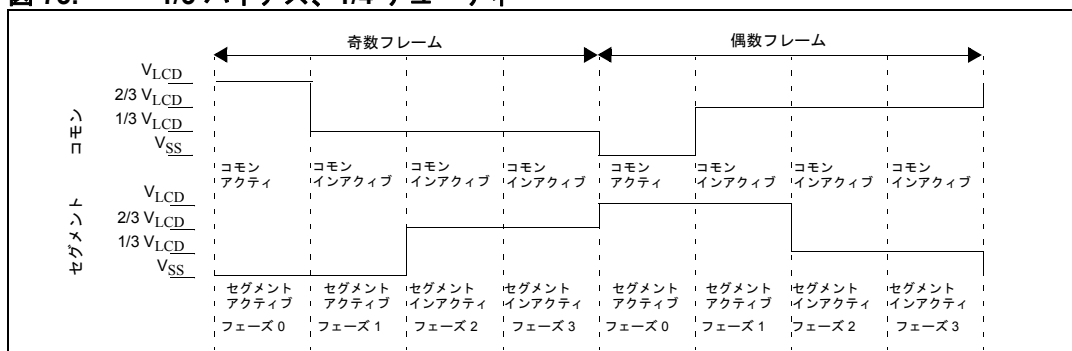
各 COM 信号の波形は同一ですが、位相が異なります。フレームサイクルの対応するフェーズでは V_{LCD} または V_{SS} の最大振幅になりますが、他のフェーズでの信号振幅は、次のようになります。

- 1/4 バイアスの場合 $1/4 V_{LCD}$ または $3/4 V_{LCD}$
- 1/3 バイアスの場合 $1/3 V_{LCD}$ または $2/3 V_{LCD}$
- 1/2 バイアスの場合 $1/2 V_{LCD}$ 。

LCD_CR レジスタの BIAS ビットを使って 1/2、1/3、1/4 バイアスモードを選択することができます。

ピクセルが有効になるのは、対応するコモンラインとセグメントラインの両方が同じフェーズ中に最大振幅を持つときです。EMI を削減するため、コモン信号の位相が反転されます。図 73 に示すように、位相反転では、各奇数サイクルの終わりに $1/2 V_{LCD}$ の平均電圧が生じます。

図 73. 1/3 バイアス、1/4 デューティ



1/2 バイアス (バイアス = 01) の場合、 V_{LCD} ピンは $V_{LCDrail2}$ に中間電圧を生成し、奇数フレームと偶数フレームに対して $1/2 V_{LCD}$ になります (図 80 参照)。

COM 信号のデューティ

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 75 参照)、1/2 デューティ (図 76 参照)、1/3 デューティ (図 77 参照)、1/4 デューティ (図 78) または 1/8 デューティ (図 79 参照) で生成されます。

COM[n] $n[0$ to 7] は、奇数フレームのフェーズ n でアクティブになるため、COM ピンは V_{LCD} に駆動されます。

偶数フレームのフェーズ 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:

- COM[n] は n 以外のフェーズでインアクティブになるため、COM ピンは奇数フレームでは $1/3$ ($1/4$) V_{LCD} に、偶数フレームでは $2/3$ ($3/4$) V_{LCD} に、それぞれ駆動されます。

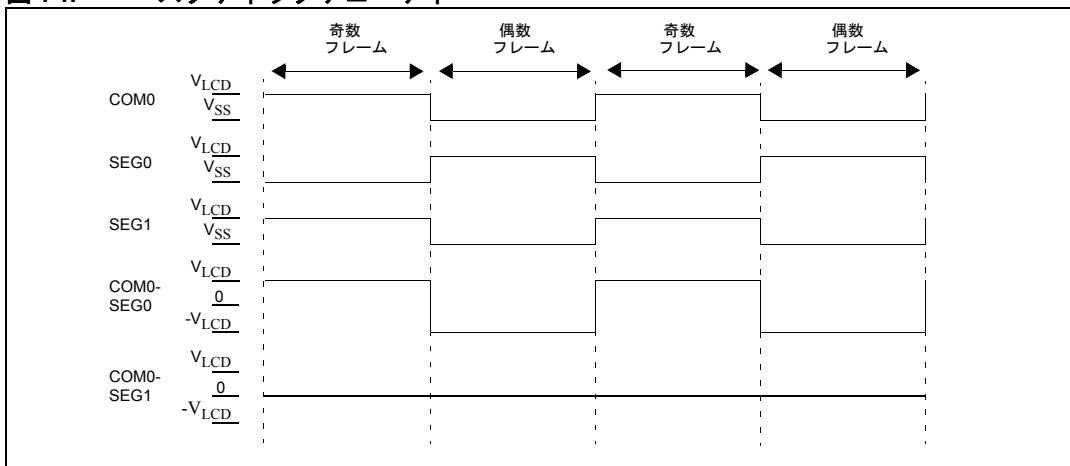
1/2 デューティの場合:

- COM[n] が n 以外のフェーズでインアクティブになると、COM ピンは常に (奇数フレームと偶数フレームで) $1/2 V_{LCD}$ 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マルチプレクスされないため、各セグメント出力は 1 ピクセルに対応します。この方法では 44 ピクセルまで駆動できます。COM[0] は常にアクティブで、COM[7:1] は使用されないため V_{SS} に駆動されます。

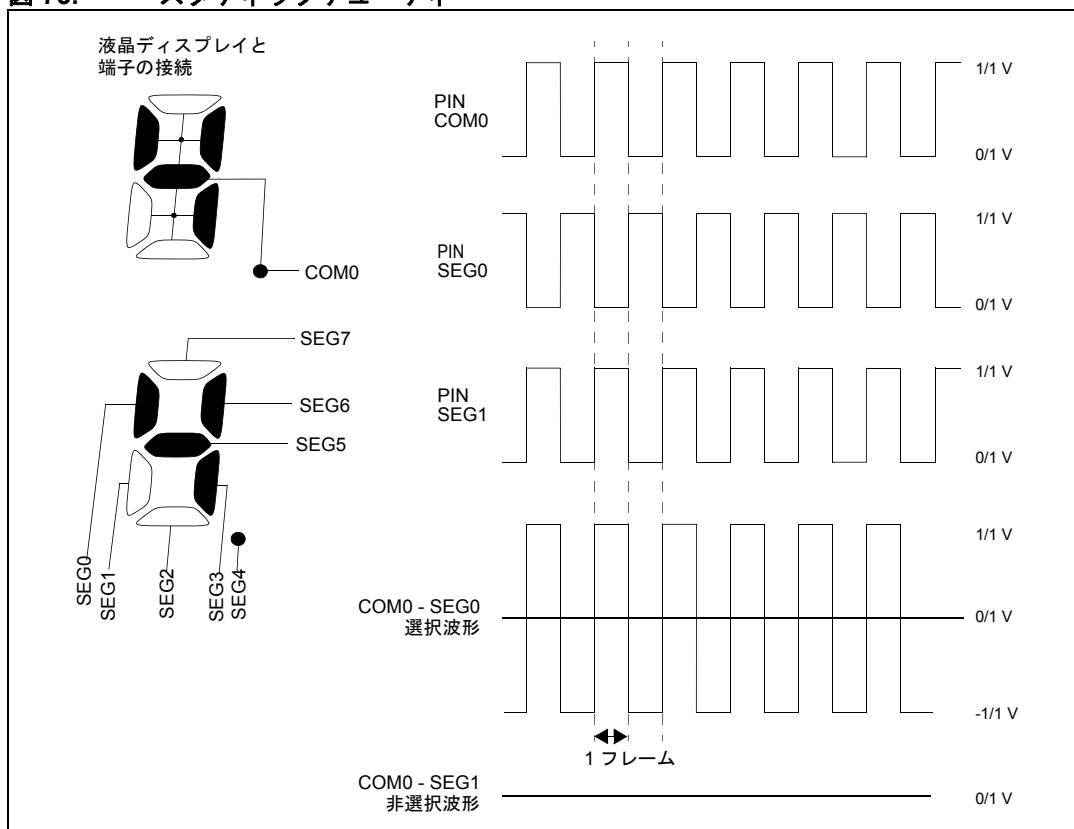
LCD_CR レジスタの LCDEN ビットがリセットされると、すべてのコモンラインが V_{SS} にプルダウンされ、LCD_SR レジスタの ENS フラグが 0 になります。スタティックデューティとは、COM[0] が常にアクティブで、セグメントラインとコモンラインに対して V_{LCD} と V_{SS} の 2 つの電圧レベルのみを使用することを意味します。対応する SEG ラインが COM と反対の電圧を持つとき、ピクセルはアクティブになり、等しい電圧を持つときインアクティブになります。この場合、LCD のコントラストは最大になります (図 74 と 図 75 を参照)。図 74 では、ピクセル 0 がアクティブで、ピクセル 1 はインアクティブです。

図 74. スタティックデューティ



各フレームには 1 つのフェーズしかないので、 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 つのフェーズがあり、フェーズ 0 では COM[0] が有効、フェーズ 1 では COM[1] が有効、フェーズ 2 では COM[2] が有効、フェーズ 3 では COM[3] が有効です。

図 75. スタティックデューティ



このモードでは、セグメント端子がマルチプレクスされ、それぞれが4つのピクセルを制御します。ピクセルが有効になるのは、その対応する SEG ラインと COM ラインの両方が同フェーズで有効な場合のみです。1/4 デューティの場合、COM[0] に接続されたピクセル 0 をインアクティブにするためには、COM[0] がアクティブのときフェーズ 0 で SEG[0] はインアクティブである必要があります。COM[1] に接続されたピクセル 44 をアクティブにするためには、COM[1] がアクティブのときフェーズ 1 で SEG[0] はアクティブである必要があります（[図 78](#) 参照）。COM[0] に接続されたピクセル 0 から 43 をアクティブにするためには、COM[0] がアクティブのときフェーズ 0 で SEG[0:43] はアクティブである必要があります。これらの考え方は、他のピクセルにも適用できます。

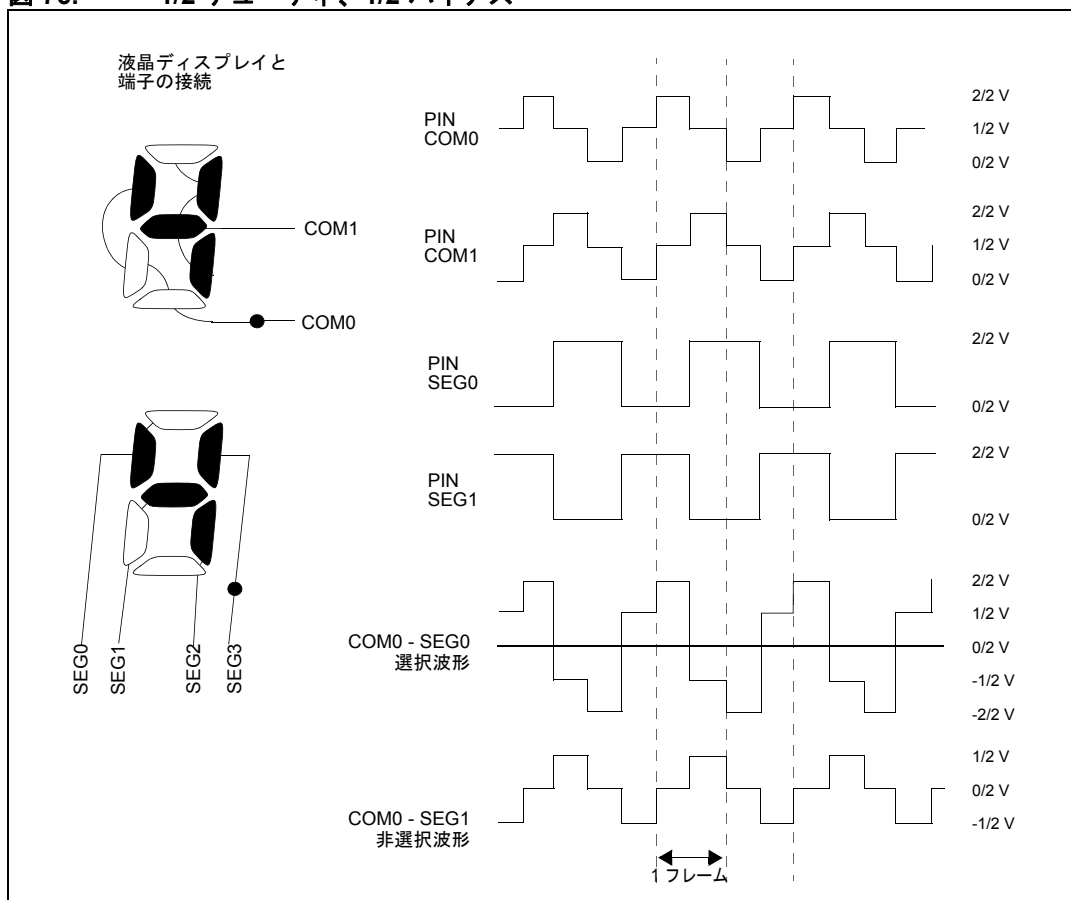
8→1 マルチプレクサ

COM[0] がアクティブなコモンドライバブロックである場合、[図 72](#) に示す 8→1 マルチプレクサも駆動して、最初の 2 つの RAM レジスタロケーションの値を選択するようにします。COM[7] がアクティブのとき、8→1 マルチプレクサ出力は最後の 2 つの RAM ロケーションの値になります。

フレーム開始 (SOF)

コモンドライバブロックも SOF (フレーム開始フラグ) を生成することができます（[セクション 16.5.3: LCD ステータスレジスタ \(LCD_SR\)](#) 参照）。SOFIE (フレーム開始割り込みイネーブル) 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 16.5.2: LCD フレーム制御レジスタ \(LCD_FCR\)](#) 参照）。対応する割り込み処理ベクタの実行時に LCD_CLR レジスタの SOFC ビットに 1 を書き込むと SOF がクリアされます。

図 76. 1/2 デューティ、1/2 バイアス



16.4.4 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から各フェーズで駆動される 8→1 マルチプレクサ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] がアクティブのとき、COM[0] に接続されたピクセルに関するピクセル情報（アクティブ/インアクティブ）は 8→1 マルチプレクサを通過します（最初の 2 つの LCD_RAM ロケーションの値）。

SEG[n] ピン n [0 から 43] は、奇数フレームのフェーズ 0 で V_{SS} に駆動されます（COM[0] がアクティブのときピクセル n がアクティブであることを示します）。

SEG[n] ピンは、偶数フレームのフェーズ 0 で V_{LCD} に駆動されます。ピクセル n がインアクティブの場合、SEG[n] ピンは奇数フレームで $2/3$ ($2/4$) V_{LCD} に、偶数フレームで $1/3$ ($2/4$) V_{LCD} に (V_{LCD} パッドで電流反転)、それぞれ駆動されます（図 73 参照）。

1/2 バイアスの場合、ピクセルがインアクティブのとき、SEG[n] ピンは奇数フレームで V_{LCD} に、偶数フレームで V_{SS} に、それぞれ駆動されます。

LCD コントローラが無効化された場合（LCD_CR レジスタで LCDEN ビットをクリア）、SEG ラインは V_{SS} にプルダウンされます。

図 77. 1/3 デューティ、1/3 バイアス

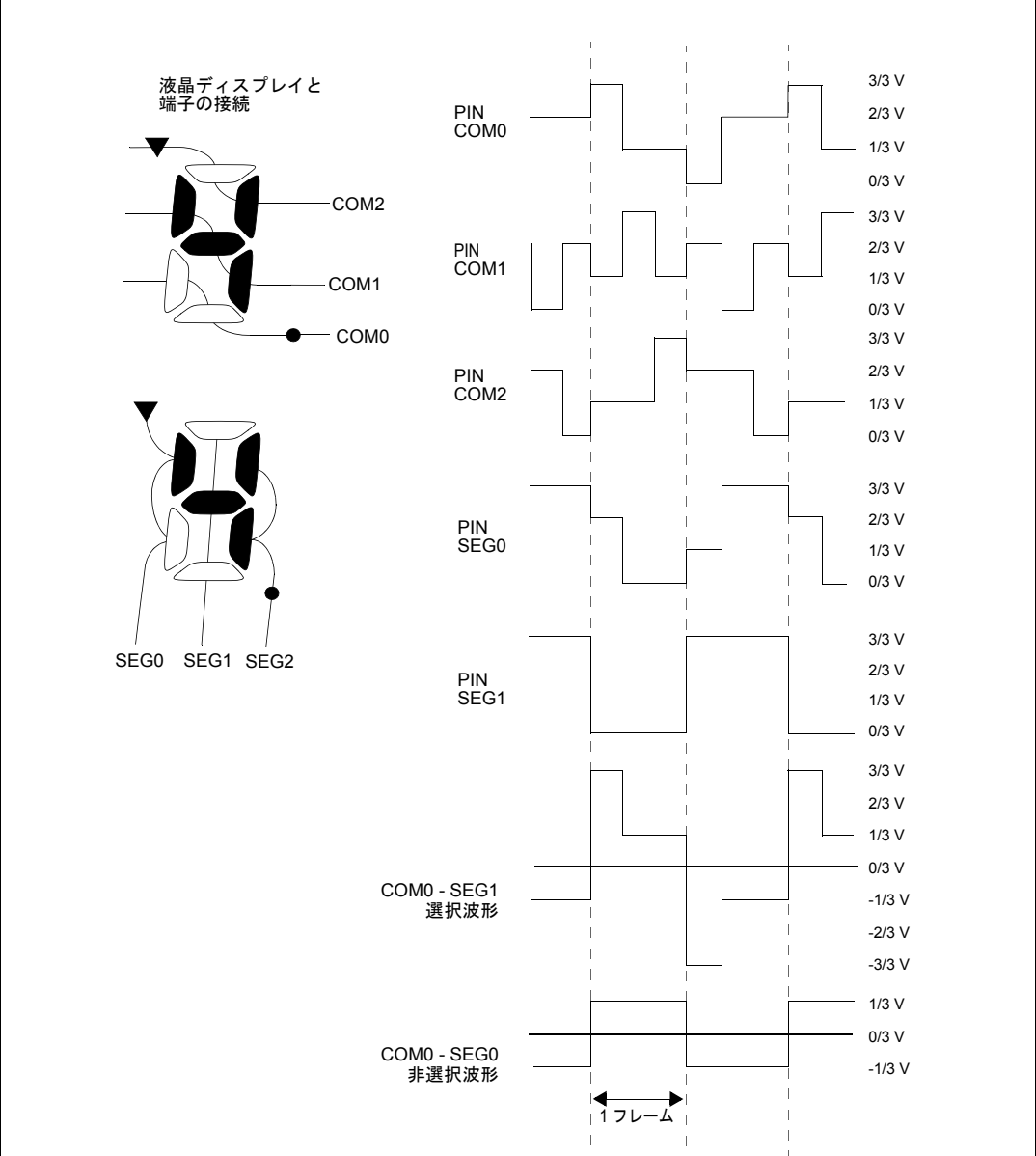


図 78. $1/4$ デューティ、 $1/3$ バイアス

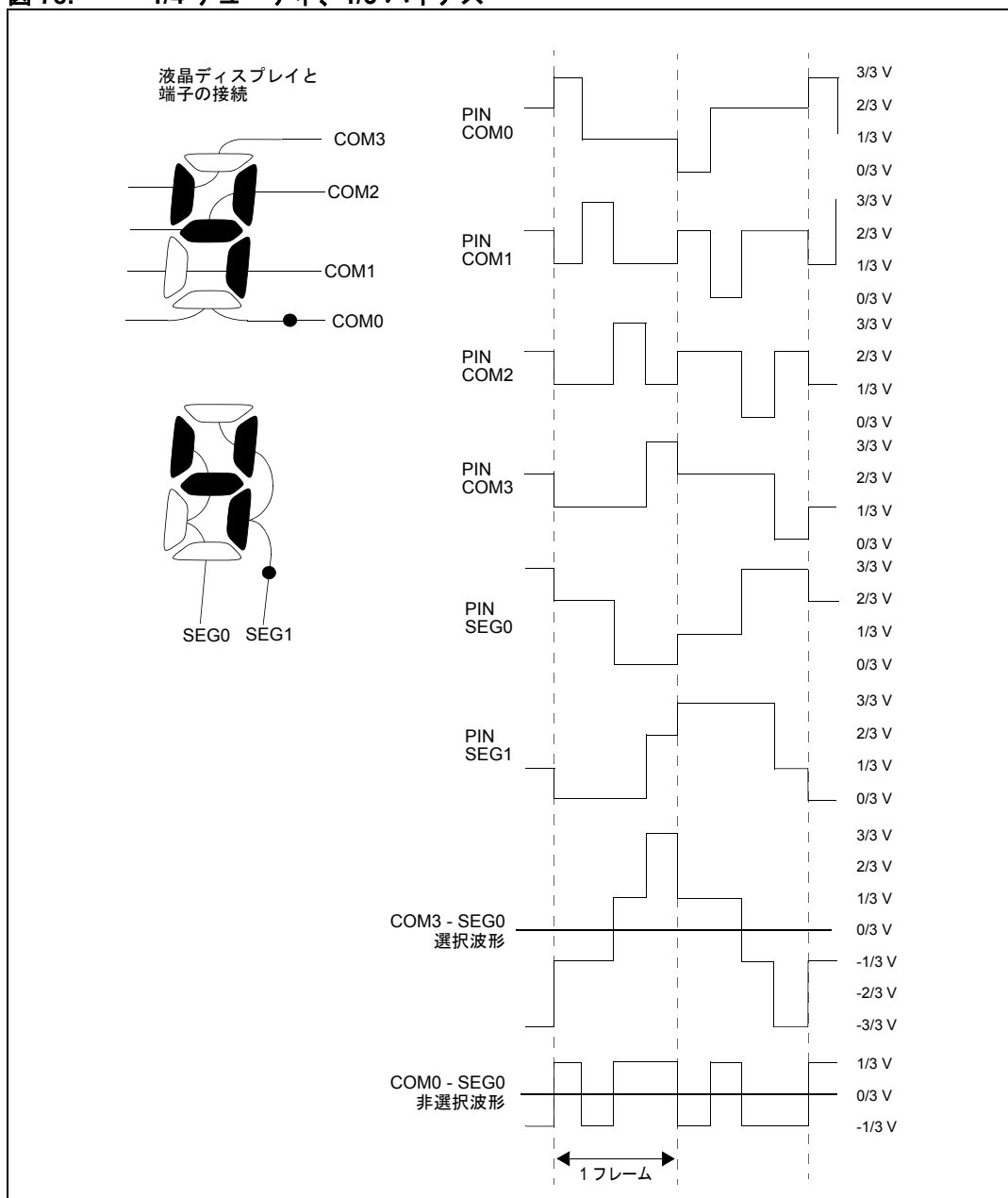
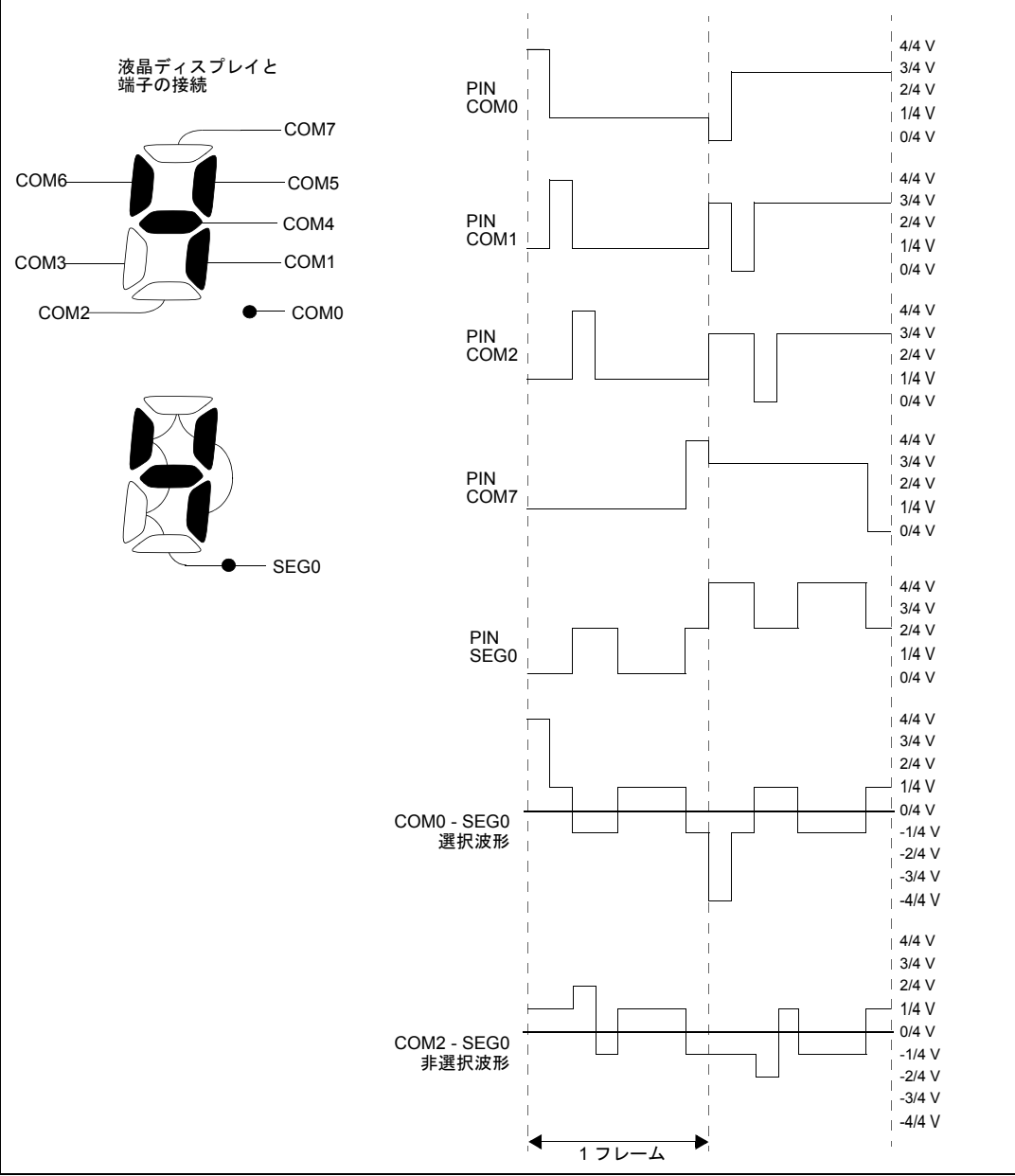


図 79. 1/8 デューティ、1/4 バイアス



点滅

セグメントドライバはプログラマブルなブリンク機能も持っているため、複数のピクセルを連続的に指定周波数でオン/オフさせることができます。ブリンクモードは LCD_FCR レジスタの BLINK[1:0] ビットで設定することができ、1、2、4、8 または全ピクセルをブリンクさせることができます（[セクション 16.5.2 : LCD フレーム制御レジスタ \(LCD_FCR\)](#) 参照）。ブリンク周波数は、LCD_FCR レジスタの BLINKF[2:0] ビットを使って 8 種類の値を選択することができます。

[表 74](#) に、ブリンク周波数（ck_div 周波数の関数）の例を示します。

表 74. ブリンク周波数

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

16.4.5 電圧ジェネレータ

LCD 電圧レベルは、[図 80](#) に示す内部抵抗分圧回路を使って、 V_{LCD} ピンまたは内部電圧ステップアップコンバータにより生成されます（LCD_CR レジスタの VSEL ビットで指定）。

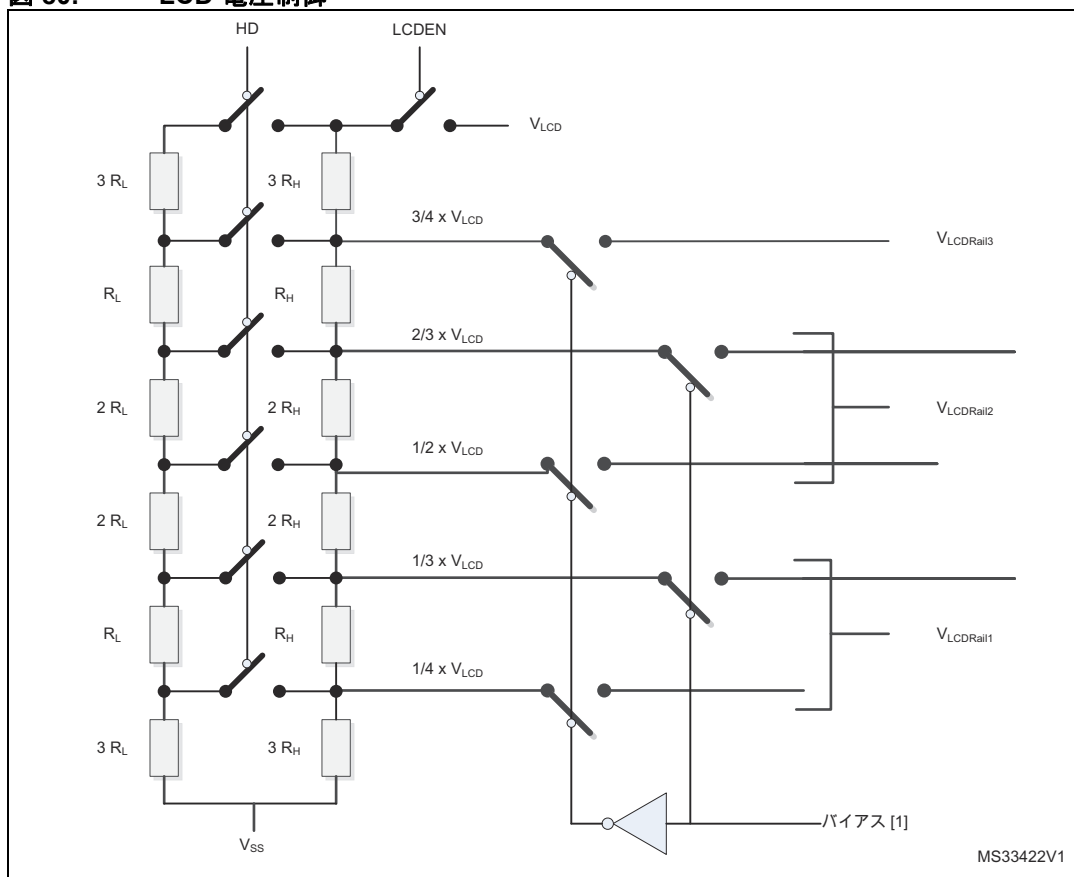
LCD 電圧ジェネレータは、 V_{SS} から V_{LCD} の中間電圧レベルを生成します：

- 1/3 バイアスの場合、 $1/3 V_{LCD}$ 、 $2/3 V_{LCD}$
- 1/4 バイアスの場合、 $1/4 V_{LCD}$ 、 $2/4 V_{LCD}$ 、 $3/4 V_{LCD}$
- 1/2 バイアスの場合、 $1/2 V_{LCD}$ 。

分圧回路に対して 2 本の抵抗を使用して（1 本は低値の抵抗 (R_L)、もう 1 本は高値の抵抗 (R_H) をそれぞれ使用）、変化時に電流を大きくし、スタティック状態で消費電力を削減しています。

LCD_FCR レジスタの PON[2:0]（パルスオン継続時間）ビットで、コモンラインとセグメントラインのレベルが変化する際に有効化される R_L の時間を設定します（[図 72](#) 参照）。駆動時間を短くすると消費電力は減少しますが、内部抵抗が高いディスプレイでは、十分なコントラストを得るために長い駆動時間を必要とすることがあります。

図 80. LCD 電圧制御



LCD_FCR 設定レジスタの HD ビットを使って、 R_L 分圧器を常にオンにします（[セクション 16.5.2](#) 参照）。 V_{LCD} 値は、LCD_FCR（[セクション 16.5.2](#) 参照）レジスタの CC[2:0]（コントラスト制御）ビットを使って、 V_{LCDmin} から V_{LCDmax} の広い範囲から選択することができます。 V_{LCD} の新しい値は、新しいフレームの開始ごとに有効になります。

LCDEN ビットがアクティブになると、電圧ジェネレータは LCD_SR レジスタの RDY ビットをセットして、電圧レベルが安定し、LCD コントローラが動作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参照）では、 V_{LCD} 中間電圧レール（ $V_{LCDrail1}$ 、 $V_{LCDrail2}$ 、 $V_{LCDrail3}$ - [図 80](#) 参照）にデカップリングコンデンサを接続して、さらに安定化させることができる場合があります。ピクセルに加えられる電圧が変化する際に、スパイクが発生することがあります。この場合、これらのデカップリングコンデンサが電圧安定化に役立ち、コントラストが高くなります。この機能は特に、LCD_FCR レジスタで小さい PON[2:0] 値の選択が可能になるため省消費電力に役立ちます。

V_{lcd} レールを [表 75](#) の説明のように専用 GPIO に接続するときは、SYSCFG_PMC レジスタの LCD_CAPA[4:0] ビットを設定します。

表 75. GPIO ピン へのV_{LCDrail} の接続

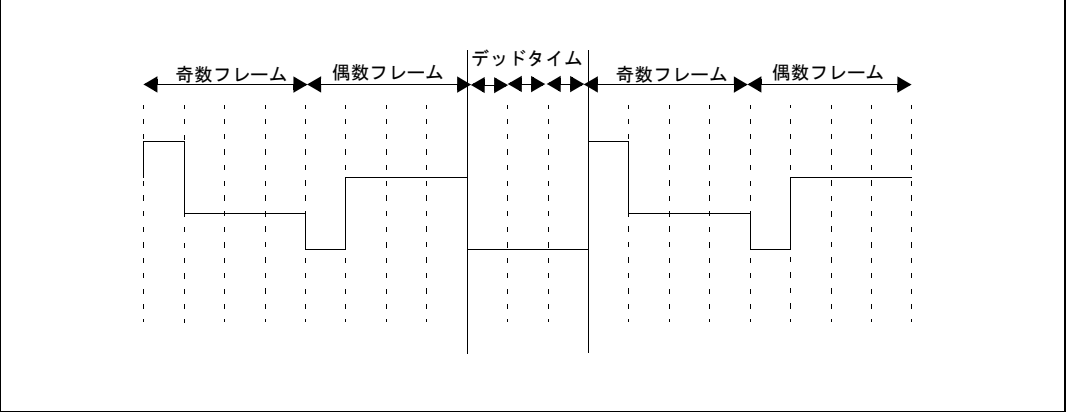
	バイアス			ピン (LCD_CAPA[4:0] ビットで選択)
	1/2	1/3	1/4	
V _{LCDrail3}	未使用	未使用	3/4 V _{lcd}	PB0 または PE12
V _{LCDrail2}	1/2 V _{lcd}	2/3 V _{lcd}	1/2 V _{lcd}	PB2
V _{LCDrail1}	未使用	1/3 V _{lcd}	1/4 V _{lcd}	PB12 または PE11

効果的にするためには、LCD ガラスと PCB 容量に従ってこれらのデカップリングコンデンサの値を調整する必要があります。ガイドラインとして、デカップリングコンデンサ値を LCD 容量の 10 倍することができます。

16.4.6 デッドタイム

CC[2:0] ビットの使用の他に、各フレーム間のデッドタイムを設定してコントラストを制御することができます。デッドタイムの間、COM 値と SEG 値は V_{SS} になります。LCD_FCR レジスタの DEAD[2:0] ビットを使って、8 フェーズ期間までの時間を設定することができます。このデッドタイムによって、フレームレートを変えることなく、コントラストが小さくなります。

図 81. デッドタイム



16.4.7 ダブルバッファメモリ

ダブルバッファメモリの使用により、LCD コントローラは、LCD_RAM の変更を制御するために割り込みを使わないでディスプレイ情報の整合性を保証します。

アプリケーションソフトウェアは、APB インタフェースを使って先頭バッファレベル (LCD_RAM) にアクセスすることができます。LCD_RAM を変更したら、LCD_SR レジスタの UDR フラグをセットします。この UDR フラグ (ディスプレイ更新リクエスト) は、更新された情報を 2 番目のバッファレベル (LCD_DISPLAY) へ移動することを要求します。

この動作はフレーム (次のフレームの開始) に同期して行われ、更新が完了するまで、LCD_RAM は書き込み保護され、UDR フラグはハイレベルを維持します。更新が完了すると、別のフラグ (UDD - ディスプレイ更新完了) 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、割り込みが生成されます。

LCD_DISPLAY の更新に要する時間は、最悪ケースで、1 奇数および 1 偶数フレームです。

ディスプレイが有効化されるまで (LCDEN = 1)、更新は行われません (UDR = 1 および UDD = 0)

16.4.8 COM と SEG のマルチプレクシング

出力ピン対デューティモード

出力ピンは次で構成されています。

- SEG[43:0]
- COM[3:0]

デューティの設定に応じて、COM 出力ピンと SEG 出力ピンは異なる次の機能を持つことができます。

- スタティックでは、1/2、1/3、1/4 デューティモード、最大 44 SEG ピンあり、それぞれ 1、2、3、4 COM ピンになります。
- 1/8 デューティモード (DUTY[2:0] = 100) では、SEG[43:40] ピンが COM[7:4] 出力になり、使用可能セグメント数が 40 に減少します。

再配置機能

さらに、LCD_CR レジスタの MUX_SEG ビットをセットして、4 セグメントを再配置することができます。これは、特に、外部ピンが少ない小型デバイスタイプを使う際に役立ちます。

MUX_SEG をセットすると、出力ピン SEG[43:40] は SEG[31:28] の機能を持ちます。

COM 機能および SEG 機能対デューティおよび再配置のまとめ

COM 機能と SEG 機能をマルチプレクスするすべての方法を表 76 に示します。図 82 に、外部ピンへ接続される信号の例を示します。

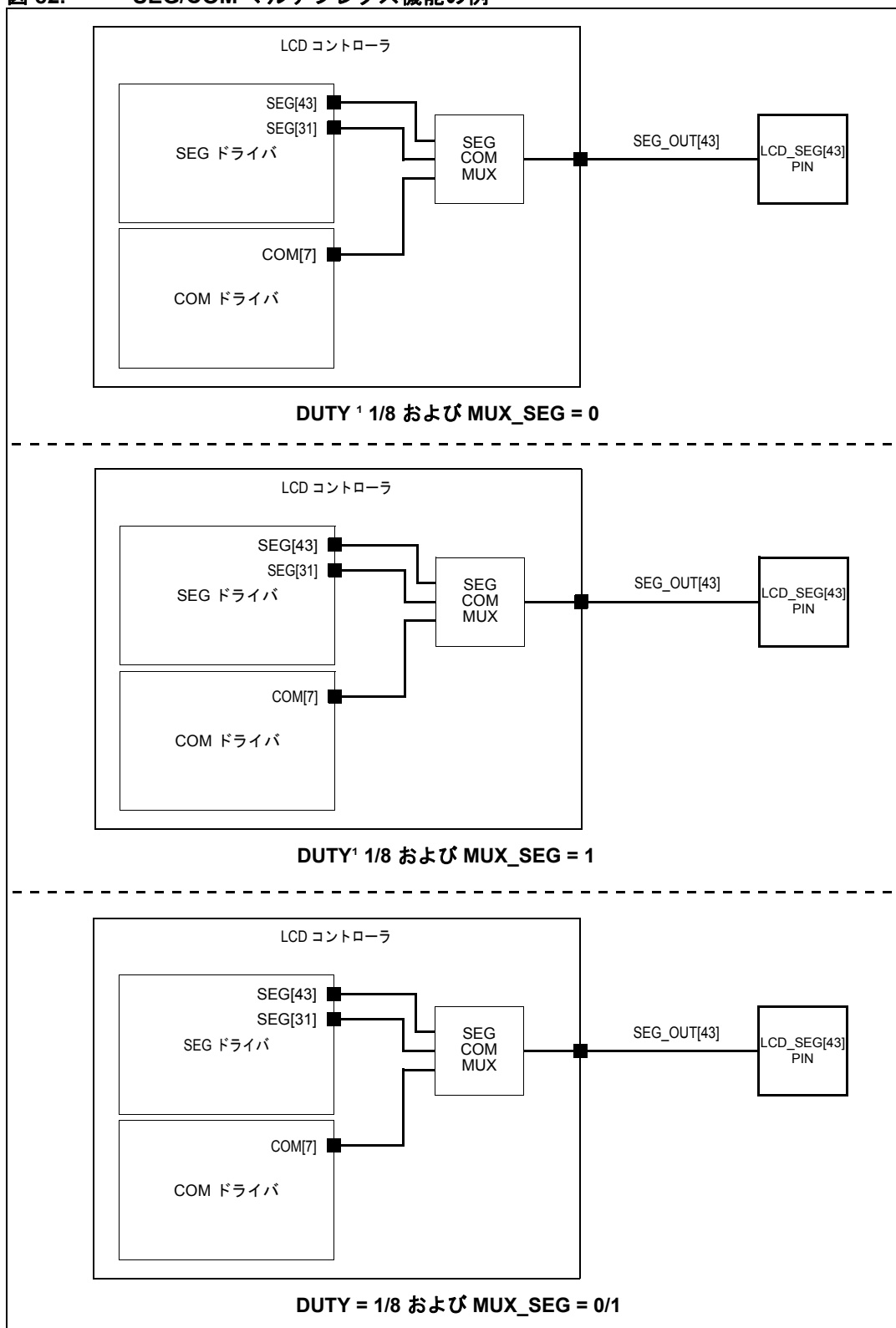
表 76. 再配置機能

設定ビット		機能	出力ピン	機能
DUTY	MUX_SEG			
1/8	0	40x8	SEG[43:40]	COM[7:4]
			COM[3:0]	COM[3:0]
			SEG[39:0]	SEG[39:0]
	1	28x8	SEG[43:40]	COM[7:4]
			COM[3:0]	COM[3:0]
			SEG[39:28]	未使用
			SEG[27:0]	SEG[27:0]
1/4	0	44x4	COM[3:0]	COM[3:0]
			SEG[43:0]	SEG[43:0]
	1	32x4	COM[3:0]	COM[3:0]
			SEG[43:40]	SEG[31:28]
			SEG[39:28]	未使用
			SEG[27:0]	SEG[27:0]
1/3	0	44x3	COM[3]	未使用
			COM[2:0]	COM[2:0]
			SEG[43:0]	SEG[43:0]
	1	32x3	COM[3]	未使用
			COM[2:0]	COM[2:0]
			SEG[43:40]	SEG[31:28]
			SEG[39:28]	未使用
			SEG[27:0]	SEG[27:0]
1/2	0	44x2	COM[3:2]	未使用
			COM[1:0]	COM[1:0]
			SEG[43:0]	SEG[43:0]
	1	32x2	COM[3:2]	未使用
			COM[1:0]	COM[1:0]
			SEG[43:40]	SEG[31:28]
			SEG[39:28]	未使用
			SEG[27:0]	SEG[27:0]

表 76. 再配置機能 (続き)

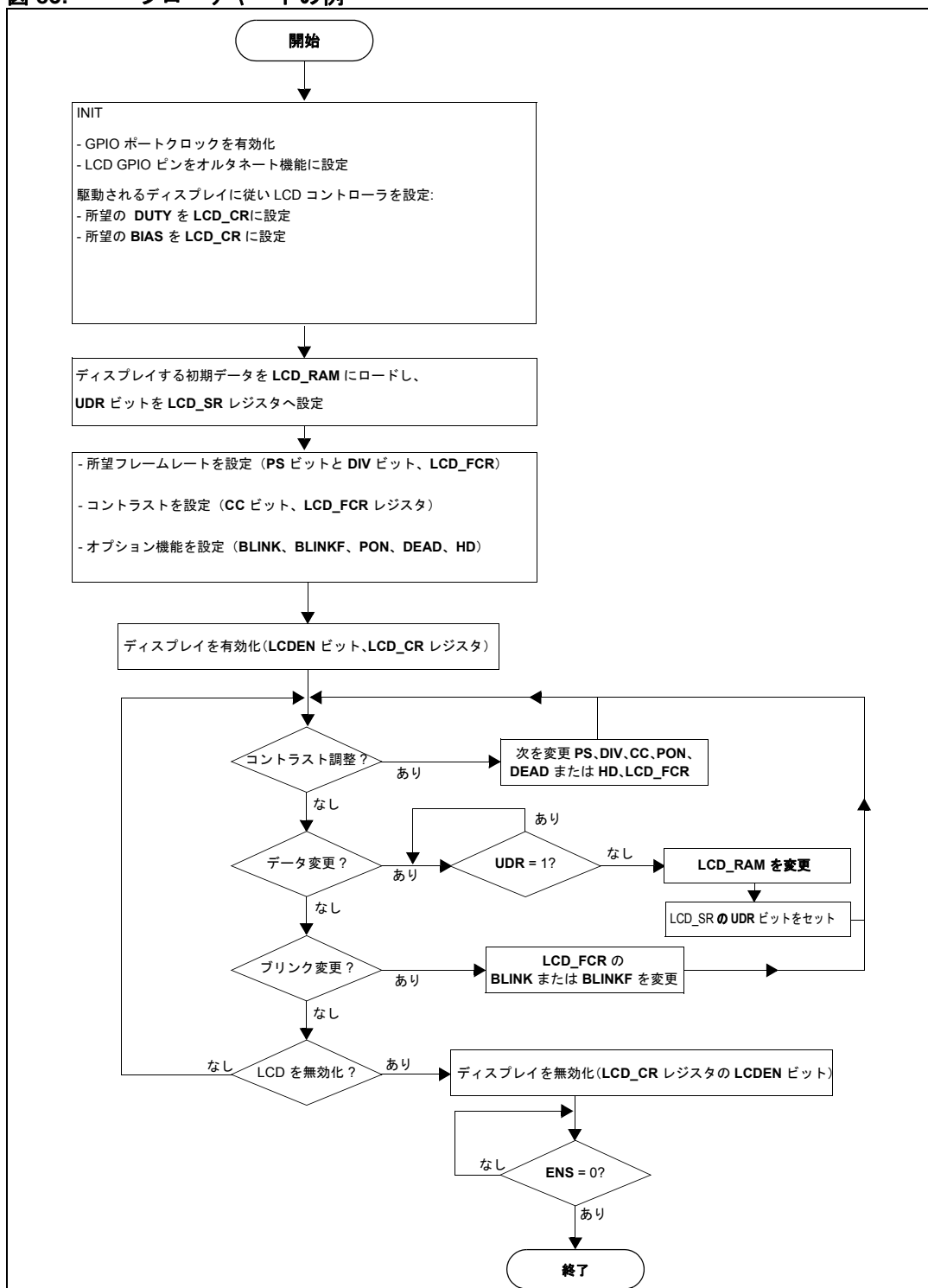
設定ビット		機能	出力ピン	機能
DUTY	MUX_SEG			
スタティック	0	44x1	COM[3:1]	未使用
			COM[0]	COM[0]
			SEG[43:0]	SEG[43:0]
	1	32x1	COM[3:1]	未使用
			COM[0]	COM[0]
			SEG[43:40]	SEG[31:28]
			SEG[39:28]	未使用
			SEG[27:0]	SEG[27:0]

図 82. SEG/COM マルチプレクス機能の例



16.4.9 フローチャート

図 83. フローチャートの例



16.5 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。

16.5.1 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								MUX_SEG	BIAS[1:0]		DUTY[2:0]			VSEL	LCDEN
								rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	rw/r	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 **MUX_SEG** : マルチプレクサセグメントイネーブル
 このビットを使って SEG ピン再配置を有効化します。4 SEG ピンを SEG[31:28] でマルチプレクスすることができます。 [セクション 16.4.8](#) 参照。
 0 : SEG ピンマルチプレクスを無効化
 1 : SEG[31:28] を SEG[43:40] でマルチプレクス

ビット 6:5 **BIAS[1:0]** : バイアスセクタ
 これらのビットで使用するバイアスを決定。値 11 は禁止。
 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
 これらのビットでデューティサイクルを決定。値 101、110、111 は禁止。
 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** : 電圧源選択
 VSEL ビットで LCD 電圧源を指定。
 0 : 内部電源 (電圧ステップアップコンバータ)
 1 : 外部電源 (V_{LCD} ピン)



ビット 0 **LCDEN** : LCD コントローラインープル

ソフトウェアからこのビットをセットして LCD コントローラ／ドライバを有効化します。ソフトウェアからクリアして、次のフレームの開始で LCD をオフにします。LCD を無効化すると、すべての COM ピンと SEG ピンは V_{SS} に駆動されます。

0 : LCD コントローラを無効化

1 : LCD コントローラを有効化

注 : **LCD を有効化すると (LCD_SR の ENS ビット = 1)、VSEL、MUX_SEG、BIAS、DUTY ビットは書き込み保護されます。**

16.5.2 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
予約済み						PS[3:0]				DIV[3:0]				BLINK[1:0]	
						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
BLINKF[2:0]			CC[2:0]			DEAD[2:0]			PON[2:0]			UDDIE	予約済み	SOFIE	HD
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

ビット 31:26 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 25:22 **PS[3:0]** : PS 16 ビットプリスケアラ

ソフトウェアからこれらのビットを書き込んで、PS 16 ビットプリスケアラの分周比を指定します。

ck_ps = LCDCLK / (2)。 [セクション 16.4.2](#) 参照。

0000 : ck_ps = LCDCLK

0001 : ck_ps = LCDCLK/2

0011 : ck_ps = LCDCLK/4

...

1111 : ck_ps = LCDCLK/32768

ビット 21:18 **DIV[3:0]** : DIV クロック分周比

ソフトウェアからこれらのビットを書き込んで、DIV 分周器の分周比を指定します。 [セクション 16.4.2](#) 参照。

0000 : ck_div = ck_ps/16

0001 : ck_div = ck_ps/17

0011 : ck_div = ck_ps/18

...

1111 : ck_div = ck_ps/31

ビット 17:16 **BLINK[1:0]** : ブリンクモード選択

00 : ブリンクを無効化

01 : SEG[0]、COM[0] でブリンクを有効化 (1 ピクセル)

10 : SEG[0]、全 COM でブリンクを有効化 (設定したデューティに応じて最大 8 ピクセル)

11 : 全 SEG と全 COM でブリンクを有効化 (全ピクセル)



ビット 15:13 **BLINKF[2:0]** : ブリンク周波数選択

000 : $f_{LCD}/8100$: $f_{LCD}/128$
 001 : $f_{LCD}/16101$: $f_{LCD}/256$
 010 : $f_{LCD}/32110$: $f_{LCD}/512$
 011 : $f_{LCD}/64111$: $f_{LCD}/1024$

ビット 12:10 **CC[2:0]** : コントラスト制御

これらのビットで、 V_{LCD} 最大電圧 (V_{DD} に無関係) を指定。2.60 V から 3.51V の範囲。

000 : $V_{LCD0}100$: V_{LCD4}
 001 : $V_{LCD1}101$: V_{LCD5}
 010 : $V_{LCD2}110$: V_{LCD6}
 011 : $V_{LCD3}111$: V_{LCD7}

注 : 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 継続時間

ソフトウェアからこれらのビットを書き込んで、フレーム間のデッドタイム長を設定します。デッドタイムの間、COM と SEG の電圧レベルが 0 V に維持されて、フレームレートを変更することなく、コントラストを小さくします。

000 : デッドタイムなし
 001 : 1 フェーズ周期のデッドタイム
 010 : 2 フェーズ周期のデッドタイム

 111 : 7 フェーズ周期のデッドタイム

ビット 6:4 **PON[2:0]** : パルスオン継続時間

ソフトウェアからこれらのビットを書き込んで、パルス継続時間を ck_ps パルス単位で指定します。この区間低抵抗分圧器が有効化されます。パルスを短くすると、低消費電力になりますが、高内部抵抗のディスプレイの場合、十分なコントラストを実現するためには長いパルスが必要になります。

パルスは、プリスケール済み LCD クロック周期の 1/2 より長くすることはできないことに注意してください。

000 : 0 100 : $4/ck\_ps$
 001 : $1/ck\_ps101$: $5/ck\_ps$
 010 : $2/ck\_ps110$: $6/ck\_ps$
 011 : $3/ck\_ps111$: $7/ck\_ps$

PON 継続時間の例、LCDCLK = 32.768 kHz かつ PS=0x03:

000 : 0 μs 100 : 976 μs
 001 : 244 μs 101 : 1.22 ms
 010 : 488 μs 110 : 1.46 ms
 011 : 782 μs 111 : 1.71 ms

ビット 3 **UDDIE** : ディスプレイ更新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/クリアします。

0 : LCD ディスプレイ更新完了割り込みを無効化
 1 : LCD ディスプレイ更新完了割り込みを有効化

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 **SOFIE** : フレーム開始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/クリアします。

0 : LCD フレーム開始割り込みを無効化

1 : LCD フレーム開始割り込みを有効化

ビット 0 **HD** : 高駆動イネーブル

ソフトウェアからこのビットを書き込んで低抵抗分圧器を有効化します。高内部抵抗のディスプレイの場合、十分なコントラストを実現するためには強い駆動が必要になります。消費電力の増加が許容できる場合、このビットが役立ちます。

0 : 高駆動を無効化

1 : 高駆動を有効化HD=1 の場合、PON ビットを 000 以外の値に設定する必要があります。

注 : このレジスタのデータは何時でも更新できますが、新しい値は次のフレームの開始でのみ適用されます (デバイス動作に直ちに影響を与える CC、UDDIE、SOFIE は除きます)。

このレジスタを読み出すと、レジスタに書き込まれた直前の値が得られ、現在のフレームのディスプレイに使用された設定とは異なります。

16.5.3 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み										FCRSF	RDY	UDD	UDR	SOF	ENS
										r	r	r	rs	r	r

ビット 31:6 予約済みであり、リセット値に保持する必要があります。

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ごとに、ハードウェアによってセットされます。LCD_FCR レジスタに書き込みを行うと、ハードウェアからクリアされます。

0 : LCD フレーム制御レジスタ未同期

1 : LCD フレーム制御レジスタ同期済み

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表示します。

0 : レディでない状態

1 : ステップアップコンバータが有効化され、正しい電圧供給が可能

ビット 3 **UDD** : ディスプレイ更新完了

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 を書き込むとクリアされます。ビットのセットはクリアより優先されます。

0 : イベントなし

1 : ディスプレイ更新リクエストが完了LCD_FCR レジスタの UDDIE ビットがセットされている場合、UDD 割り込みが発生します。

注 : デバイスが STOP モードの場合 (PCLK がない場合)、UDDIE = 1 であっても、UDD は割り込みを生成しません。

ディスプレイを有効化しない場合、UDD 割り込みは生成されません。



- ビット 2 **UDR** : ディスプレイ更新リクエスト
- ソフトウェアから LCD_RAM を変更するごとに、UDR ビットをセットして、更新したデータを 2 番目のレベルバッファへ転送する必要があります。UDR ビットは更新が終わるまでセット状態を維持し、この間 LCD_RAM は書き込み保護されます。
- 0 : 影響なし。
- 1 : ディスプレイ更新リクエスト
- 注 :** ディスプレイを無効化すると、全 LCD_DISPLAY ロケーションに更新が行われます。ディスプレイを有効化すると、更新はコモンがアクティブになっているロケーションにのみ行われます (DUTY に依存)。例えば DUTY = 1/2 の場合、COM0 と COM1 の LCD_DISPLAY のみが更新されます。
- 注 :** このビットへの 0 の書き込み、またはこのビットが 1 であるときの 1 の書き込みは無視されます。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 の場合にのみクリアされます。
- ビット 1 **SOF** : フレーム開始フラグ
- このビットは、新しいフレームの開始でハードウェアによってセットされ、ディスプレイデータと同時に更新されます。LCD_CLR レジスタの SOFC ビットに 1 を書き込むと、クリアされます。ビットのクリアがセットより優先されます。
- 0 : イベントなし
- 1 : フレーム開始イベントが発生 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。
- ビット 0 **ENS** : LCD イネーブルステータス
- 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表示します。
- 0 : LCD コントローラを無効化。
- 1 : LCD コントローラを有効化
- 注 :** LCD_CR の LCDEN ビットが 0 から 1 へ変化すると直ちに ENS ビットがセットされます。インアクティブになると、LCD の実際のステータスを反映するため、最後のディスプレイフレームの終わりに 0 になります。

16.5.4 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み												UDDC		SOFC	
												w	予約済み	w	予約済み

- 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 3 **UDDC** : ディスプレイ更新完了クリア
- ソフトウェアからこのビットを書き込んで、LCD_SR レジスタの UDD フラグをクリアします。
- 0 : 影響なし。
- 1 : UDD フラグをクリア



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 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 **SOFC** : フレーム開始フラグクリア
ソフトウェアからこのビットを書き込んで、LCD_SR レジスタの SOF フラグをクリアします。
0 : 影響なし。
1 : SOF フラグをクリア
- ビット 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。

16.5.5 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、液晶ディスプレイの 1 ピクセルに対応します。
0 : ピクセル無効
1 : ピクセル有効

16.5.6 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 77. 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	予約済み																								MUX_SEG	BIAS[1:0]		DUTY[2:0]			VSEL	LCDEN
	リセット値																									0	0	0	0	0	0	0	0
0x04	LCD_FCR	予約済み				PS[3:0]				DIV[3:0]				BLINK[1:0]		BLINKF[2:0]				CC[2:0]				DEAD[2:0]		PON[2:0]			UDDIE	予約済み	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	0	0
0x08	LCD_SR	予約済み																								FCRSF		RDY	UDD	UDR	SOF	ENS	
	リセット値																									1	0	0	0	0	0	0	
0x0C	LCD_CLR	予約済み																								UDDC		予約済み	SOFC	予約済み			
	リセット値																									0	0		0		0	0	0
0x14	LCD_RAM (COM0)	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
0x18		予約済み																				S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
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		予約済み																				S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
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		予約済み																				S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32
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		予約済み																				S43	S42	S41	S40	S39	S38	S37	S36	S35	S34	S33	S32



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 77. 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	
0x38		予約済み																										S39	S38	S37	S36	S35	S34	S33
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	
0x40		予約済み																										S39	S38	S37	S36	S35	S34	S33
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	
0x48		予約済み																										S39	S38	S37	S36	S35	S34	S33
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	
0x50		予約済み																										S39	S38	S37	S36	S35	S34	S33

レジスタ境界アドレステーブルについては、表 5 (47 ページ) を参照してください。

17 汎用タイマ（TIM2 から TIM5）

17.1 TIM2 から TIM5 の概要

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ビットまたは 32ビット自動再ロードカウンタで構成されています。

入力信号（入力キャプチャ）のパルス長の測定や出力波形の生成（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、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。

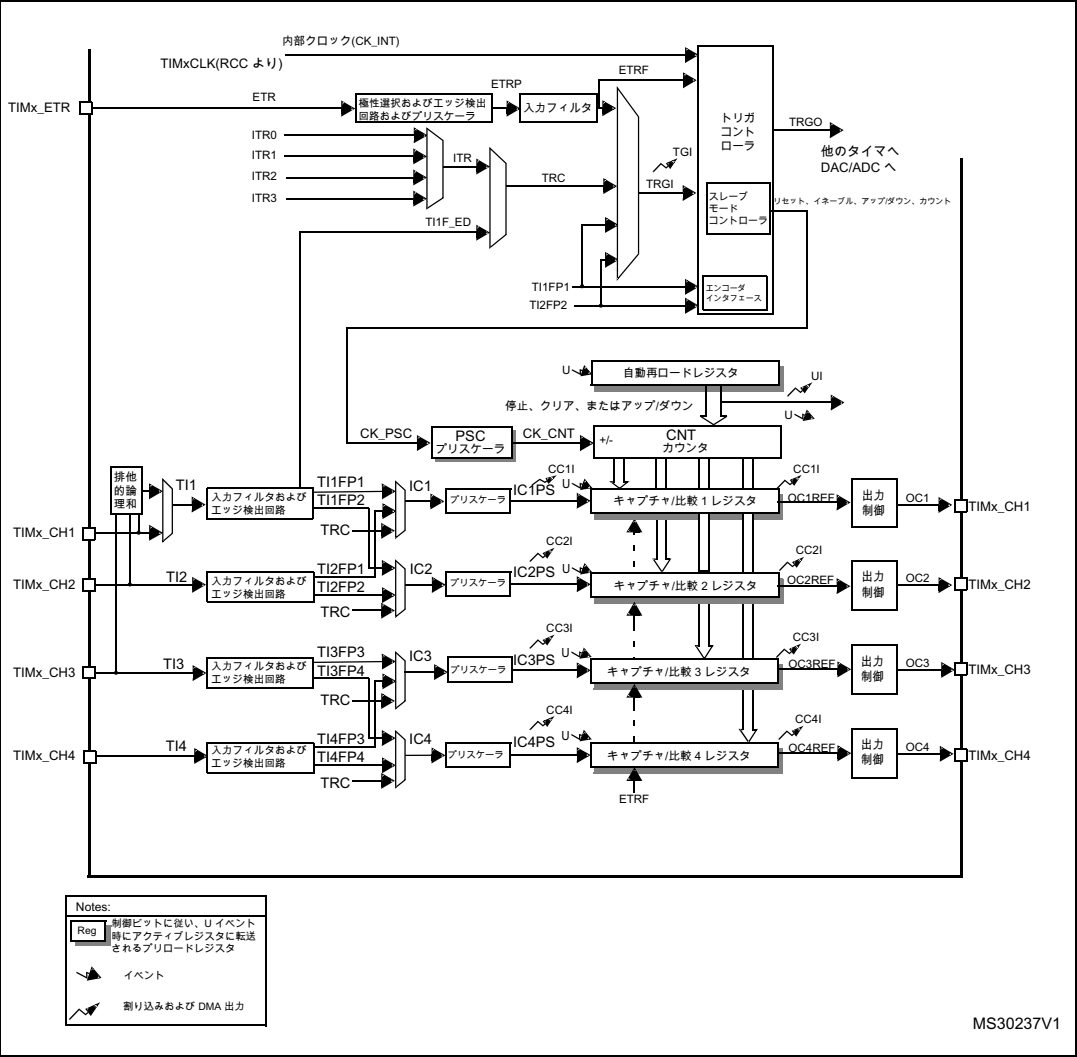
タイマは完全に独立していて、いかなるリソースも共有しません。また、[セクション 17.3.15](#) に説明されているように、共に同期させることができます。

17.2 TIM2 から TIM5 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（TIM3 および TIM4）または 32ビット（TIM5）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 65536 の間で（その場で）分周するための 16ビットプログラマブルプリスケラ。
- 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
- 位置決め目的で、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 84. 汎用タイマのブロック図



17.3 TIM2 から TIM5 の機能詳細

17.3.1 タイムベースユニット

プログラマブルタイマのメインブロックは、関連自動再ロードレジスタを持つ 16ビット/32ビットカウンタです。カウンタは、カウントアップできますが、カウントダウンまたはアップ/ダウン両方を実行することも可能です。カウンタクロックは、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、ソフトウェアによる読み書きができます。カウンタが動作中でも、読み書きは可能です。

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。更新イベントは、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) に達すると送信されます。これは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、各構成ごとに後で詳しく説明します。

カウンタのクロックは、TIMx_CR1 レジスタのカウンタ有効ビット (CEN) がセットされている場合にのみ、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)。

実際のカウンタ有効信号 CNT_EN は、CEN の 1 クロックサイクル後にセットされます。

プリスケアラの詳細

プリスケアラは、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、16ビット/32ビットレジスタ (TIMx_PSC レジスタ) を通じて制御される 16ビットカウンタに基づきます。また、この制御レジスタはバッファされるため、その場で変更できます。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。

 85 と  86 は、プリスケアラ比がその場で変更されたときのカウンタの動作例を示します。

図 85. プリスケアラ分周比が 1 から 2 に変化するときのカウンタのタイミング図

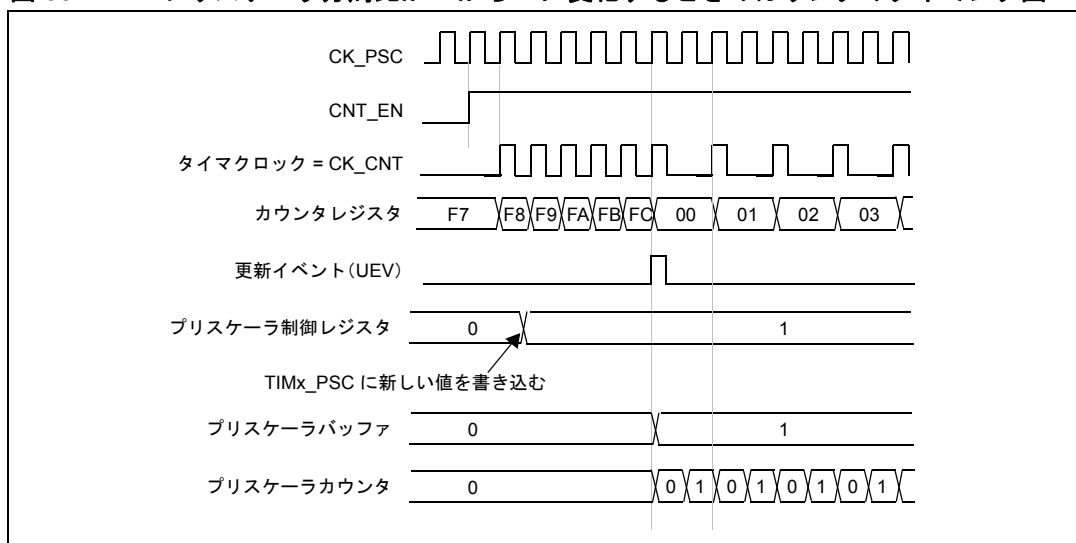
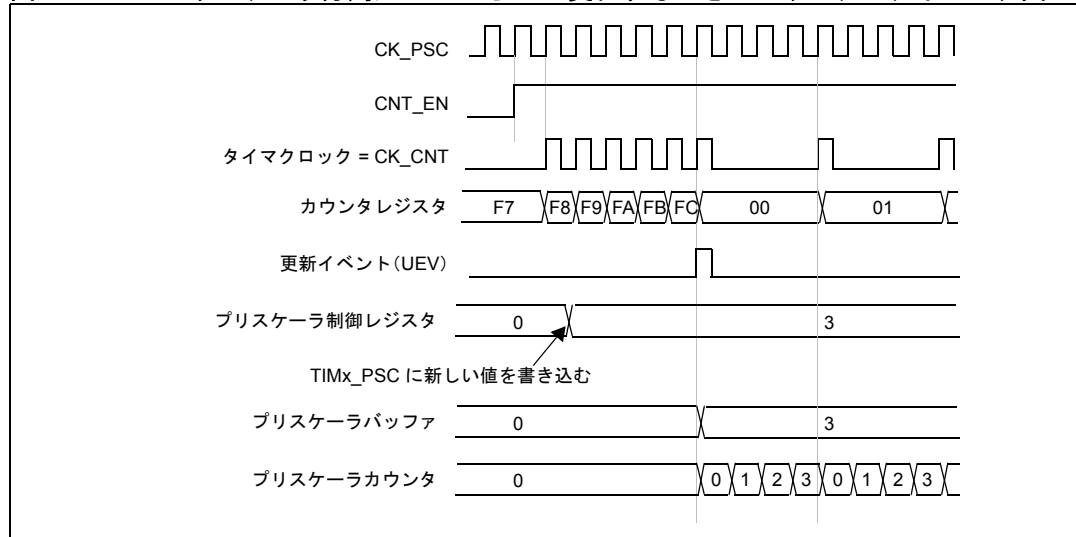


図 86. プリスケアラ分周比が 1 から 4 に変化するときのカウンタのタイミング図



17.3.2 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、TIMx_EGR レジスタの UG ビットをセットすることによって (ソフトウェアによって、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、TIMx_CR1 レジスタの UDIS ビットをセットすることで、ソフトウェア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、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が、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せん。
- 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 87. 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

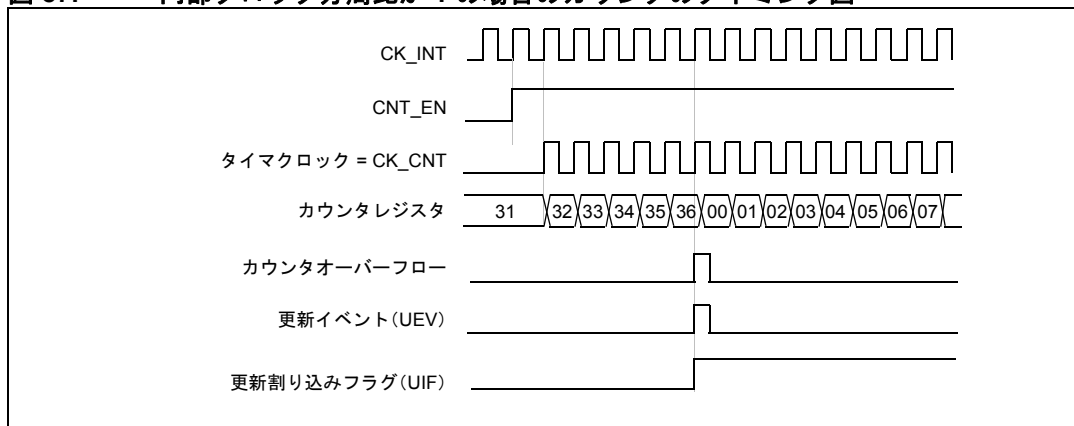


図 88. 内部クロック分周比が 2 の場合のカウンタのタイミング図

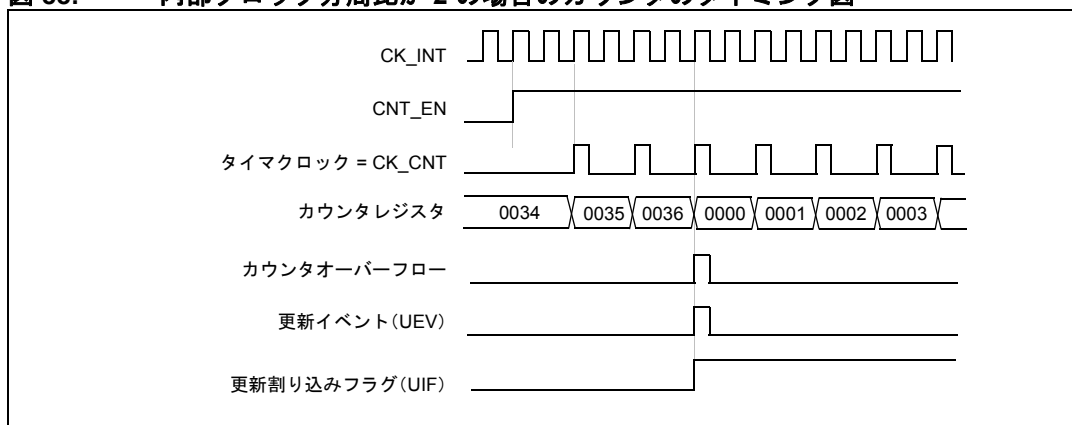


図 89. 内部クロック分周比が 4 の場合のカウンタのタイミング図

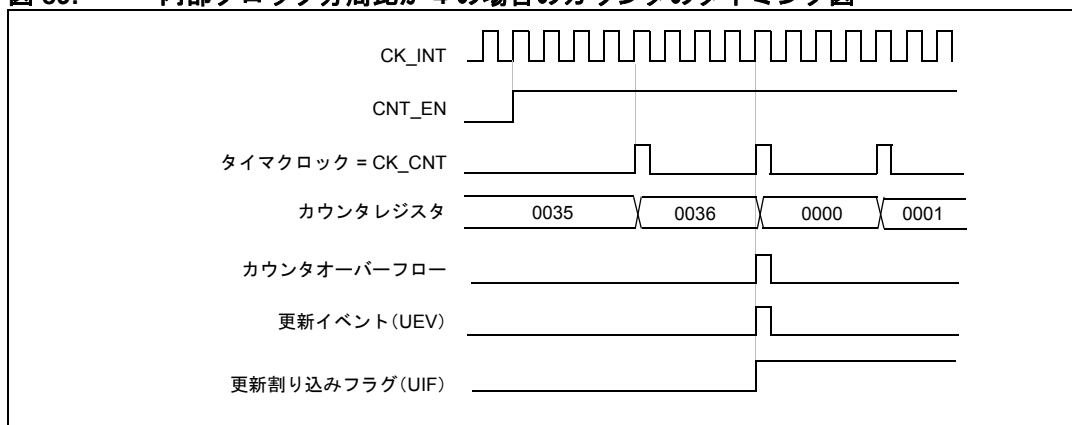


図 90. 内部クロック分周比が N の場合のカウンタのタイミング図

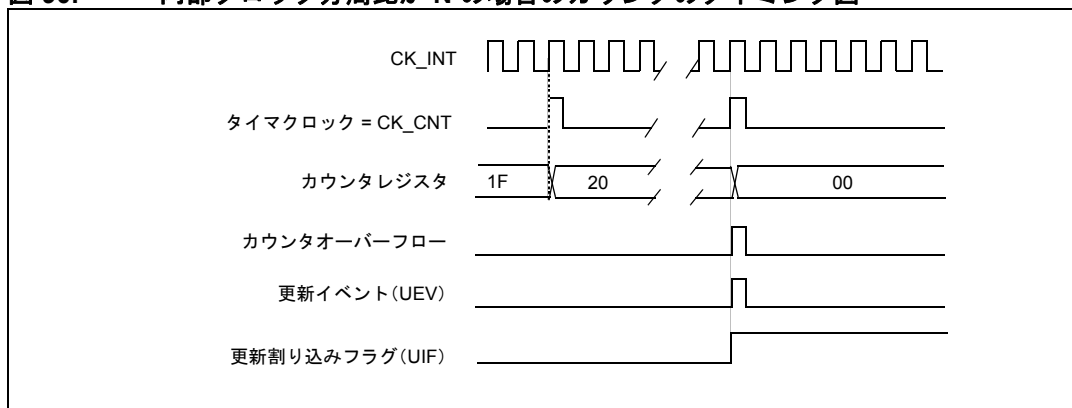


図 91. ARPE=0 (TIMx_ARR はプリロードされない) のときの更新イベントのカウンタタイミング図

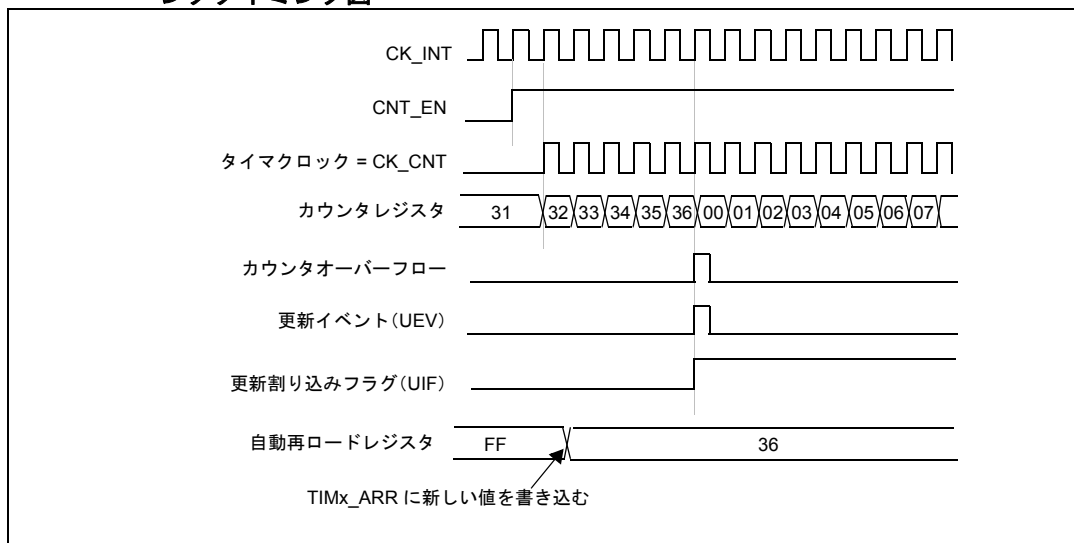
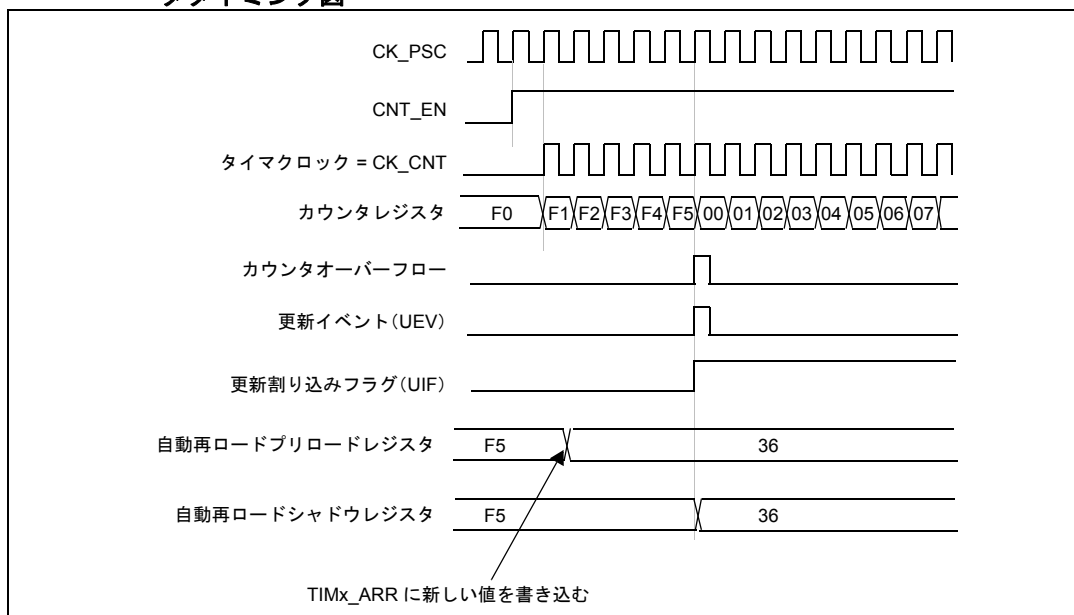


図 92. ARPE=1 (TIMx_ARR はプリロードされる) のときの更新イベントのカウンタタイミング図



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、TIMx_EGR レジスタの UG ビットをセットすることによって (ソフトウェアによって、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 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。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 (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。カウンタが再ロードされる前に自動再ロードが更新されるので、次の周期は予想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 93. 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

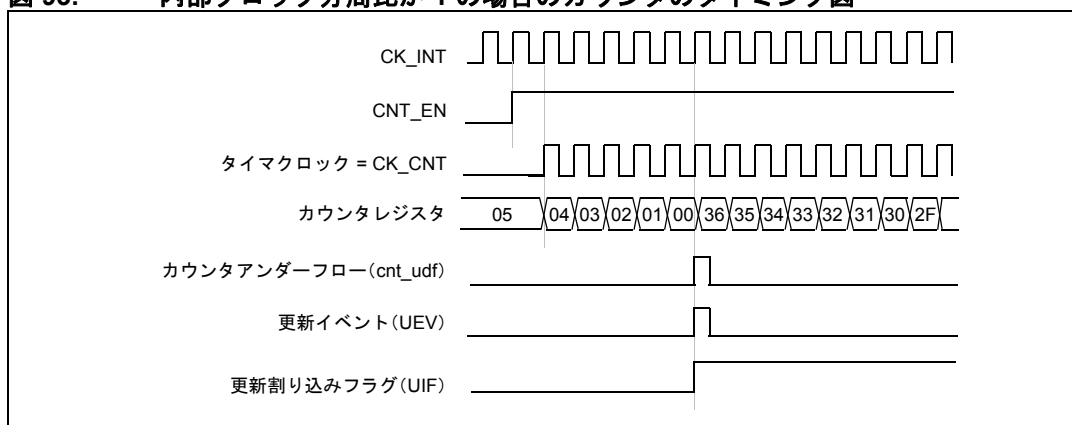


図 94. 内部クロック分周比が 2 の場合のカウンタのタイミング図

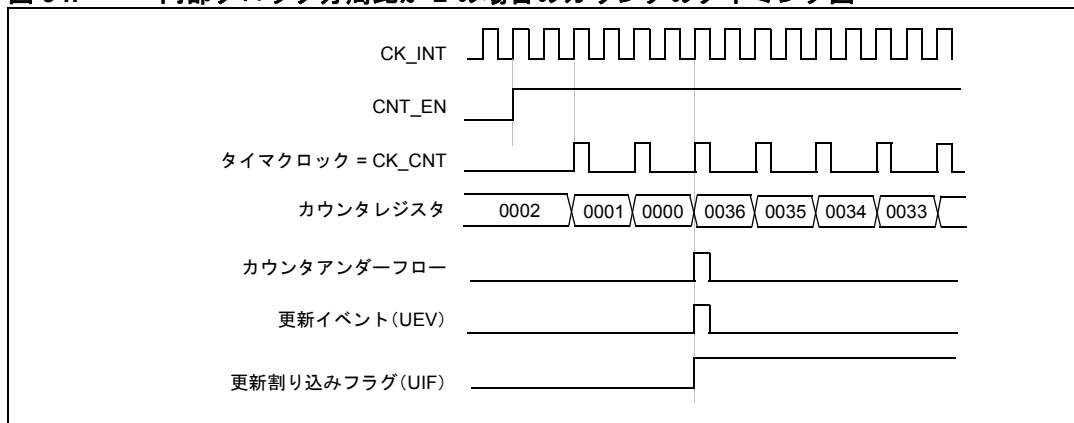


図 95. 内部クロック分周比が 4 の場合のカウンタのタイミング図

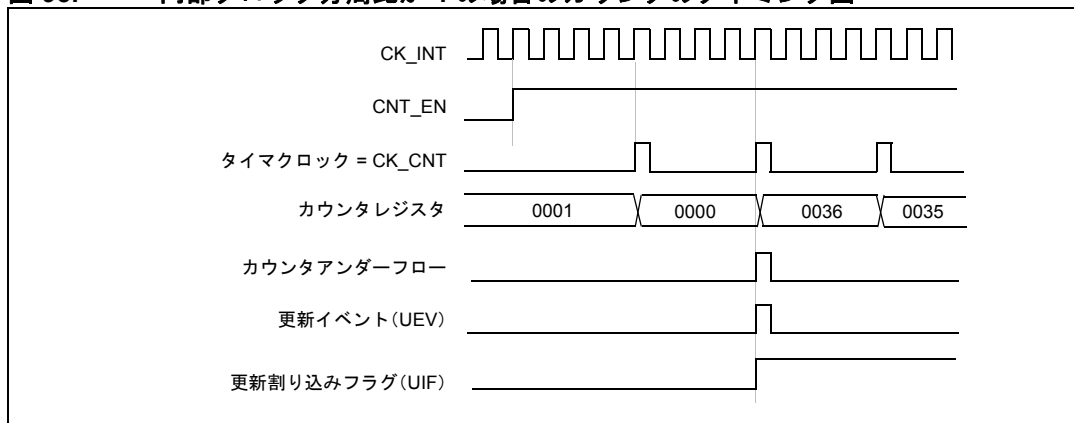


図 96. 内部クロック分周比が N の場合のカウンタのタイミング図

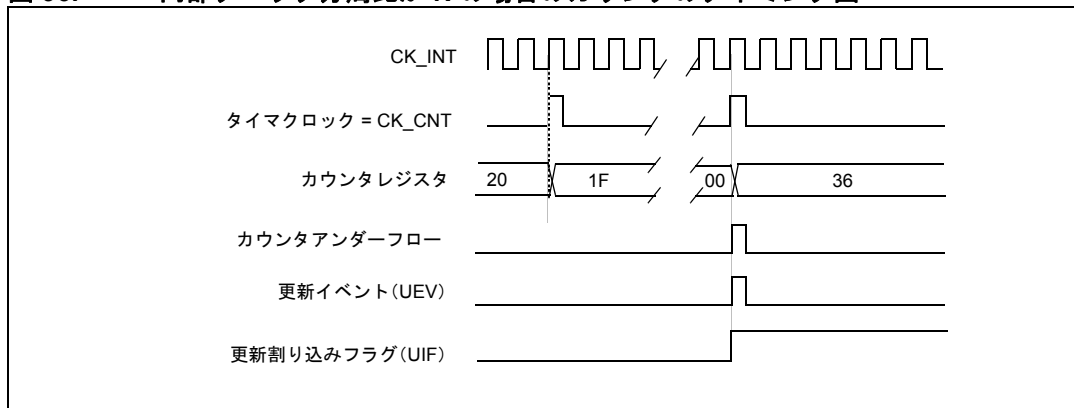
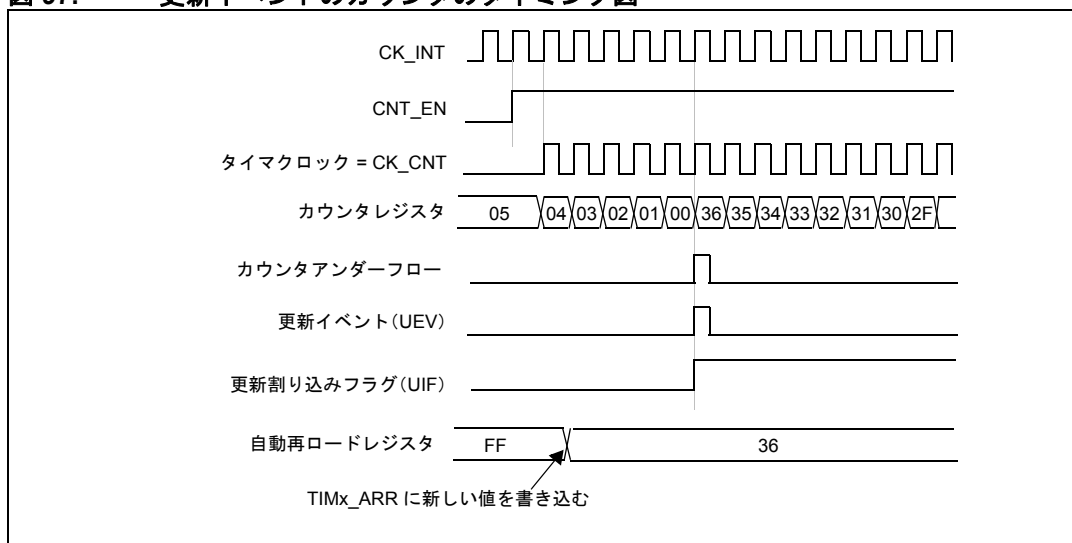


図 97. 更新イベントのカウンタのタイミング図



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" に等しくないときにアクティブとなります。出力で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、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、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)、生成できます。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。

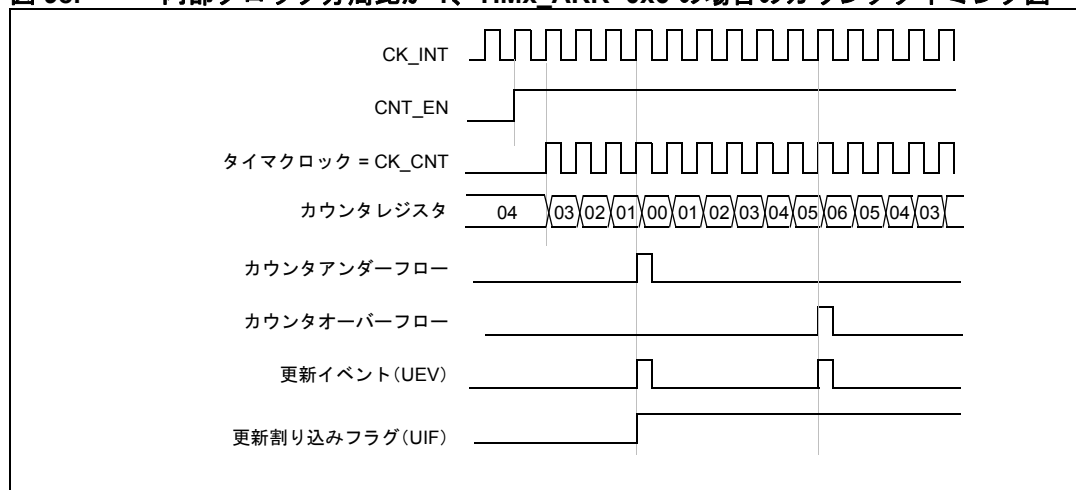
さらに、TIMx_CR1 レジスタの URS ビット (更新リクエスト選択) がセットされている場合は、UG ビットをセットすると更新イベント UEV が生成されますが、UIF フラグはセットされません (したがって、割り込みや DMA リクエストは送信されません)。これは、キャプチャイベント時にカウンタをクリアしているときに、更新とキャプチャの両方の割り込みを生成するのを防ぐためです。

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 98. 内部クロック分周比が 1、TIMx_ARR=0x6 の場合のカウンタタイミング図



1. ここでは、センターアラインモード 1 が使用されています (詳細については、[セクション 17.4.1 : TIMx 制御レジスタ 1 \(TIMx_CR1\) \(432 ページ\)](#) を参照してください)。

図 99. 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

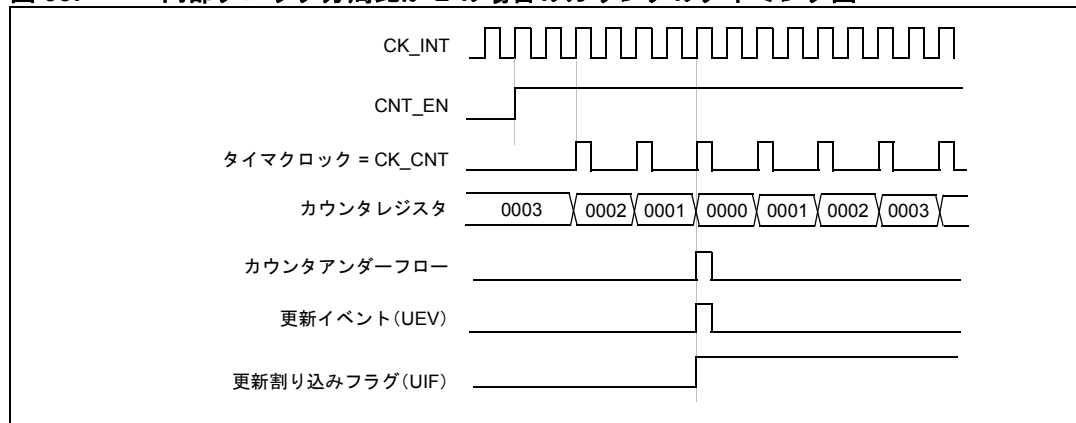
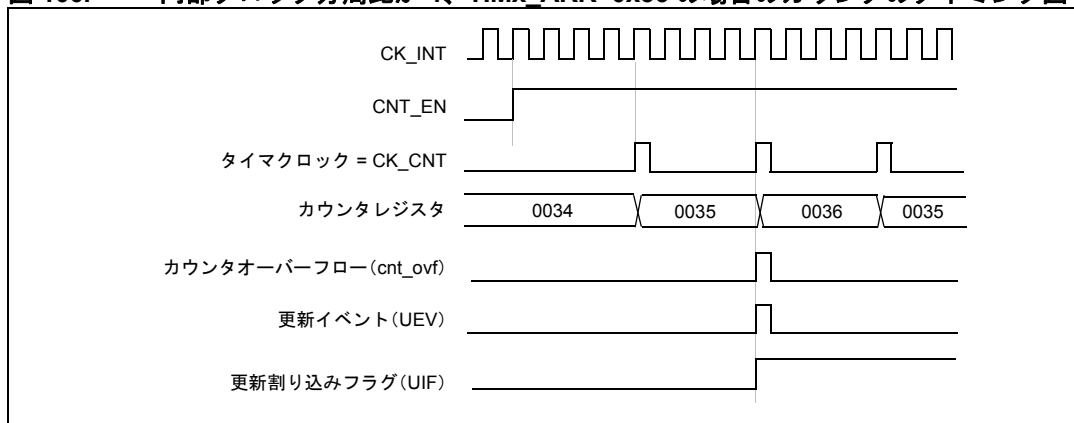


図 100. 内部クロック分周比が 4、TIMx_ARR=0x36 の場合のカウンタのタイミング図



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 101. 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

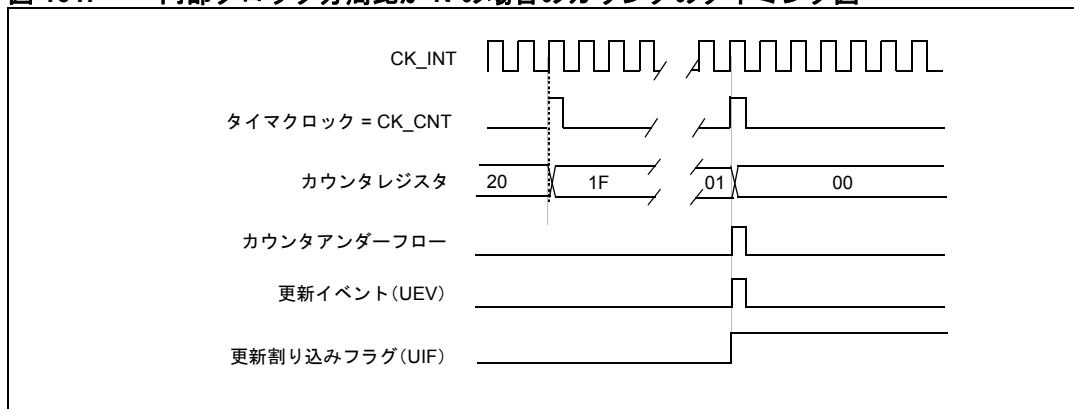


図 102. ARPE=1(カウンタアンダーフロー)のときの更新イベント時のカウンタタイミング図

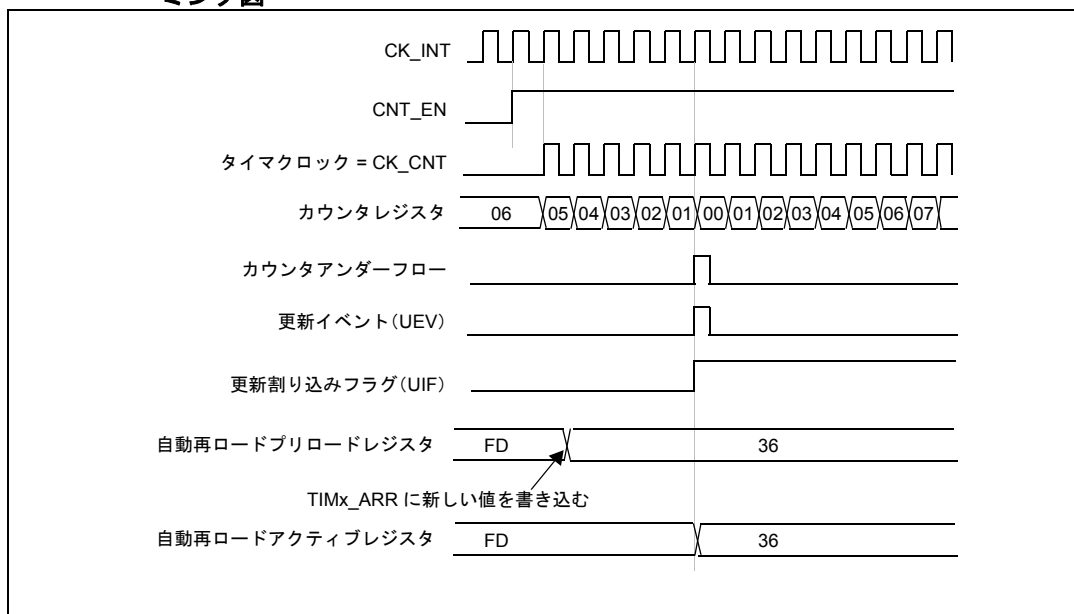
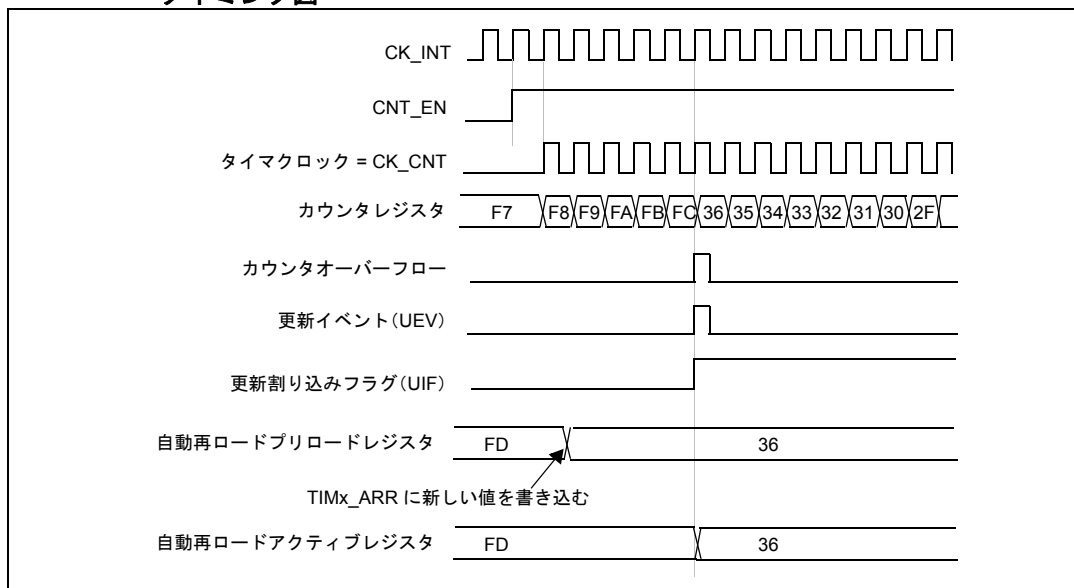


図 103. ARPE=1 (カウンタオーバーフロー) のときの更新イベント時のカウンタのタイミング図



17.3.3 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できます。

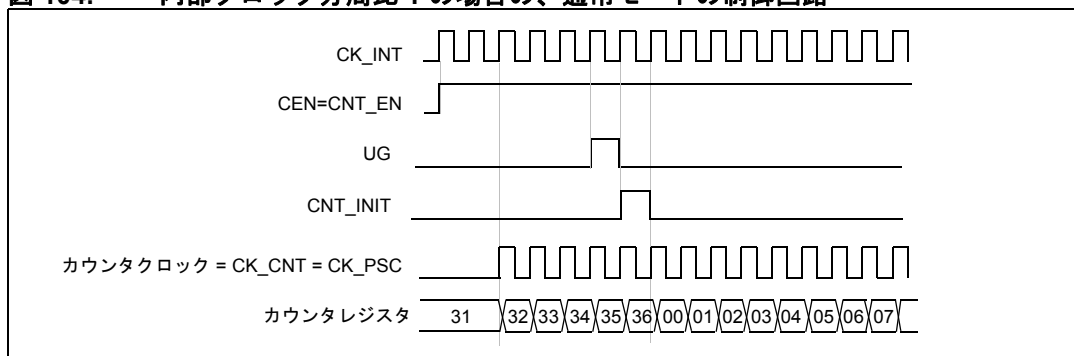
- 内部クロック (CK_INT)
- 外部クロックモード 1: 外部入力ピン (TIX)
- 外部クロックモード 2: 外部トリガ入力 (ETR)
- 内部トリガ入力 (ITRx): あるタイマを別のタイマのプリスケアラとして使用します。たとえば、タイマ 3 がタイマ 2 のプリスケアラとして動作するように構成できます。詳細については、[1 つのタイマを別のタイマのプリスケアラとして使用する \(426 ページ\)](#) を参照してください。

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、ソフトウェアによってのみ変更できます (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 104 は、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。

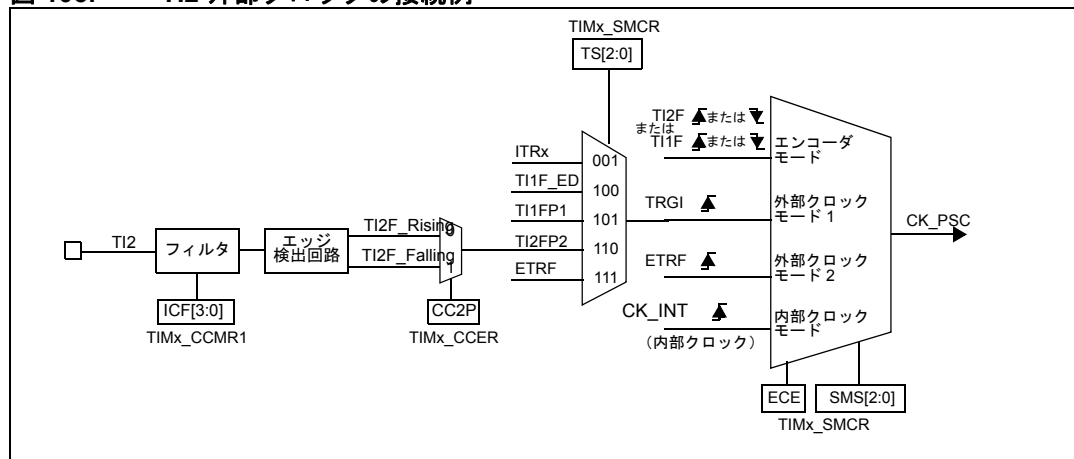
図 104. 内部クロック分周比 1 の場合の、通常モードの制御回路



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 105. 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：

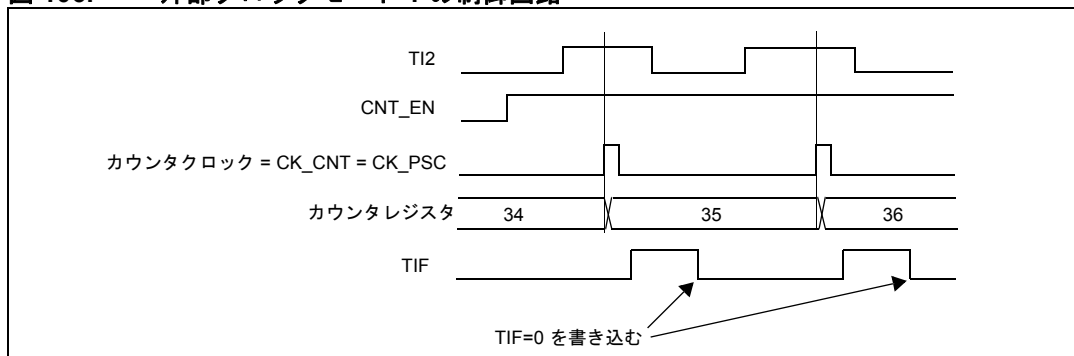
キャプチャプリスケアラはトリガには使用されないで、設定は不要です。

3. TIMx_CCER レジスタに CC2P=0 および CC2NP=0 を書き込むことによって、立ち上がりエッジの極性を選択します。
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 で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 106. 外部クロックモード 1 の制御回路



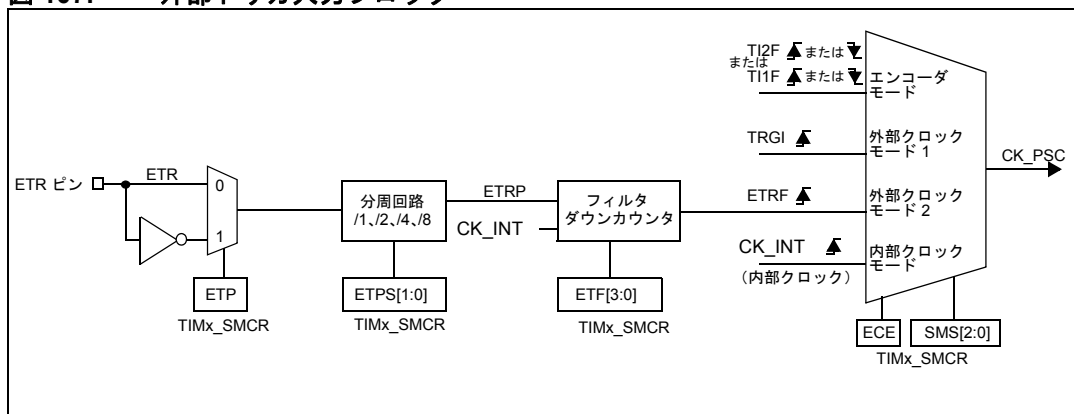
外部クロックソースモード 2

このモードは、TIM1 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でカウントできます。

図 107 に、外部トリガ入力ブロックの概要を示します。

図 107. 外部トリガ入力ブロック



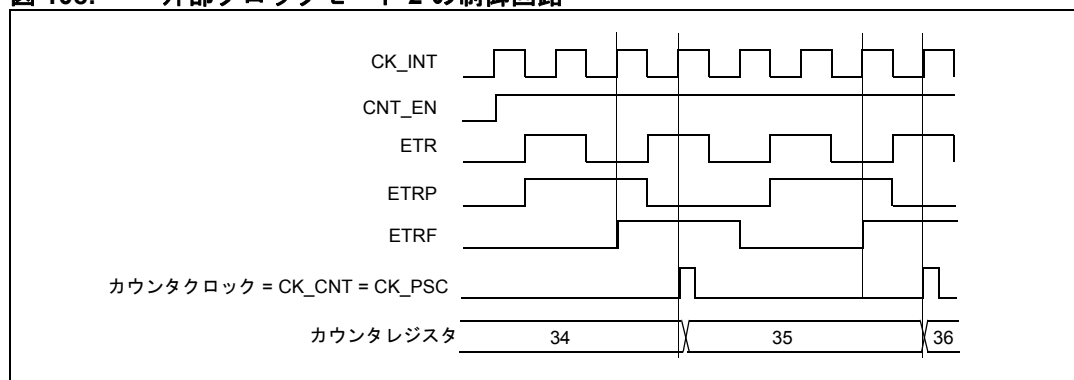
たとえば、アップカウンタを設定して、ETR の 2 回の立ち上がりエッジごとにカウントするように設定するには、以下の手順に従います。

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トクロックまでの遅延は、ETRP 信号の再同期回路によるものです。

図 108. 外部クロックモード 2 の制御回路



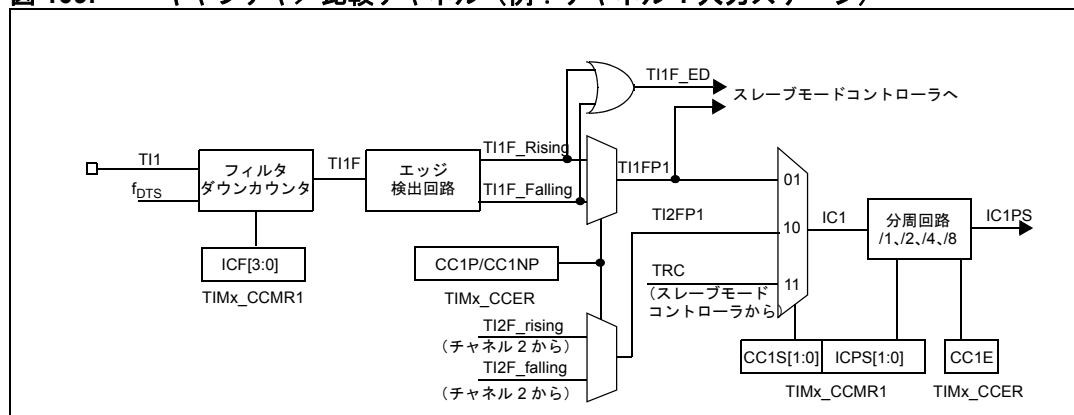
17.3.4 キャプチャ／比較チャネル

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、プリスケラを含む)、および出力カステージ (比較回路と出力制御を含む) 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。

入力カステージは、対応する TIx 入力をサンプリングして、フィルタリングを行った TIxF 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できる信号 (TIxFPx) を生成します。この信号はプリスケラを通じて、キャプチャレジスタ (ICxPS) に渡されます。

図 109. キャプチャ／比較チャネル (例：チャネル 1 入力カステージ)



出力カステージは、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 110. キャプチャ／比較チャネル 1 メイン回路

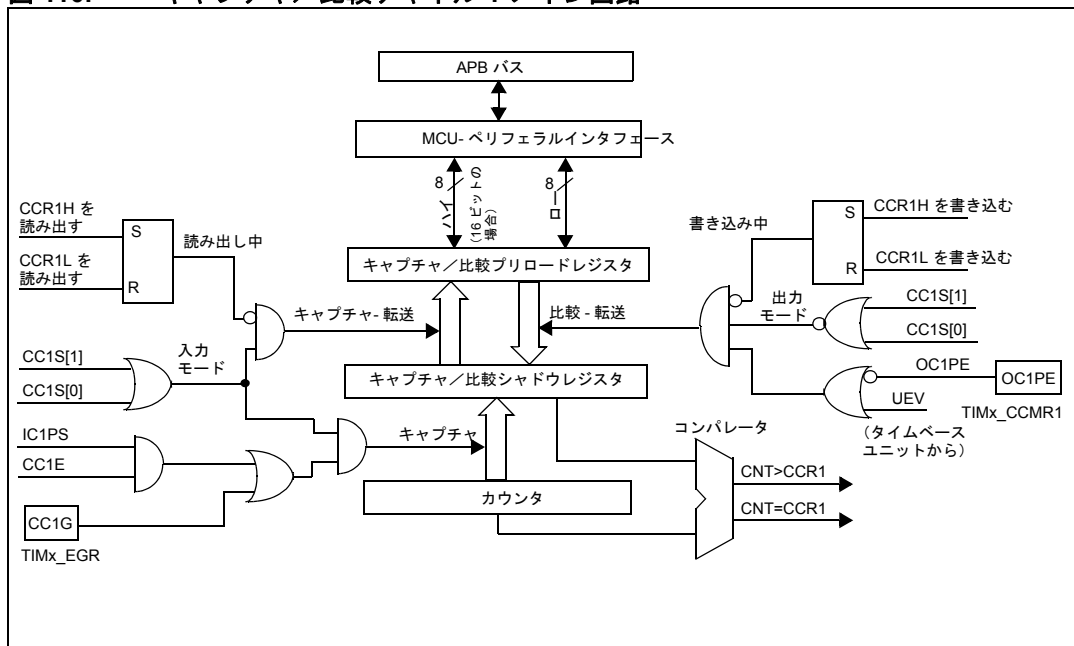
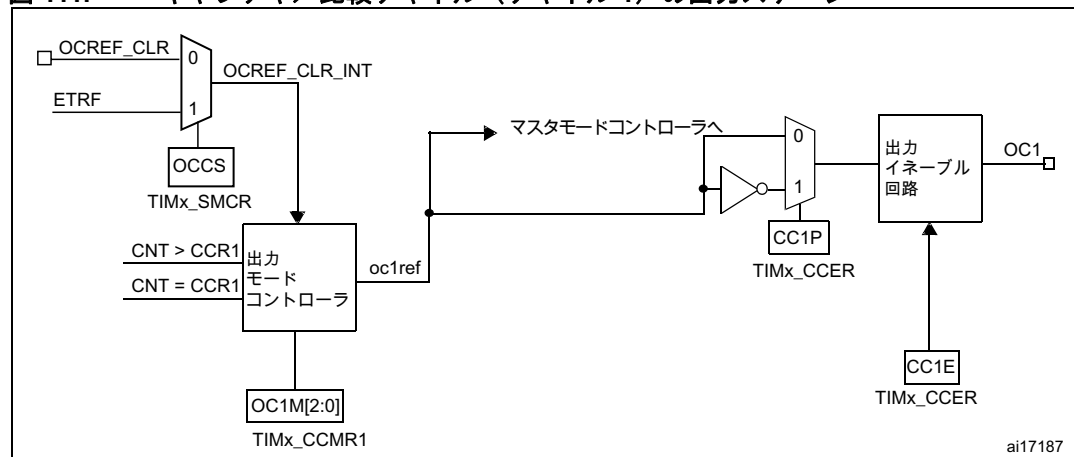


図 111. キャプチャ／比較チャネル（チャネル 1）の出力ステージ



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。読み書きのアクセスは、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。

17.3.5 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、キャプチャ/比較レジスタ (TIMx_CCRx) を使用してカウンタの値をラッチします。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。このためには、次の手順を使用します。

- 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。
- 入力が TIx の 1 つであるときに、タイマに接続する信号に応じて必要な入力フィルタ時間をプログラミングします (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 で変化を検証できます (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。
- TI1 チャンネルの変化でアクティブとされるエッジを選択します。このためには、TIMx_CCER レジスタの CC1P および CC1NP ビットに 0 を書き込みます (この場合、立ち上がりエッジの選択)。
- 入力プリスケアラをプログラムします。この例では有効な信号変化ごとにキャプチャを行うので、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)。
- 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。
- 必要の場合は、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できます。

17.3.6 PWM 入力モード

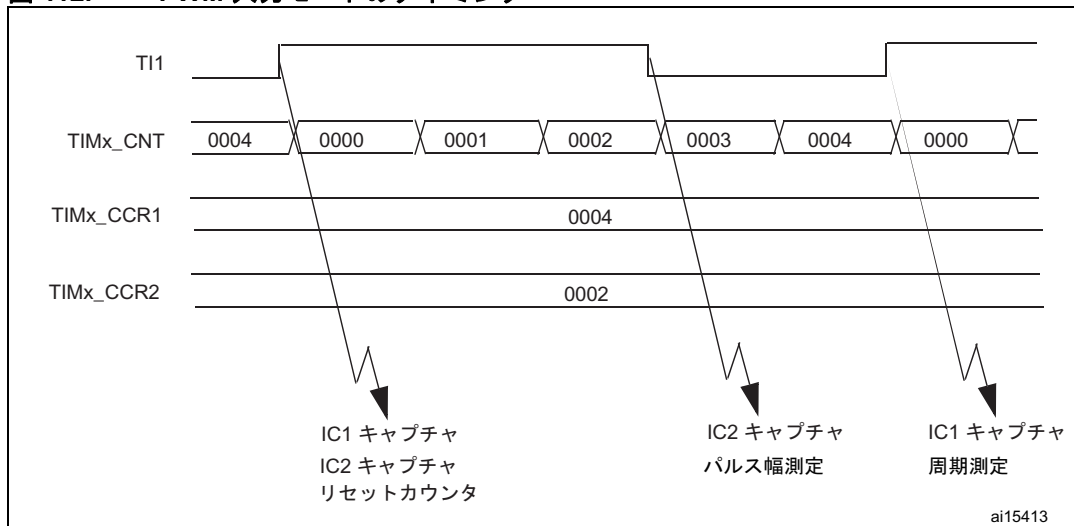
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 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、反対の極性を持つエッジ上でアクティブです。
- 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 周波数とプリスケール値によって、若干異なることがあります)。

- 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。
- CC1P ビットに "0"、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されます)。
- 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。
- CC2P ビットに "1"、CC1NP 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)。
- 有効なトリガ入力を選択します。すなわち、TIMx_SMCR レジスタの TS ビットに "101" を書き込みます (TI1FP1 を選択)。
- 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。すなわち、TIMx_SMCR レジスタの SMS ビットに "100" を書き込みます。
- キャプチャを有効にします。すなわち、TIMx_CCER レジスタの CC1E ビットと CC2E ビットに "1" を書き込みます。

図 112. PWM 入力モードのタイミング



17.3.7 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 と 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とするには、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。

この間、TIMx_CCRx シャドウレジスタとカウンタの比較は実行されており、これによりフラグがセットされます。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。

17.3.8 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します。

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マブル値を割り当てます。一致すると、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 (OCxM=001)、インアクティブにセット (OCxM=010)、または反転 (OCxM=011) などとなります。
- 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、TIMx_CCMRx レジスタの OCxPE ビットを使用して、プリロードレジスタを使用して、またはプリロードレジスタを使用せずにプログラミングできます。

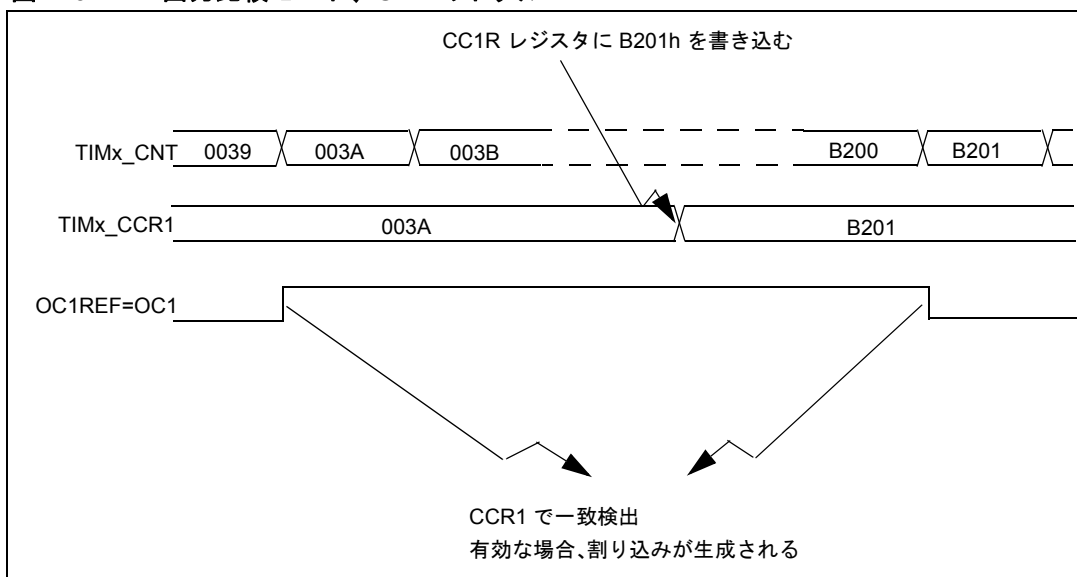
出力比較モードでは、更新イベント UEV は OCxREF および OCx 出力には影響を与えません。タイミングの分解能はカウンタの 1 カウント分です。出力比較モードは単一パルスを出力するためにも使用することができます (ワンパルスモード)。

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、CCR x 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。

ソフトウェアによって TIMx_CCRx レジスタを更新して、いつでも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 113](#) に示します。

図 113. 出力比較モード、OC1 のトグル



17.3.9 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、独自に選択することができます。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ため、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、TIM1_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フ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、 $TIMx\_CCRx \leq TIMx\_CNT$ または $TIMx\_CNT \leq TIMx\_CCRx$ かどうか判断されます (カウントの方向によります)。ただし、ETRF (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。

- 比較結果が変化したとき、または
- 出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) が固定構成 (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、ソフトウェアによって強制的に 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。

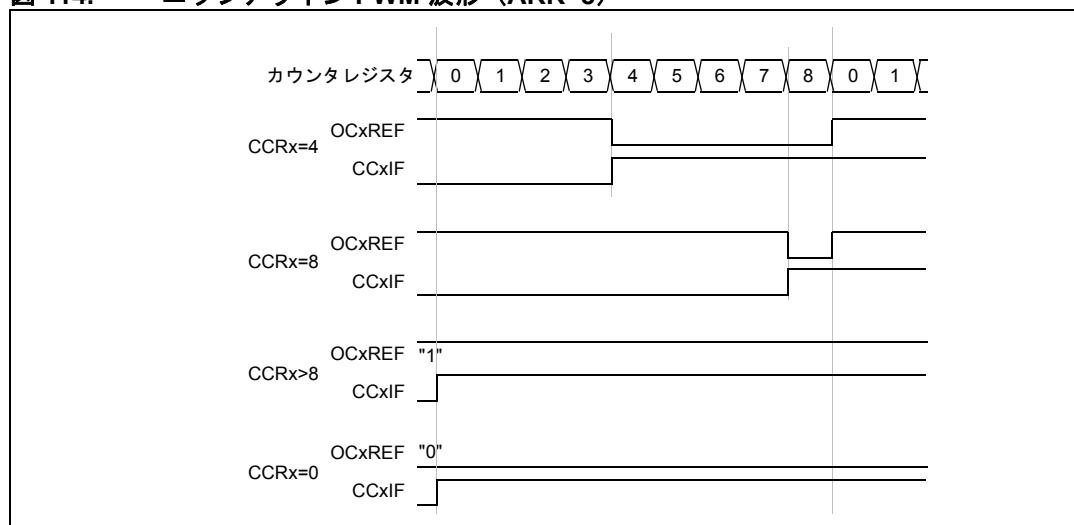
PWM エッジアラインモード

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 \(399 ページ\)](#) 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 (TIMx_ARR レジスタの) 自動再ロード値より大きい場合、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 114](#) 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 114. 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 \(402 ページ\)](#) 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。

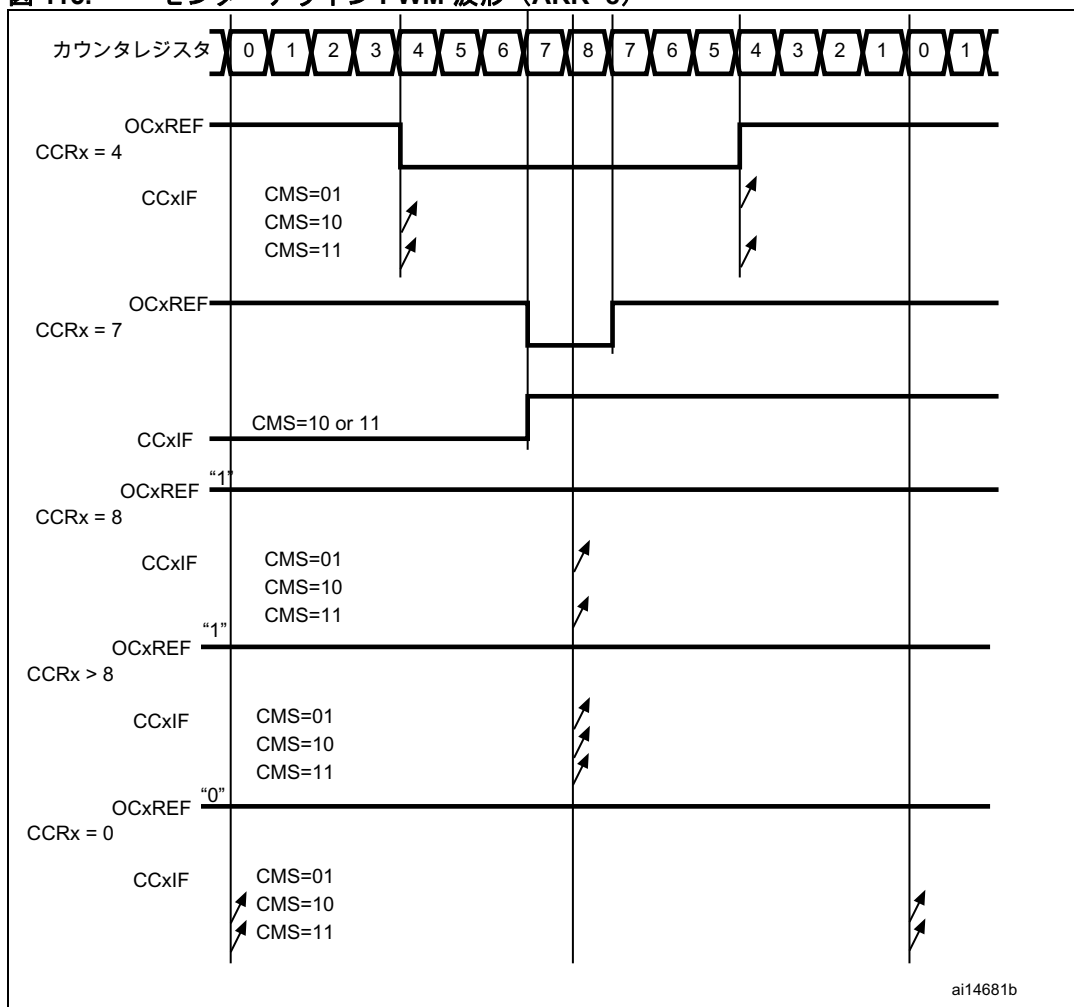
PWM センターアラインモード

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、ソフトウェアによって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 \(アップ/ダウンカウント\) \(404 ページ\)](#) を参照してください。

[図 115](#) 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 115. 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で開始するときには、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。

17.3.10 ワンパルスモード

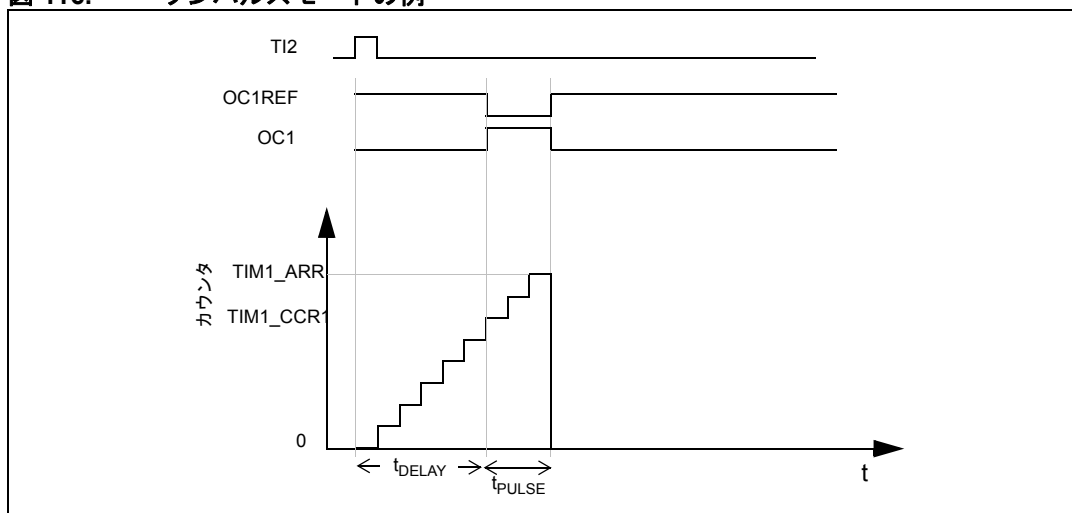
ワンパルスモード (OPM : One Pulse Mode) は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

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。

- アップカウントの場合 : $CNT < CCRx \leq ARR$ (特に、 $0 < CCRx$)
- ダウンカウントの場合 : $CNT > CCRx$

図 116. 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} 後に生成するとします。

TI2FP2 をトリガ 1 として使用します。

- TIMx_CCMR1 レジスタに $CC2S=01$ を書き込むことによって、TI2FP2 を TI2 に配置します。
- TI2FP2 で立ち上がりエッジを検出します。TIMx_CCER レジスタに $CC2P="0"$ および $CC2NP="0"$ 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 ($TIMx\_ARR - TIMx\_CCR + 1$) によって定義されます。
- 比較値と一致したときに "0" から "1" に変化し、カウンタが自動再ロード値に達したときに "1" から "0" に変化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）、次の更新イベント（カウンタが自動再ロード値から 0 に戻る時点）でカウンタが停止するように、TIMx_CR1 レジスタの OPM ビットに 1 を書き込みます。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 高速有効：

ワンパルスモードでは、TIMx 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 ($t_{\text{DELAY min}}$) 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ときにのみ動作します。

17.3.11 外部イベントによる OCxREF 信号のクリア

指定されたチャンネルの OCxREF 信号は ETRF 入力をハイレベルとする（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）ことでローレベルにすることができます。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。

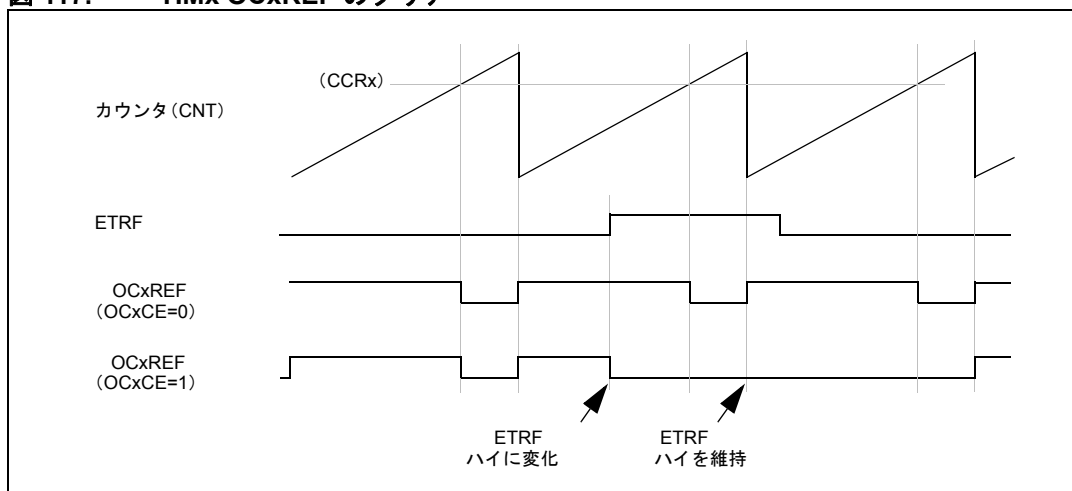
この機能は、出力比較モードと PWM モードでのみ使用でき、強制モードでは機能しません。

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アラはオフ状態に維持する必要があります。このため、TIMx_SMCR レジスタの ETPS[1:0] ビットを "00" にクリアします。
2. 外部クロックモード 2 は無効にする必要があります。このため、TIM1_SMCR レジスタの ECE ビットを "0" にクリアします。
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構成できます。

[図 117](#) は、OCxCE 有効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 117. TIMx OCxREF のクリア



注： デューティサイクル 100% の PWM の場合 (CCR_x>ARR)、OC_xREF は次のカウンタオーバーフローで再び有効になります。

17.3.12 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 はクリアされたままでなければなりません。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 78](#) 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の 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供給されます。ただし、カウンタが有効である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、カウントパルスと方向信号を生成します。カウンタがカウントアップまたはカウントダウンするシーケンスに応じて、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、あらゆる入力（TI1 または TI2）において変化がある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、TIMx_ARR は、開始前に設定されなければなりません。同様に、キャプチャ、比較、プリスケール、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 78. 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、外部に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、通常は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 118 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。この例では、次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、TI1FP2 は TI2 に配置)
- CC1P="0"、CC1NP="0"、IC1F="0000" (TIMx_CCER レジスタ、TI1FP1 非反転、TI1FP1=TI1)
- CC2P="0"、CC2NP="0"、IC2F="0000" (TIMx_CCER レジスタ、TI2FP2 非反転、TI2FP2=TI2)
- SMS="011" (TIMx_SMCR レジスタ、両方の入力が立ち上がりと立ち下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 118. 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

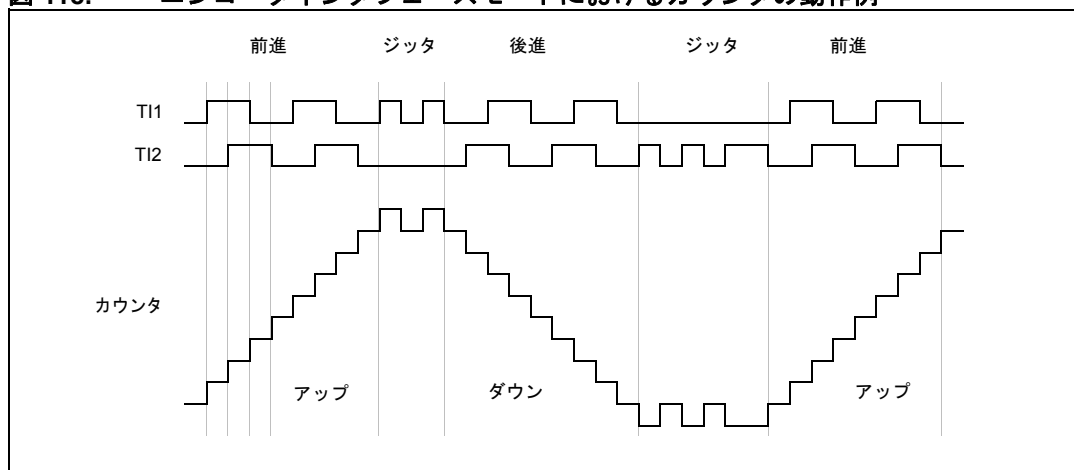
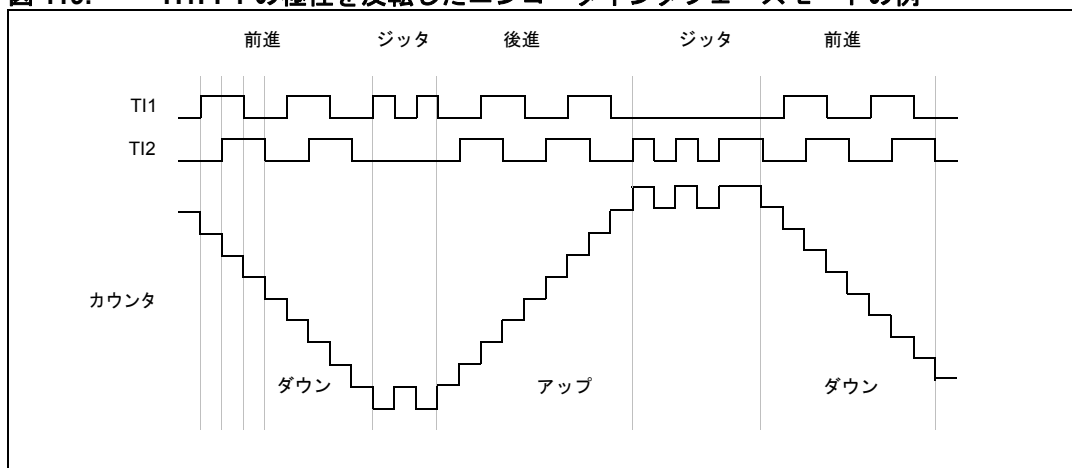


図 119 は、TI1FP1 の極性を反転したときのカウンタの動作を示します (上記と同じ構成ですが、CC1P=1)。

図 119. 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 リクエストでこの値を読み出すこともできます。

17.3.13 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 個の入力ピンを組み合わせた 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。

17.3.14 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、トリガモードなど、いくつかの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。

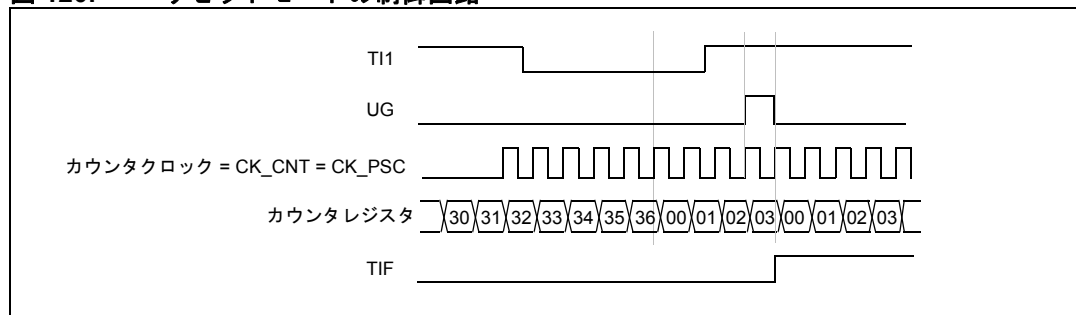
- TI1 の立ち上がりエッジを検出するように、チャンネル 1 を設定します。入力フィルタ時間を設定します（この例ではフィルタは不要なので、IC1F=0000 のままにしておきます）。キャプチャプリスケアラはトリガには使用されないため、設定は不要です。CC1S ビットは、入力キャプチャソースのみを選択します (TIMx_CCMR1 レジスタの CC1S=01)。TIMx_CCER レジスタに CC1P=0 および CC1NP=0 を書き込んで、極性を有効にします（そして、立ち上がりエッジのみを検出します）。
- TIMx_SMCR レジスタに SMS=100 を書き込むことによって、タイマをリ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 入力の再同期回路によるものです。

図 120. 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。

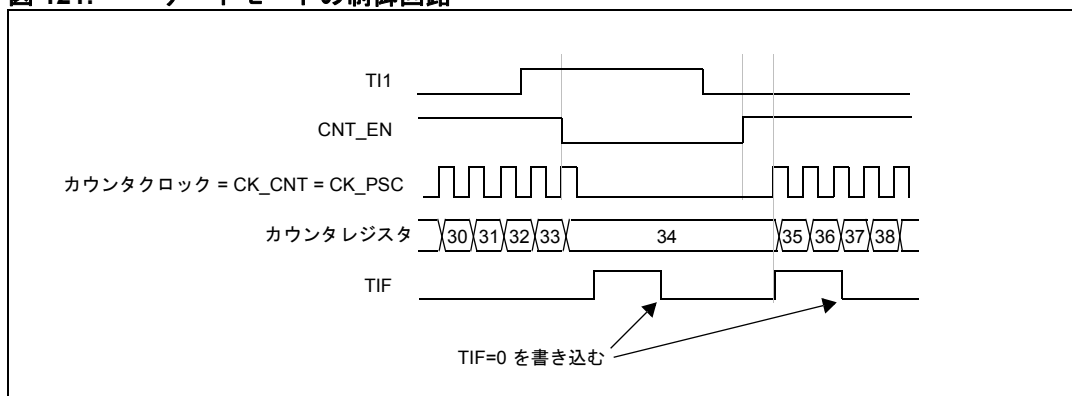
次の例では、アップカウンタは TI1 入力があるときだけカウントします。

- 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 レジスタに CC1P=1 および CC1NP=0 を書き込んで、極性を有効にします (そして、ローレベルのみを検出します)。
- 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。
- 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 (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 がローである限り内部クロックでカウントを開始し、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 121. ゲートモードの制御回路



注: ゲートモードはエッジではなくレベルに対応して動作するため、ゲートモードでは設定「 $CCxP=CCxNP=1$ 」(立ち上がりおよび立ち下がリエッジの検出)の影響はありません。

スレーブモード：トリガモード

選択された入力のイベントに対応して、カウンタが開始できます。

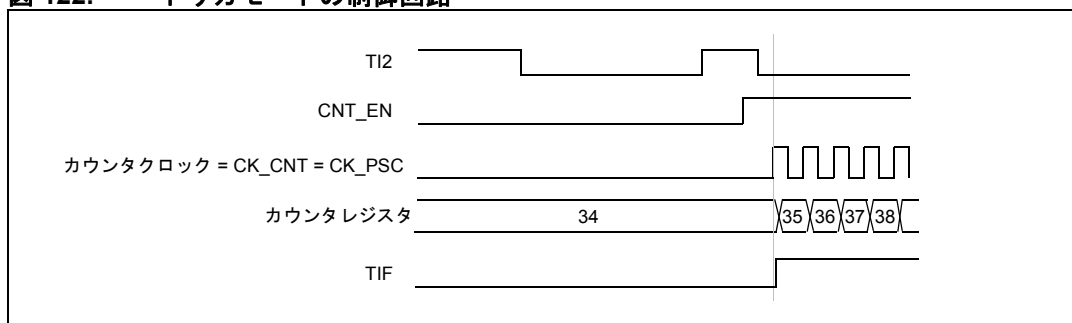
次の例では、TI2 入力の立ち上がりエッジに応じて、アップカウンタがスタートします。

- 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ラはトリガには使用されないので、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 レジスタに CC2P=1 および CC2NP=0 を書き込んで、極性を有効にします (そして、ローレベルのみを検出します)。
- 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 122. 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。

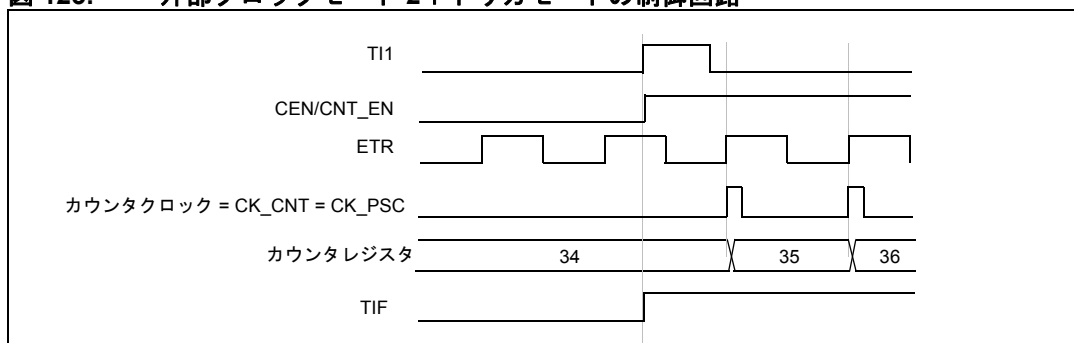
次の例では、アップカウンタは、TI1 の立ち上がりエッジが発生すると、ETR 信号の立ち上がりエッジのたびにインクリメントされます。

1. TIMx_SMCR レジスタを次のようにプログラミングして、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 を次のように設定します。
 - IC1F=0000：フィルタなし
 - 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 レジスタに CC1P=0 および 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）。
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 123. 外部クロックモード 2+トリガモードの制御回路



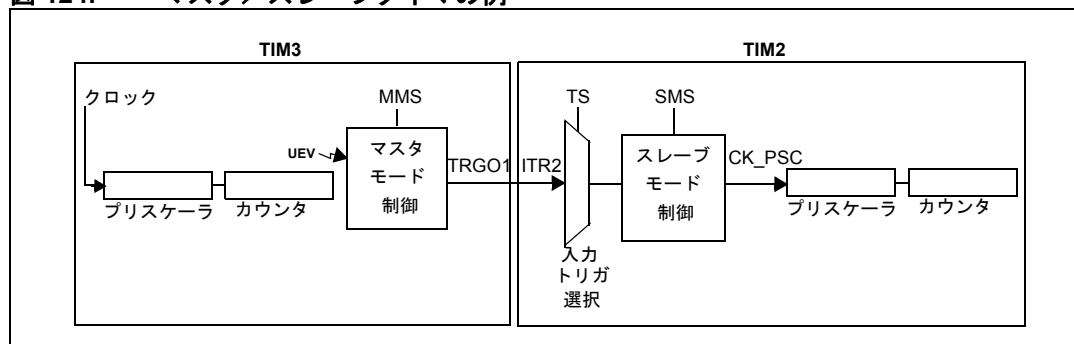
17.3.15 タイマの同期

タイマ間で同期をとったり連結を行うなど、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 124: 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。

1 つのタイマを別のタイマのプリスケアラとして使用する

図 124. マスタ/スレーブタイマの例



たとえば、TIM3 が TIM2 のプリスケアラとして動作するように設定できます。図 124 を参照してください。このためには、次の操作を行います。

- 更新イベント UEV ごとに定期的なトリガ信号を出力するように、TIM3 をマスタモードに設定します。TIM3_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。
- TIM3 の出力 TRGO1 を TIM2 に接続するには、内部トリガとして ITR2 を使用して TIM2 をスレーブモードに設定します。このためには、TIM2_SMCR レジスタの TS ビットに 010 を書き込みます。
- 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 (TIM2_SMCR レジスタの SMS=111 を書き込みます)。これにより TIM2 は、TIM3 の定期的なトリガ信号の立ち上がりエッジ (TIM3 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。
- 最後に、それぞれの TIMx_CR1 レジスタの CEN ビット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: TIM3 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 TIM2 カウンタのクロックとして使用されます。

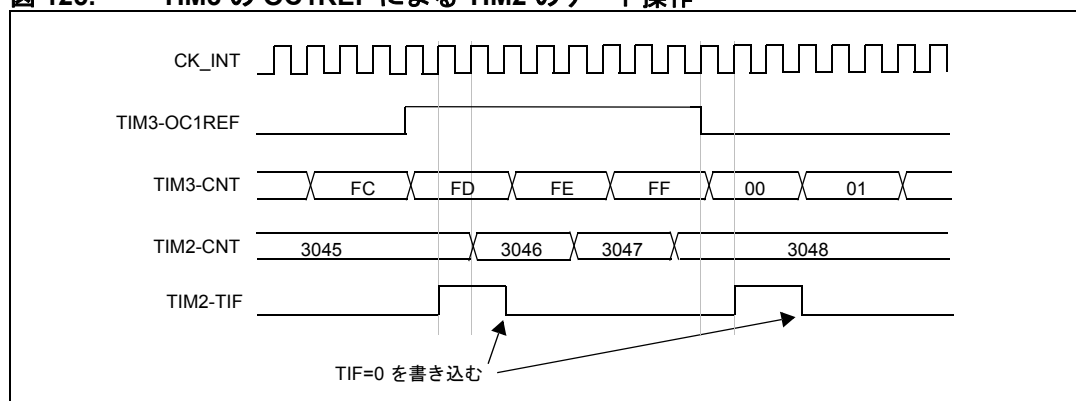
1 つのタイマを使用して別のタイマを有効にする

この例では、タイマ3 の出力比較 1 で TIM2 の有効/無効を制御します。接続については、[図 124](#) を参照してください。TIM2 は、TIM3 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 ($f_{CK\_CNT} = f_{CK\_INT}/3$) をプリスケアラで 3 分周したものです ($f_{CK\_CNT} = f_{CK\_INT}/3$)。

- TIM3 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 (TIM3_CR2 レジスタの MMS=100)。
- TIM3 の OC1REF 波形を設定します (TIM3_CCMR1 レジスタ)。
- TIM2 が TIM3 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)。
- TIM2 をゲートモードに設定します (TIM2_SMCR レジスタの SMS=101)。
- CEN ビット (TIM2_CR1 レジスタ) に "1" を書き込んで、TIM2 を有効にします。
- CEN ビット (TIM3_CR1 レジスタ) に "1" を書き込んで、TIM3 を開始します。

注： カウンタ 2 のクロックは、カウンタ 1 と同期されません。このモードは、TIM2 カウンタ有効信号にのみ影響します。

図 125. TIM3 の OC1REF による TIM2 のゲート操作

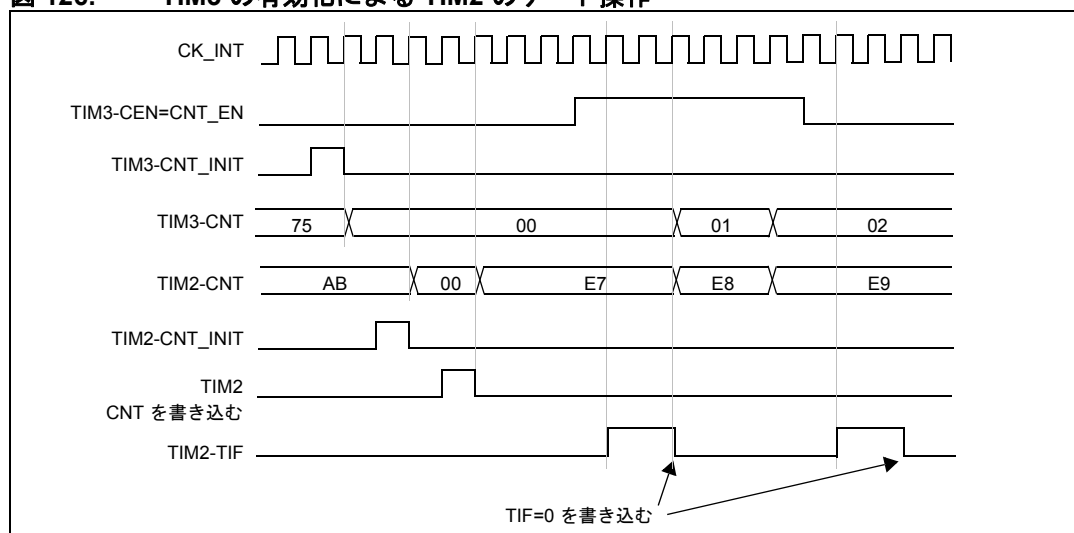


[図 125](#) の例では、TIM2 のカウンタとプリスケアラは、開始前に初期化されていません。したがって、現在値からカウントを開始します。TIM3 を開始する前に、両方のタイマをリセットすることによって、任意の値からカウントを開始することが可能です。この場合、タイマカウンタに任意の値を書き込むことができます。TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアによって容易にタイマをリセットできます。

次の例では、TIM3 と TIM2 を同期させます。TIM3 はマスタであり、0 から開始します。TIM2 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。TIM3_CR1 レジスタの CEN ビットに"0"を書き込むことによって TIM3 を無効にすると、TIM2 は停止します。

- TIM3 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 (TIM3_CR2 レジスタの MMS=100)。
- TIM3 の OC1REF 波形を設定します (TIM3_CCMR1 レジスタ)。
- TIM2 が TIM3 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)。
- TIM2 をゲートモードに設定します (TIM2_SMCR レジスタの SMS=101)。
- UG ビット (TIM3_EGR レジスタ) に"1"を書き込むことによって、TIM3 をリセットします。
- UG ビット (TIM2_EGR レジスタ) に"1"を書き込むことによって、TIM2 をリセットします。
- TIM2 のカウンタ (TIM2_CNT) に"0xE7"を書き込むことによって、TIM2 を 0xE7 に初期化します。
- CEN ビット (TIM2_CR1 レジスタ) に"1"を書き込んで、TIM2 を有効にします。
- CEN ビット (TIM3_CR1 レジスタ) に"1"を書き込んで、TIM3 を開始します。
- CEN ビット (TIM3_CR1 レジスタ) に"0"を書き込んで、TIM3 を停止します。

図 126. TIM3 の有効化による TIM2 のゲート操作

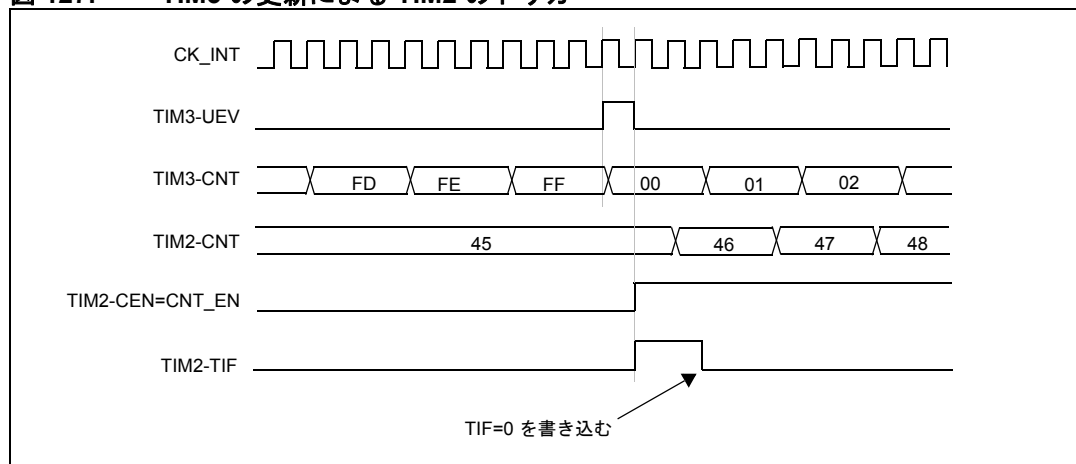


1 つのタイマを使用して別のタイマを開始する

この例では、タイマ 3 の更新イベントによってタイマ 2 の有効/無効をセットします。接続については、[図 124](#) を参照してください。タイマ 1 によって更新イベントが生成されると、タイマ 2 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 2 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 ($f_{CK\_CNT} = f_{CK\_INT}/3$) をプリスケアラで 3 分周したものです ($f_{CK\_CNT} = f_{CK\_INT}/3$)。

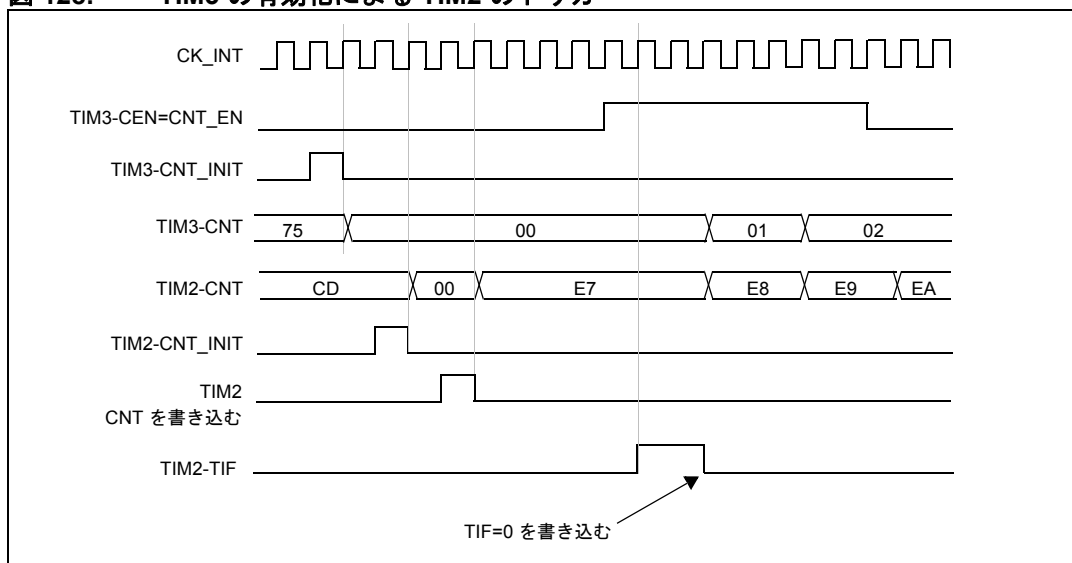
- 更新イベント (UEV) をトリガ出力として送信するように、TIM3 をマスタモードに設定します (TIM3_CR2 レジスタの MMS=010)。
- TIM3 の周期を設定します (TIM3_ARR レジスタ)。
- TIM2 が TIM3 から入力トリガを受け取るように設定します (TIM2_SMCR レジスタの TS=000)。
- TIM2 をトリガモードに設定します (TIM2_SMCR レジスタの SMS=110)。
- CEN ビット (TIM3_CR1 レジスタ) に“1”を書き込んで、TIM3 を開始します。

図 127. TIM3 の更新による TIM2 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 128](#) は、[図 127](#) と同じ構成ですが、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 (TIM2_SMCR レジスタの SMS=110)。

図 128. TIM3 の有効化による TIM2 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、TIM3 が TIM2 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 124](#) を参照してください。このためには、次の操作を行います。

- TIM3 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 (TIM3_CR2 レジスタの MMS=010)。TIM3 は、カウンタオーバーフローごとに定期的な信号を出力します。
- TIM3 の周期を設定します (TIM3_ARR レジスタ)。
- TIM2 が TIM3 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)。
- TIM2 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)。
- CEN ビット (TIM2_CR1 レジスタ) に "1" を書き込んで、TIM2 を開始します。
- CEN ビット (TIM3_CR1 レジスタ) に "1" を書き込んで TIM3 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する

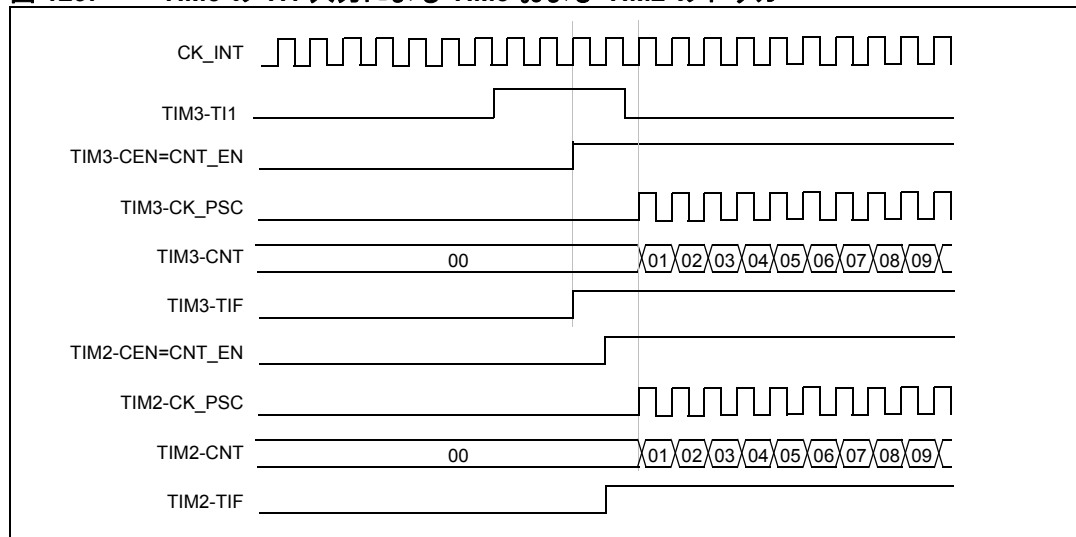
この例では、TI1 入力の立ち上がりエッジで TIM3 を有効にし、TIM3 が有効になると、TIM2 を有効にします。接続については、[図 124](#) を参照してください。カウンタの同時性を確保するため、TIM3 はマスタ/スレーブモードに設定する必要があります (TI1 に対してはスレーブ、TIM2 に対してはマスタ)。

- 有効信号をトリガ出力として送信するように、TIM3 をマスタモードに設定します (TIM3_CR2 レジスタの MMS=001)。
- TIM3 をスレーブモードに設定して、TI1 から入力トリガを取得するようにします (TIM3_SMCR レジスタの TS=100)。
- TIM3 をトリガモードに設定します (TIM3_SMCR レジスタの SMS=110)。
- MSM=1 (TIM3_SMCR レジスタ) を書き込むことによって、TIM3 をマスタ/スレーブモードに設定します。
- TIM2 が TIM3 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)。
- TIM2 をトリガモードに設定します (TIM2_SMCR レジスタの SMS=110)。

TI1 (TIM3) で立ち上がりエッジが発生すると、両方のカウンタが同期して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、TIM3 の CNT_EN と CK_PSC の間に遅延が挿入されます。

図 129. TIM3 の TI1 入力による TIM3 および TIM2 のトリガ



17.3.16 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®-M3 コア停止)、DBGMCU モジュールの DBG_TIMx_STOP 設定ビットに応じて、TIMx カウンタは通常通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 30.16.2: 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。

17.4 TIMx 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。それ以外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。

17.4.1 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
予約済み						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 (ETR、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 : $t_{DTS} = 2 \times t_{CK\_INT}$
 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有効

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
- スレープ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 : 更新無効

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
- スレープ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、スレープ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 : カウンタ有効

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 が自動的にクリアされます。

17.4.2 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
予約済み								TI1S	MMS[2:0]				CCDS	予約済み	
								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 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[2:0] : 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 : 有効 - カウンタ有効信号 CNT_EN がトリガ出力 (TRGO) として使用されます。これは、複数のタイマを同時に開始したり、スレーブタイマが有効な時間枠を制御したりするのに便利です。カウンタ有効信号は、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。
 - カウンタ有効信号がトリガ入力によって制御されているときには、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。
 - 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、リセット値に保持する必要があります。



17.4.3 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]		OCCS		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	r/w

ビット 15 **ETP** : 外部トリガ極性

このビットは、ETR と 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、外部クロックモード 1 で TRGI が ETRF に接続された状態 (SMS=111、TS=111) と同じ効果があります。
 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 ETRF です。

ビット 13:12 **ETPS** : 外部トリガ極性

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 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=2
 0010 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=4
 0011 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=8
 0100 : $f_{SAMPLING}=f_{DTS}/2$ 、N=6
 0101 : $f_{SAMPLING}=f_{DTS}/2$ 、N=8
 0110 : $f_{SAMPLING}=f_{DTS}/4$ 、N=6
 0111 : $f_{SAMPLING}=f_{DTS}/4$ 、N=8
 1000 : $f_{SAMPLING}=f_{DTS}/8$ 、N=6
 1001 : $f_{SAMPLING}=f_{DTS}/8$ 、N=8
 1010 : $f_{SAMPLING}=f_{DTS}/16$ 、N=5
 1011 : $f_{SAMPLING}=f_{DTS}/16$ 、N=6
 1100 : $f_{SAMPLING}=f_{DTS}/16$ 、N=8
 1101 : $f_{SAMPLING}=f_{DTS}/32$ 、N=5
 1110 : $f_{SAMPLING}=f_{DTS}/32$ 、N=6
 1111 : $f_{SAMPLING}=f_{DTS}/32$ 、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 : 内部トリガ 2 (ITR2)
 011 : 内部トリガ 3 (ITR3)
 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)

各タイマにとっての ITRx の意味の詳細については、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、使用されていないとき (SMS=000 のときなど) にのみ変更します。

ビット 3 **OCCS** : OCREF クリア選択

このビットを使用して、OCREF クリアソースを選択します。

0 : OCREF_CLR_INT は OCREF_CLR 入力に接続されます。
 1 : OCREF_CLR_INT は ETRF に接続されます。

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)。

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 79. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM2	TIM9	TIM10/[TIM5] ⁽¹⁾	TIM3	TIM4
TIM3	TIM9	TIM2	TIM11/[TIM5] ⁽¹⁾	TIM4
TIM4	TIM10	TIM2	TIM3	TIM9

1. TIM2_OR、TIM3_OR オプションレジスタの ITR1_RMP または ITR2_RMP ビットフィールドで選択できます。

17.4.4 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
予約済み	TDE	予約済み	CC4DE	CC3DE	CC2DE	CC1DE	UDE	予約済み	TIE	予約済み	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、リセット値に保持する必要があります。

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、リセット値に保持する必要があります。

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、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 予約済み、リセット値に保持する必要があります。

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。

17.4.5 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
予約済み			CC4OF	CC3OF	CC2OF	CC1OF	予約済み		TIF	予約済み	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、リセット値に保持する必要があります。

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、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。このビットは、ソフトウェアでクリアされます。
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、リセット値に保持する必要があります。

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、例外状況もあります。TIMx_CR1 レジスタの CMS ビットの説明を参照してください)。このビットは、ソフトウェアでクリアされます。

0 : 一致なし。

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。このビットは、ソフトウェアでクリアされます。

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、オーバーフローまたはアンダーフロー時 (TIM2 から TIM4)。

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)。

17.4.6 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
予約済み									TG	予約済み	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、リセット値に保持する必要があります。

ビット 4 **CC4G**：キャプチャ／比較 4 イベント生成

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 イベント生成

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 イベント生成

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）をとりまします。



17.4.7 TIMx およびキャプチャ/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示します。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可能性がある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 : 凍結 - 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 : 一致時にチャネル 1 をイン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、比較結果が変化するとき、または出力比較モードが凍結モードから 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 : 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 (TIM1_BDTR レジスタの LOCK ビット)、CC1S=00 (チャネルは出力に設定) 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) のときにのみ、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、トリガ入力のイベントから CC 出力への影響の伝播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 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 チャネルは出力として設定されます。

01 : CC1 チャ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ために使用される周波数と、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 : 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 : $f_{SAMPLING} = f_{DTS}/2$, N = 6

0101 : $f_{SAMPLING} = f_{DTS}/2$, N = 8

0110 : $f_{SAMPLING} = f_{DTS}/4$, N = 6

0111 : $f_{SAMPLING} = f_{DTS}/4$, N = 8

1000 : $f_{SAMPLING} = f_{DTS}/8$, N = 6

1001 : $f_{SAMPLING} = f_{DTS}/8$, N = 8

1010 : $f_{SAMPLING} = f_{DTS}/16$, N = 5

1011 : $f_{SAMPLING} = f_{DTS}/16$, N = 6

1100 : $f_{SAMPLING} = f_{DTS}/16$, N = 8

1101 : $f_{SAMPLING} = f_{DTS}/32$, N = 5

1110 : $f_{SAMPLING} = f_{DTS}/32$, N = 6

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。**

17.4.8 TIMx およびキャプチャ／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。**



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 7:4 **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。**

17.4.9 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
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

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、リセット値に保持する必要があります。

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 出カインエーブル
 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、リセット値に保持する必要があります。

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 出カインエーブル
 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、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 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。CCIP の説明を参照してください。
- ビット 2 予約済み、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 アクティブハイです。
1 : OC1 はアクティブローです。
CC1 チャンネルが入力設定のとき :
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 (リセットモード、外部クロックモードまたはトリガモードにおけるキャプチャ、トリガ)、TIxFP1 は反転されません (ゲートモード、エンコーダモードにおけるトリガ)。
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 (リセットモード、外部クロックモードまたはトリガモードにおけるキャプチャ、トリガ)、
TIxFP1 は反転されます (ゲートモード、エンコーダモードにおけるトリガ)。
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 の立ち下がりエッジと立ち下がりエッジの両方に反応し (リセットモード、外部クロックモードまたはトリガモードにおけるキャプチャ、トリガ)、TIxFP1 は反転されません (ゲートモードにおけるトリガ)。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 は有効ではありません。
1 : オン - OC1 信号は、対応する出力ピンに出力されます。
CC1 チャンネルが入力設定のとき :
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 80. 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 IO ピンの状態は、OCx チャンネルの状態と GPIO および AFIO レジスタに依存します。

17.4.10 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

17.4.11 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、 $f_{CK\_PSC} / (PSC[15:0] + 1)$ となります。

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。

17.4.12 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：0xFFFF

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。

ARR の更新と動作の詳細については、[セクション 17.3.1：タイムベースユニット \(397 ページ\)](#) 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。

17.4.13 TIMx 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、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。

CC1 チャンネル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。

17.4.14 TIMx 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

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_CCMR1 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含みます。

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。

17.4.15 TIMx キャプチャ／比較レジスタ 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

チャンネル CC3 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_CCMR2 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 と比較されて、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。

17.4.16 TIMx キャプチャ／比較レジスタ 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

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_CCMR2 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。

17.4.17 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
予約済み		DBL[4:0]					予約済み		DBA[4:0]						
		r/w	r/w	r/w	r/w	r/w			r/w	r/w	r/w	r/w	r/w		

ビット 15:13 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 12:8 **DBL[4:0]**：DMA パースト長

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または書き込みアクセスがあるとパースト転送があると認識します）を定義します。

00000：転送 1回

00001：転送 2回

00010：転送 3回

...

10001：転送 18回

ビット 7:5 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 4:0 **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定義します（TIMx_DMAR アドレスを通じて読み出し/書き込みアクセスがあった場合）。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：次の転送を考えてみましょう。DBL = 転送 7回、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。



17.4.18 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 パ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操作は、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、DBA は TIMx_DCR レジスタで設定される 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る 0 から DBL(DBL は TIMx_DCR レジスタで設定される) の範囲の値です。

DMA パースト機能の使用例

この例では、タイマ DMA パ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)。この時、DMA で CCRx レジスタへハーフワード転送します。

これを次のステップで実行します。

- 対応する DMA チャンネルを次のように設定します。
 - DMA チャンネルのペリフェラルアドレスは DMAR レジスタのアドレスです
 - DMA チャンネルのメモリアドレスは、DMA から CCRx レジスタに転送されるデータが含まれている RAM のバッファのアドレスです。
 - 転送するデータの数 = 3 (下記注を参照)
 - サーキュラモードは無効です。
- DBA および DBL ビットフィールドを次のように設定して DCR レジスタを設定します。
DBL = 転送 3 回、DBA = 0xE。
- 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)。
- TIMx を有効にします。
- DMA チャンネルを有効にします。

注 : この例は、すべての CCRx レジスタを 1 回更新する場合です。たとえばすべての CCRx レジスタを 2 回更新する必要がある場合は、転送するデータの数に 6 にします。data1、data2、data3、data4、data5、data6 を含む RAM のバッファの例を見てみましょう。データは次のように CCRx レジスタに転送されます。最初の更新 DMA リクエストで、data1 が CCR2 に、data2 が CCR3 に、data3 が CCR4 に転送され、2 回目の更新 DMA リクエストで data4 が CCR2 に、data5 が CCR3 に、data6 が CCR4 に転送されます。



17.4.19 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

このレジスタは、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み															ITR1_RMP
															rw

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 ITR1_RMP : タイマ 2 内部トリガ 1 の再割り付け

ソフトウェアによってセット/クリアされます。

0 : TIM2 ITR1 入力 が TIM10 OC に接続されます。

1 : TIM2 ITR1 入力 が TIM5 TGO に接続されます。

17.4.20 TIM3 オプションレジスタ (TIM3_OR)

アドレスオフセット : 0x50

リセット値 : 0x0000

このレジスタは、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み															ITR2_RMP
															rw

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 ITR2_RMP : タイマ 3 内部トリガ 2 の再配置

ソフトウェアによってセット/クリアされます。

0 : TIM3 ITR2 入力 が TIM11 OC に接続されます。

1 : TIM3 ITR2 入力 が TIM5 TGO に接続されます。

17.4.21 TIMx レジスタマップ

TIMx レジスタは、次の表のようにマッピングされます。

表 81. TIMx 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	予約済み																						CKD [1:0]		ARPE	CMS [1:0]		DIR	OPM	URS	UDIS	CEN	
	0																							0	0	0	0	0	0	0	0	0		
0x04	TIMx_CR2	予約済み																						T11S	MMS[2:0]		CCDS	予約済み						
	0																							0	0	0	0							
0x08	TIMx_SMCR	予約済み																	ETP	ECE	ETPS [1:0]		ETF[3:0]		MSM	TS[2:0]		予約済み	SMS[2:0]					
	0																		0	0	0	0	0	0	0	0	0					0	0	0
0x0C	TIMx_DIER	予約済み																	TDE	COMDE	CC4DE	CC3DE	CC2DE	CC1DE	UDE	予約済み	TIE	予約済み		CC4IE	CC3IE	CC2IE	CC1IE	UIE
	0																		0	0	0	0	0	0	0		0	0	0	0	0	0	0	0
0x10	TIMx_SR	予約済み																			CC4OF	CC3OF	CC2OF	CC1OF	予約済み	TIF	予約済み	CC4IF	CC3IF	CC2IF	CC1IF	UIF		
	0																				0	0	0	0		0		0	0	0	0	0	0	0
0x14	TIMx_EGR	予約済み																									TG	予約済み	CC4G	CC3G	CC2G	CC1G	UG	
	0																										0		0	0	0	0	0	
0x18	TIMx_CCMR1 出力比較モード	予約済み																	OC2CE	OC2M [2:0]		OC2PE	OC2FE	CC2S [1:0]	OC1CE	OC1M [2:0]		OC1PE	OC1FE	CC1S [1:0]				
	0																		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	予約済み																	IC2F[3:0]			IC2 PSC [1:0]	CC2S [1:0]	IC1F[3:0]			IC1 PSC [1:0]	CC1S [1:0]						
	0																		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	予約済み																	O24CE	OC4M [2:0]		OC4PE	OC4FE	CC4S [1:0]	OC3CE	OC3M [2:0]		OC3PE	OC3FE	CC3S [1:0]				
	0																		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	予約済み																	IC4F[3:0]			IC4 PSC [1:0]	CC4S [1:0]	IC3F[3:0]			IC3 PSC [1:0]	CC3S [1:0]						
	0																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	TIMx_CCER	予約済み																	CC4NP	予約済み	CC4P	CC4E	CC3NP	予約済み	CC3P	CC3E	CC2NP	予約済み	CC2P	CC2E	CC1NP	予約済み	CC1P	CC1E
	0																		0		0	0	0		0	0	0		0	0	0		0	0
0x24	TIMx_CNT	CNT[32:16]（TIM5 のみ、他のタイマでは予約済み）																	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	
0x28	TIMx_PSC	予約済み																	PSC[15:0]															
	0																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	TIMx_ARR	ARR[32:16]（TIM5 のみ、他のタイマでは予約済み）																	ARR[15:0]															
	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	

表 81. TIMx 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	
0x30	予約済み																																	
0x34	TIMx_CCR1	CCR1[32:16]（TIM5 のみ、他のタイマでは予約済み）																CCR1[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		
0x38	TIMx_CCR2	CCR2[32:16]（TIM5 のみ、他のタイマでは予約済み）																CCR2[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		
0x3C	TIMx_CCR3	CCR4[32:16]（TIM5 のみ、他のタイマでは予約済み）																CCR3[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		
0x40	TIMx_CCR4	CCR4[32:16]（TIM5 のみ、他のタイマでは予約済み）																CCR4[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		
0x44	予約済み																																	
0x48	TIMx_DCR	予約済み																DBL[4:0]				手 続 扱 い	DBA[4:0]											
	リセット値																	0	0	0	0		0	0	0	0	0	0	0					
0x4C	TIMx_DMAR	予約済み																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	予約済み																														ITR1_RMP		
	リセット値																																0	
0x50	TIM3_OR	予約済み																														ITR2_RMP		
	リセット値																																0	

レジスタ境界アドレスについては、表 5 (47 ページ) 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。

18 汎用タイマ (TIM9/10/11)

18.1 TIM9/10/11の概要

TIM9/10/11 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、入力信号パルス幅の測定（入力キャプチャ）または出力波形の発生（出力コンペア、PWM）など様々な目的に使用することができます。

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。

TIM9/10/11 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ていません。これらのカウンタは [セクション 18.3.12](#) に示すように、相互に同期させることができます。

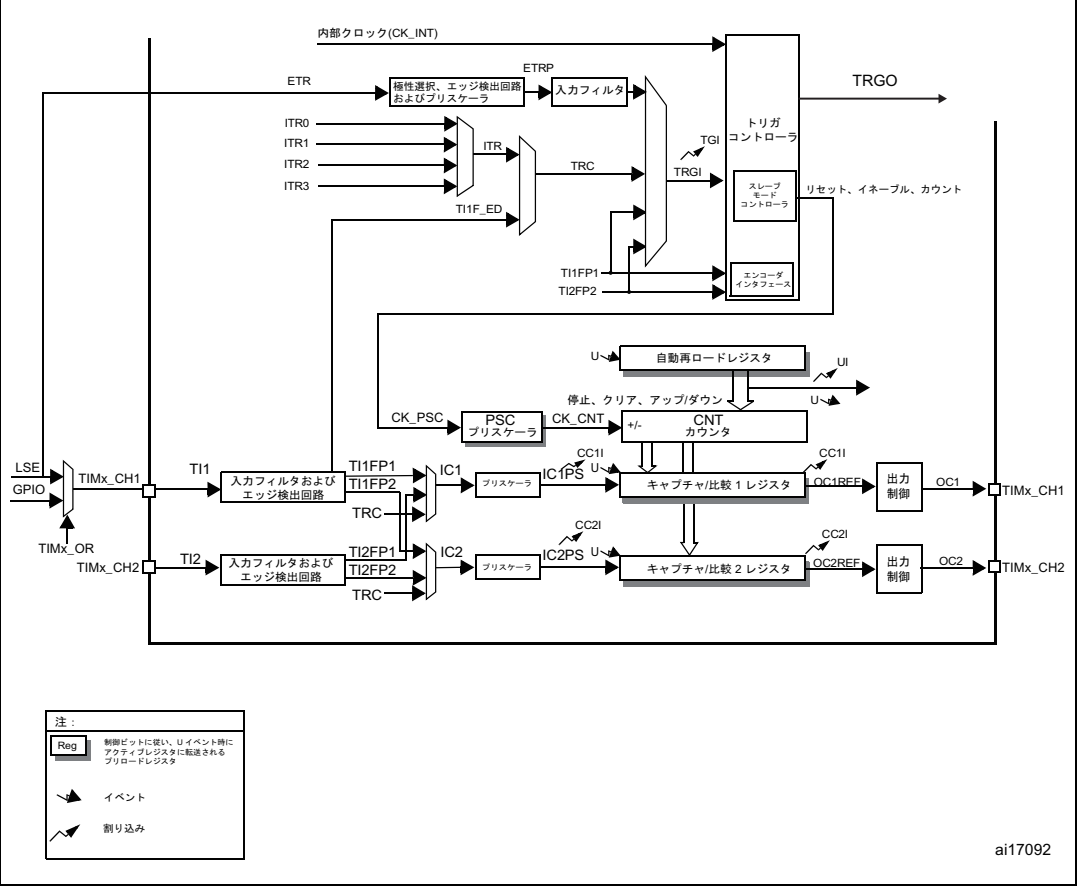
18.2 TIM9/10/11 の主な機能

18.2.1 TIM9 の主な機能

汎用 タイマの機能には、次が含まれます。

- 16 ビット自動再ロードアップカウンタ (Cat.1 と Cat.2 デバイス)
- 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 (Cat.3、Cat.4、Cat.5 デバイス)
- カウンタクロック周波数を 1 から 65536 分周する 16 ビットのプログラマブルプリスケラ (即座に変更可能)。
- 次の用途に使用する独立な最大 2 チャンネル
 - － 入力キャプチャ
 - － 出力比較
 - － PWM 生成 (エッジアラインモード)
 - － 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から）
 - － 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
- 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 130. 汎用タイマのブロック図 (TIM9)



18.2.2 TIM10/TIM11の主な機能

汎用タイマ TIM10/TIM11の機能には次が含まれます。

- 16 ビット 自動再ロードアップカウンタ
- カウンタクロック周波数を 1 から 65536 分周する 16 ビットのプログラマブルプリスケアラ(即座に変更可能)。
- 次の機能を持つ独立チャネル：
 - － 入力キャプチャ
 - － 出力比較
 - － PWM 生成 (エッジアラインモード)
- 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、カウンタの初期化 (ソフトウェアによる)
 - － 入力キャプチャ
 - － 出力比較

図 131. 汎用タイマのブロック図 (TIM10)

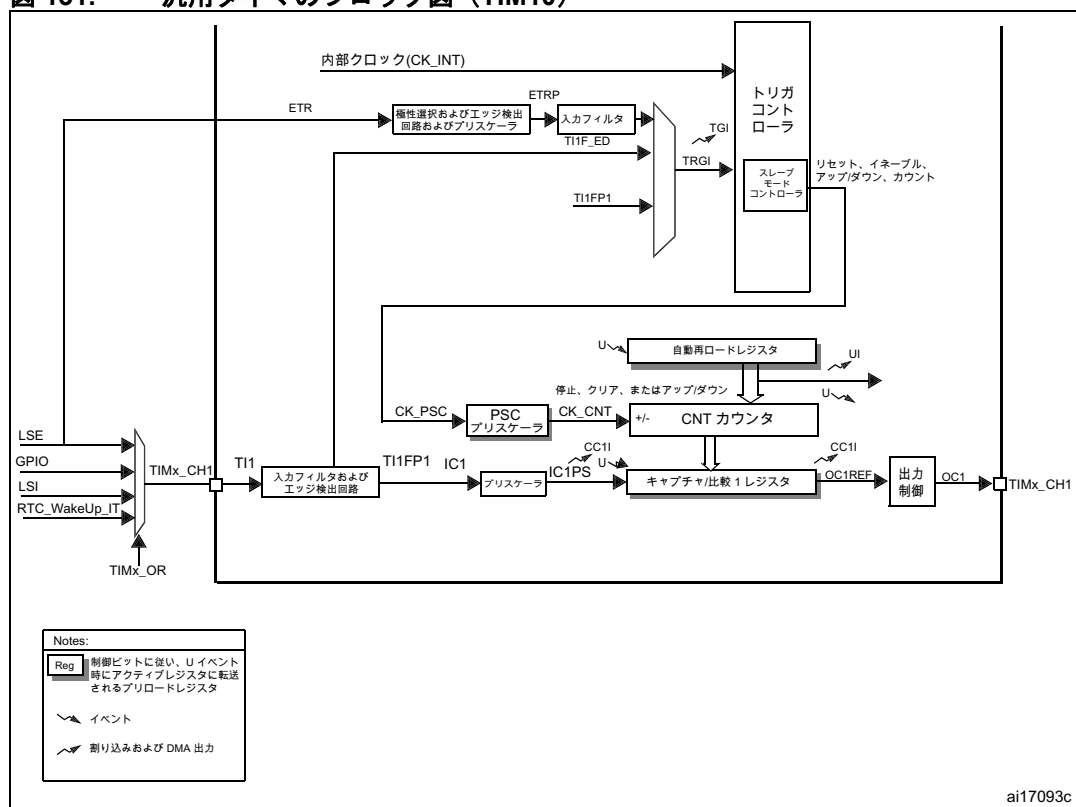
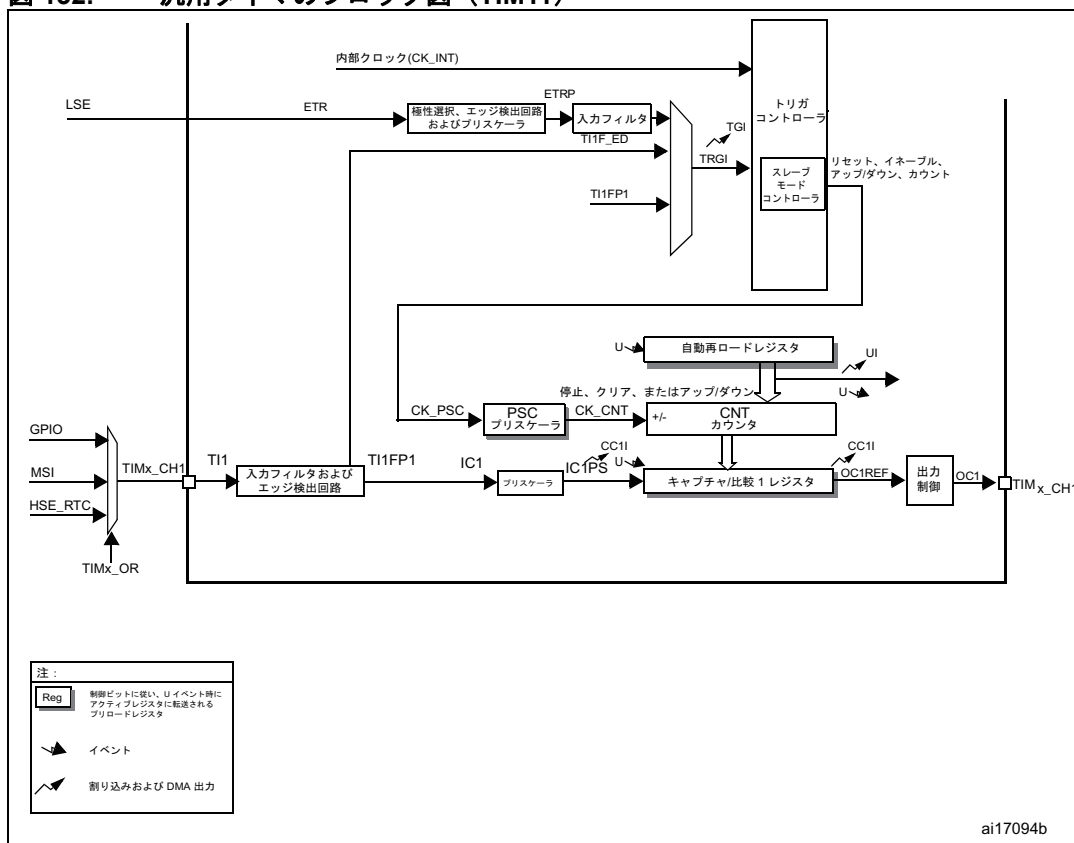


図 132. 汎用タイマのブロック図 (TIM11)



18.3 TIM9/10/11の機能説明

18.3.1 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。

- Cat.3、Cat.4、Cat.5 デバイス : TIM9 カウンタはカウントアップ、ダウンまたはカウントアップとダウンが可能です。TIM10/11 カウンタは、アップカウンティングモードでのみ動作します。
- Cat.1、Cat.2 デバイス : TIM9/10/11 カウンタはアップカウンティングモードでのみ動作します。

カウンタクロックは、プリスケアラで分周することができます。

カウンタ、自動再ロードレジスタ、プリスケアラレジスタは、ソフトウェアから読み書き可能です。カウンタの動作中も同様です。

タイムベースユニットには、次が含まれます。

- 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 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 のプリスケアラ比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。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。新しいプリスケアラ比は、次の更新イベントで有効になります。

[図 133](#) と [図 134](#) は、プリスケアラ比がその場で変更されたときのカウンタの動作の例を示します。

図 133. プリスケーラプリスケーラ比が 1 から 2 に変化したときのカウンタのタイミング図

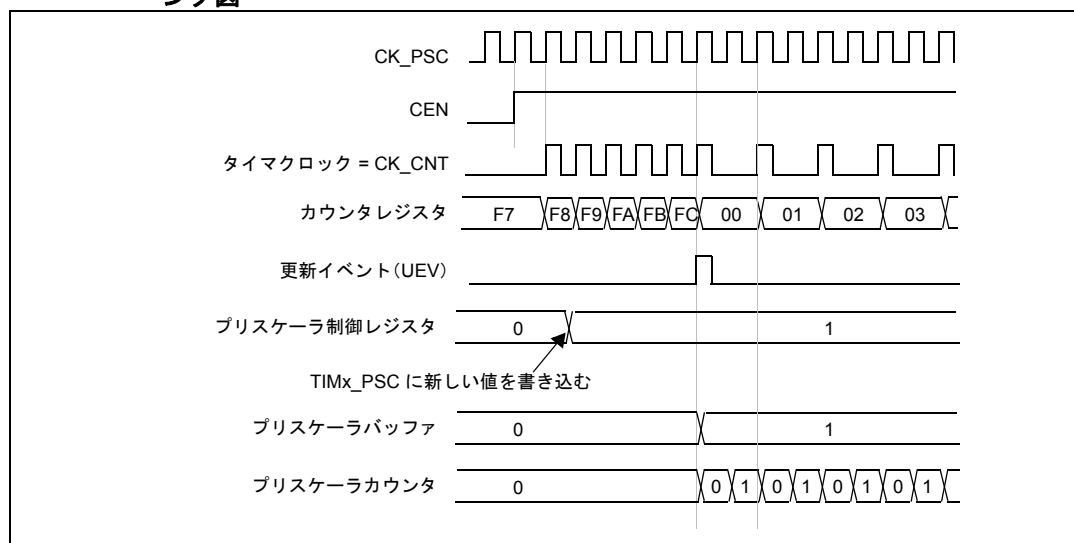
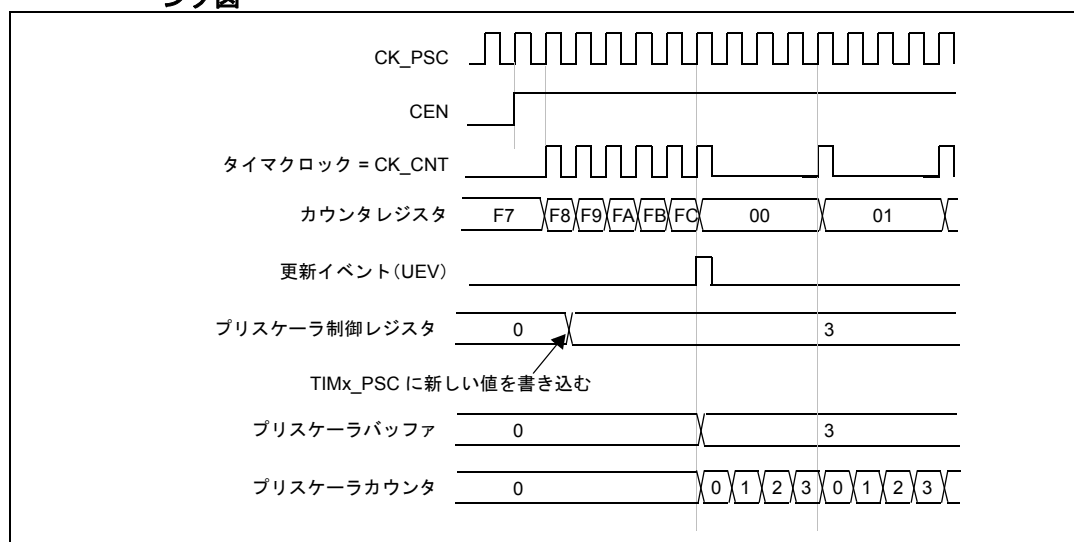


図 134. プリスケーラプリスケーラ比が 1 から 4 に変化したときのカウンタのタイミング図



18.3.2 カウンタ・モード

Cat.3、Cat.4、Cat.5 デバイスでは、TIM9 はダウンカウンティングモードとセンターアラインドモードで動作することができます。このモードの説明については [セクション 17.3.2 \(399 ページ\)](#) を参照してください。

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。

TIMx_EGR レジスタの UG ビットをセットした場合も (ソフトウェアによって、または TIM9 のスレーブモードコントローラを使用して)、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、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。

- 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) が再びロードされます。

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 135. カウンタタイミング図、内部クロック 1 分周

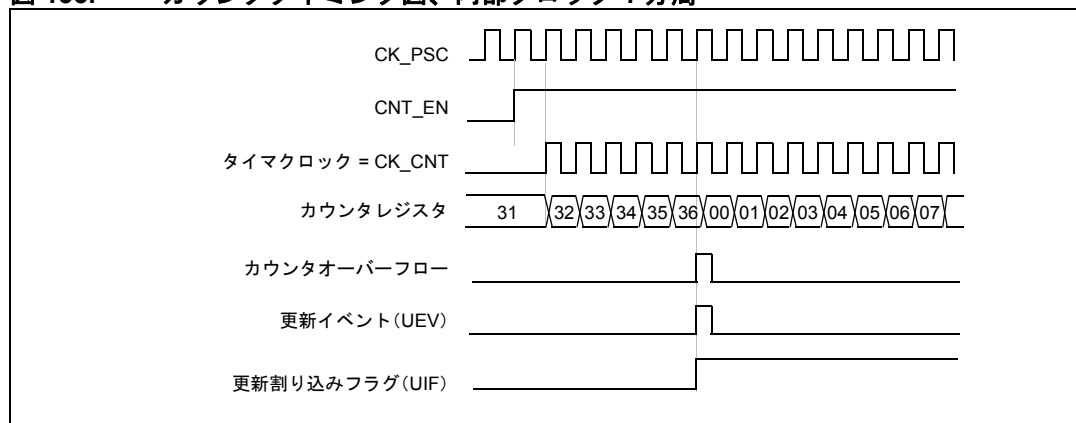


図 136. カウンタタイミング図、内部クロック 2 分周

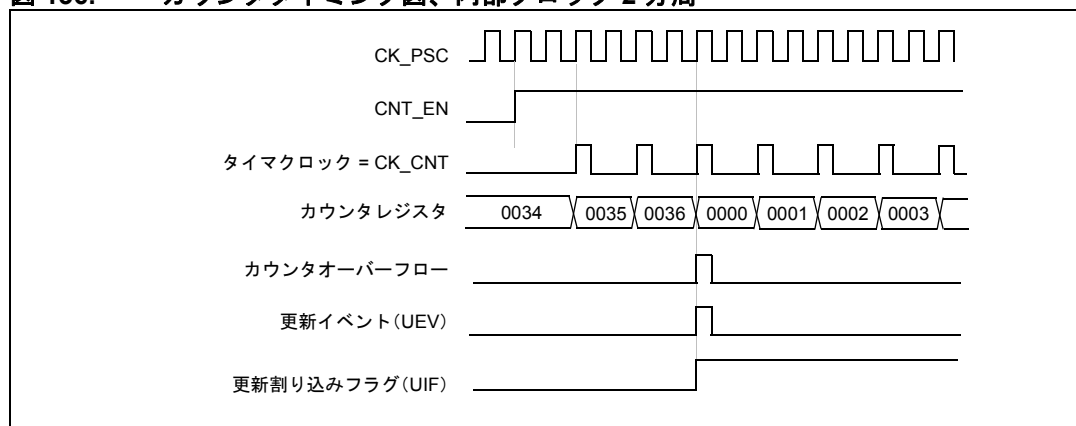


図 137. カウンタタイミング図、内部クロック 4 分周

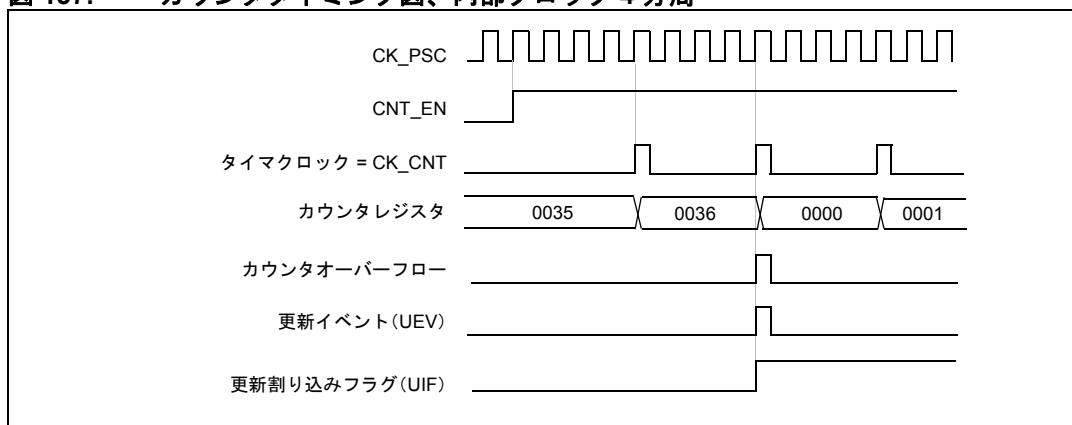


図 138. 内部クロックプリスケアラ比が N の場合のカウンタのタイミング図

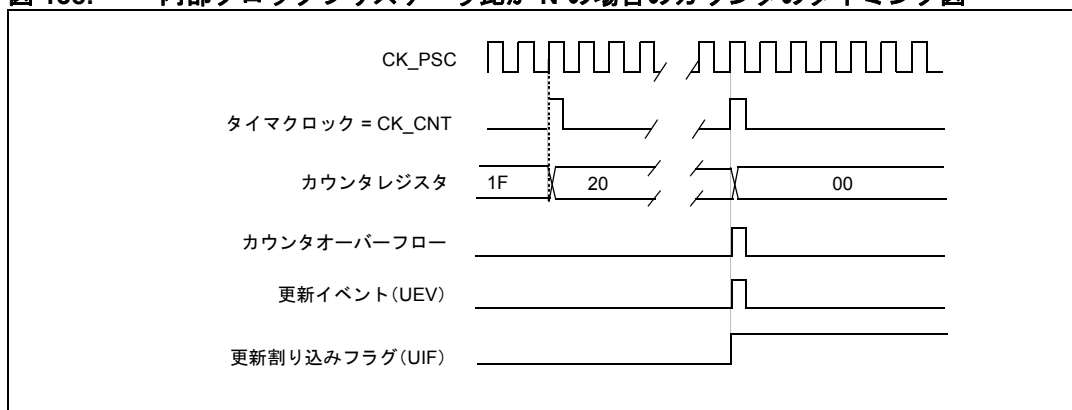


図 139. ARPE=0 時 (TIMx_ARR はプリロードなし)、更新イベント時のカウンタのタイミング図

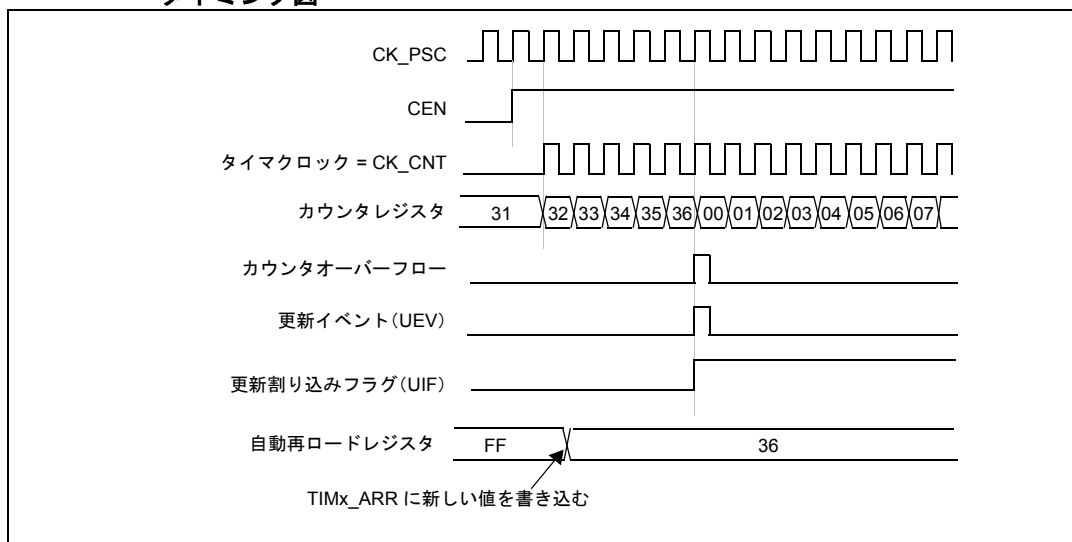
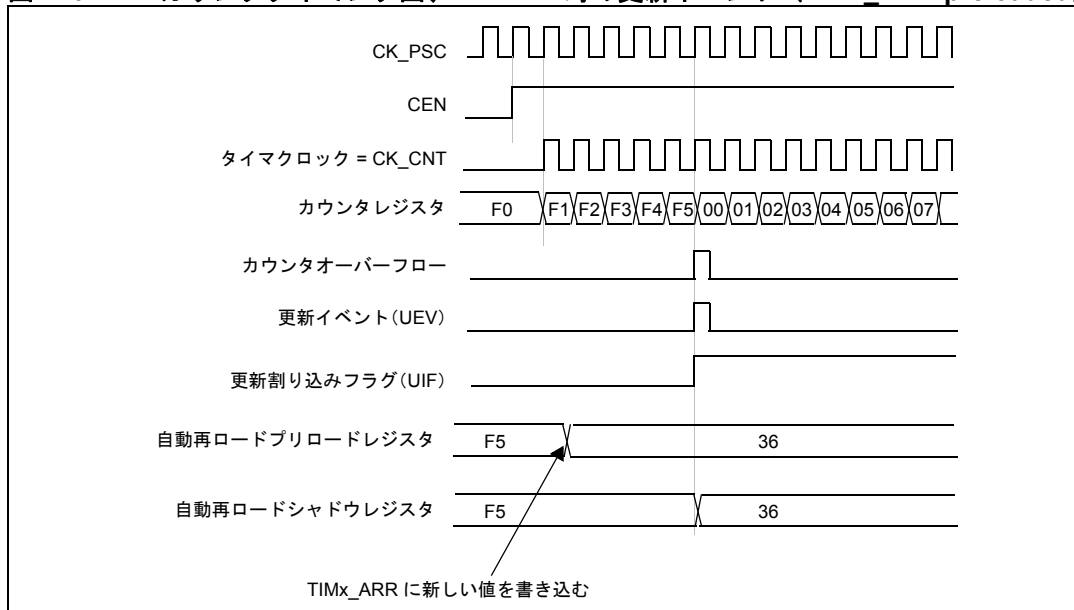


図 140. カウンタタイミング図、ARPE=1 時の更新イベント (TIMx_ARR preloaded)



18.3.3 クロック選択

カウンタクロックは、次のクロックソースから供給できます。

- 内部クロック (CK_INT)
- 外部クロックモード 1 (TIM9) : 外部入力ピン (Tlx)
- 外部クロックモード 2 : 外部トリガ入力 (内部で ETR と LSE を接続)
- 内部トリガ入力 (ITRx) (TIM9の場合) : 別のタイマからのトリガ出力を接続詳細については、[セクション : 1 つのタイマを別のタイマのプリスケラとして使用する](#)を参照してください。

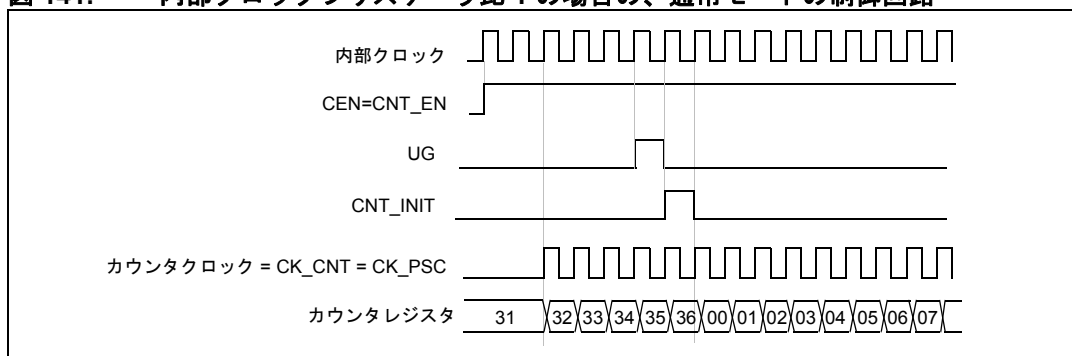
内部クロックソース (CK_INT)

内部クロックソースは、TIM10/TIM11 のデフォルトクロックソースです。

TIM9 と TIM12 の場合、スレーブモードコントローラが無効化されたとき (SMS='000')、内部クロックソースが選択されます。TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットが制御ビットとして使用され、ソフトウェアからのみ変更できます (クリアされたままの UG は除きます)。CEN ビットに 1 が設定されると直ちに、プリスケラは内部クロック CK_INT で駆動されます。

[図 141](#)に、プリスケ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。

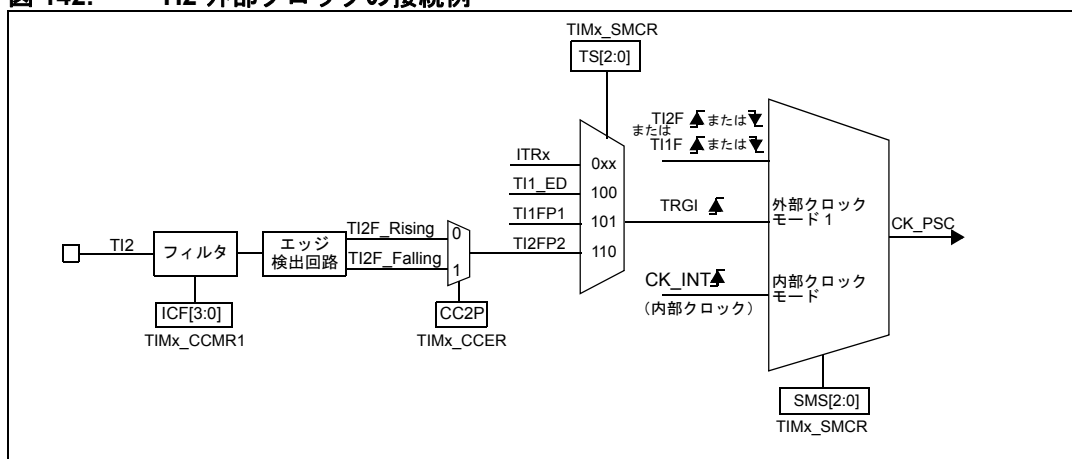
図 141. 内部クロックプリスケアラ比 1 の場合の、通常モードの制御回路



外部クロックソースモード 1 (TIM9)

このモードは TIMx_SMCR レジスタで 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 142. TI2 外部クロックの接続例



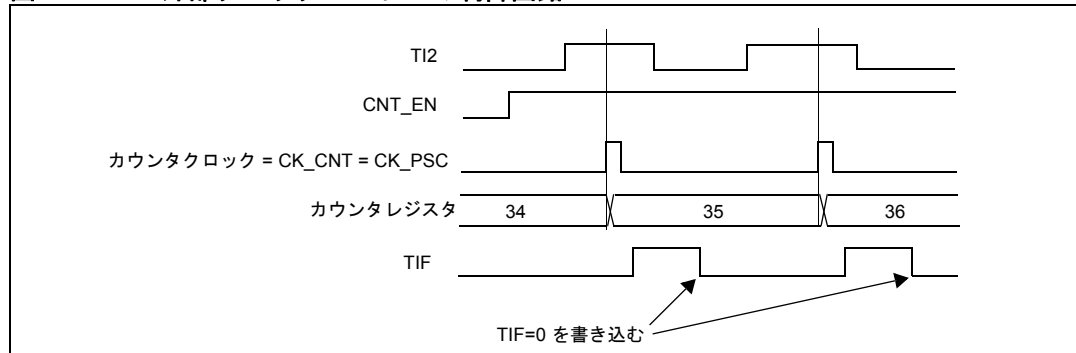
たとえば、TI2 入力の立ち上がりエッジに反応してカウントするようにアップカウンタを設定するには、次の手順で行います。

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 ICF[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. 立ち上がりエッジ極性を選択するときは、CC2P=0 と CC2NP=0 を TIMx_CCER レジスタに書き込みます。
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、トリガ入力ソースとして TI2 を選択します。
6. TIMx_CR1 レジスタに CEN=1 を書き込んで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ので、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 143. 外部クロックモード 1 の制御回路



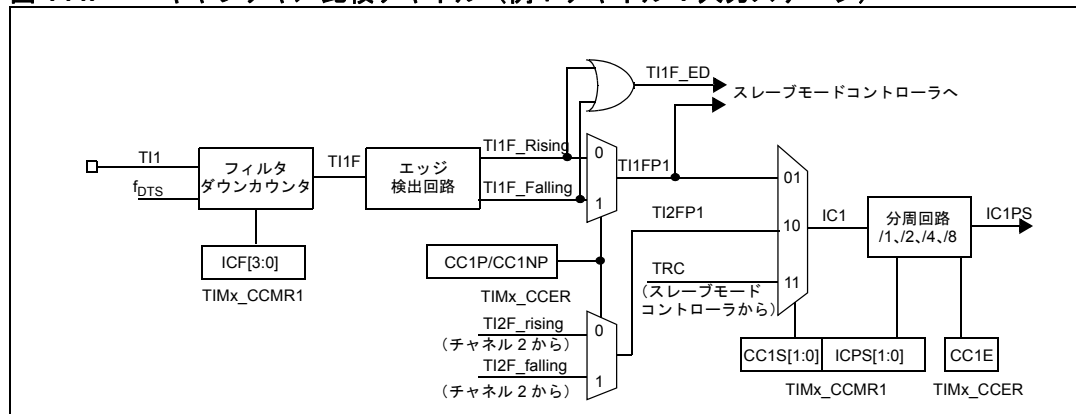
18.3.4 キャプチャ／比較チャネル

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、プリスケータ）、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。

図 144 から 図 146 に、1 つのキャプチャ／比較チャネルの概要を示します。

入力カステージは、対応する TIx 入力をサンプルして、フィルタリングを行った TIxF 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できる信号 (TIxFPx) を生成します。この信号はプリスケータを通じて、キャプチャレジスタ (ICxPS) に渡されます。

図 144. キャプチャ／比較チャネル（例：チャネル 1 入力カステージ）



出力カステージは、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 145. キャプチャ/比較チャネル 1 メイン回路

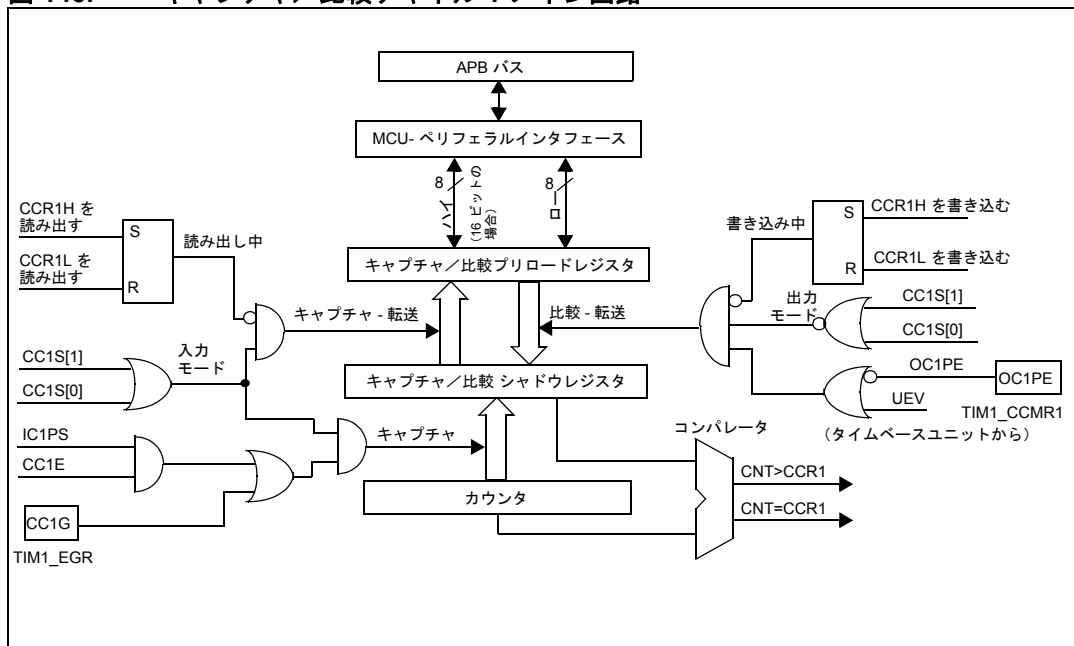
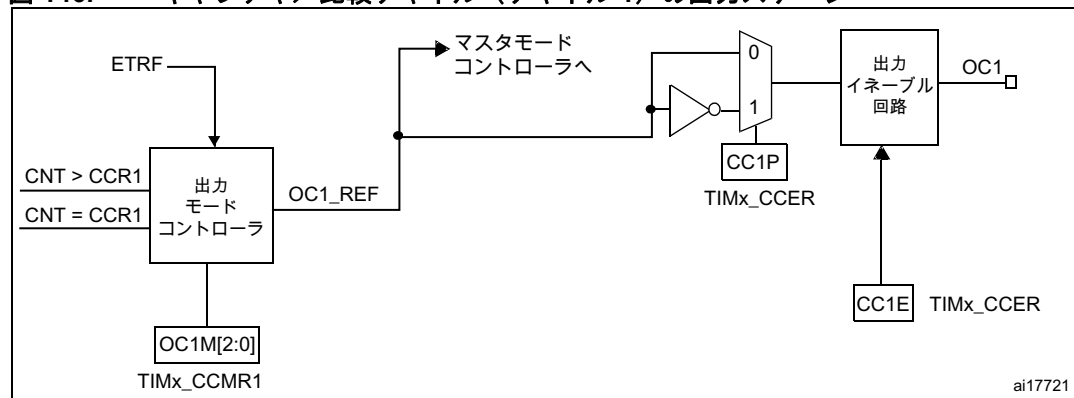


図 146. キャプチャ/比較チャネル (チャネル 1) の出力ステージ



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。読み書きのアクセスは、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。

18.3.5 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。このためには、次の手順を使用します。

1. 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てください (入力が TIx 入力の内の 1 つである場合、TIMx_CCMRx レジスタの ICxF ビットを設定して行います)。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 で変化を検証できます (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するには、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、立ち上がりエッジを選択します)。
4. 入力プリスケアラをプログラムします。この例では有効な信号変化ごとにキャプチャを行うので、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. 必要に応じて、TIMx_BDTR レジスタの CC1IE ビットをセットすることで、ブレーク機能を有効にすることができます。

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: 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。

18.3.6 PWM 入力モード (TIM9 のみ)

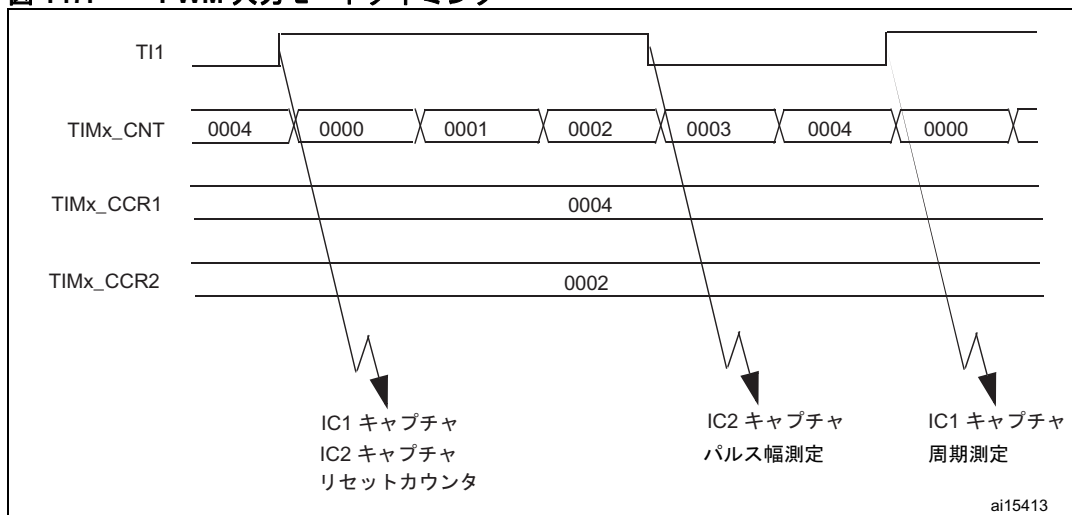
このモードは、入力キャプチャモードの特殊ケースです。操作手順は入力キャプチャモードと同様ですが、以下の点が異なります。

- 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、反対極性のエッジ上アクティブです。
- 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 周波数とプリスケール値によって、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. TI1FP1 のアクティブ極性を選択します (TIMx_CCR1 キャプチャとカウンタクリアの両方に使用します)。CC1P ビットと CC1NP ビットに "00" を書き込みます (立ち上がりエッジでアクティブ)。
3. TIMx_CCR1 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_CCMR2 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. キャプチャを有効にします。すなわち、TIMx_CCER レジスタの CC1E ビットと CC2E ビットに "1" を書き込みます。

図 147. PWM 入力モードタイミング



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。

18.3.7 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) はソフトウェアによって強制的にアクティブ、イン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_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。

たとえば、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。

18.3.8 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マブル値を割り当てます。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、TIMx_OCMRx レジスタの OCxPE ビットを使用して、プリロードレジスタを使用して、またはプリロードレジスタを使用せずに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することができます (ワンパルスモード)。

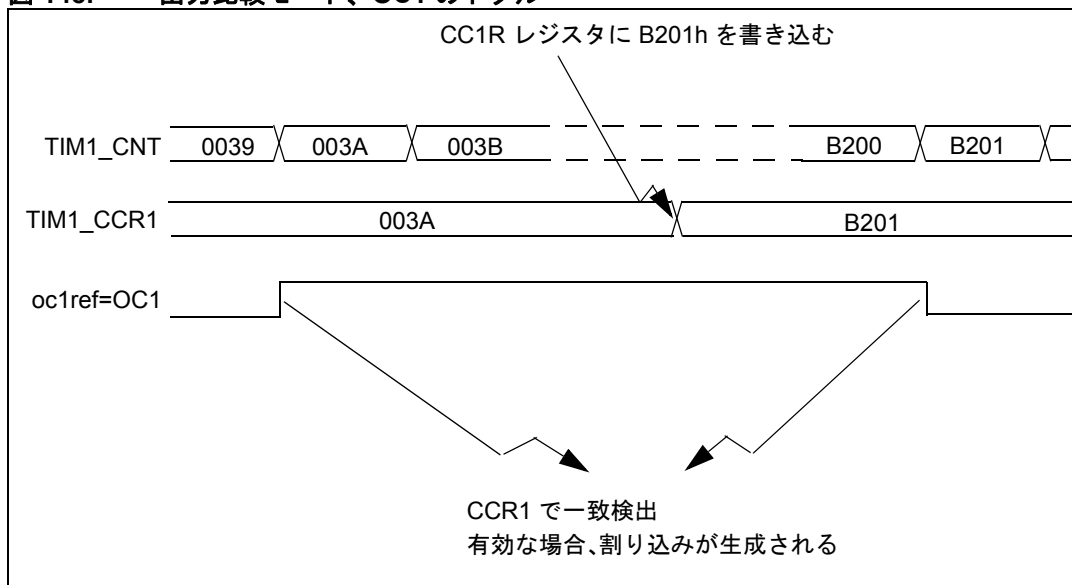
手順：

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 = 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 = 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 = 0 を書き込みます。
 - 出力を有効にするには、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 148](#) に示します。

図 148. 出力比較モード、OC1 のトグル



18.3.9 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 ビットに "110" (PWM モード 1) または "111" (PWM モード 2) 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)。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できます。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、 $TIMx\_CNT \leq TIMx\_CCRx$ かどうか判断されます。

ただし、ETRF (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。
- 出力比較モード (TIMx_CCMRx レジスタの OCxM ビット) が固定構成 (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。

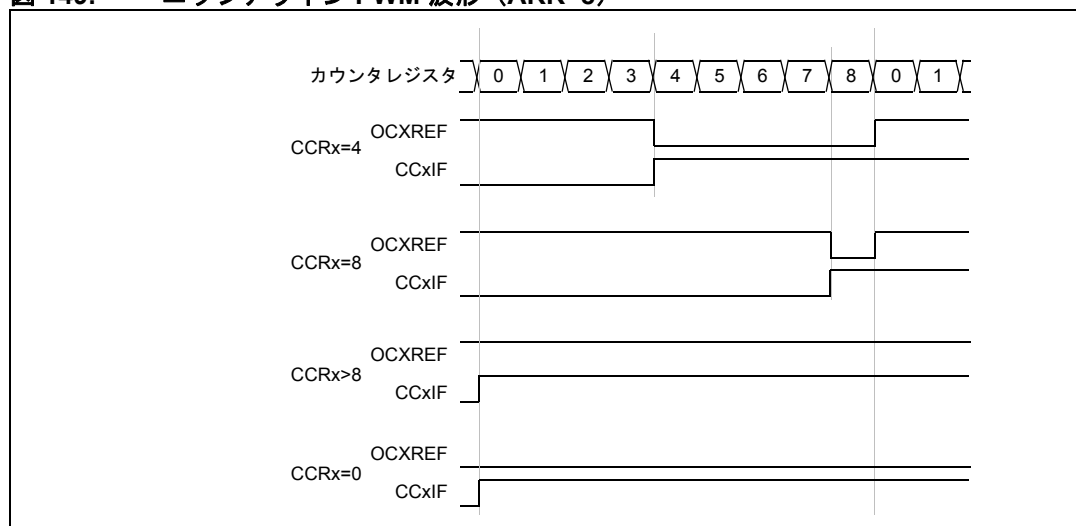
タイマの動作中は、ソフトウェアによって強制的に PWM になります。

カウンタはカウントアップしているの、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。

PWM エッジアラインモード

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、OCiREF は "0" に保持されます。図 149 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 149. エッジアライン PWM 波形 (ARR=8)



18.3.10 ワンパルスモード

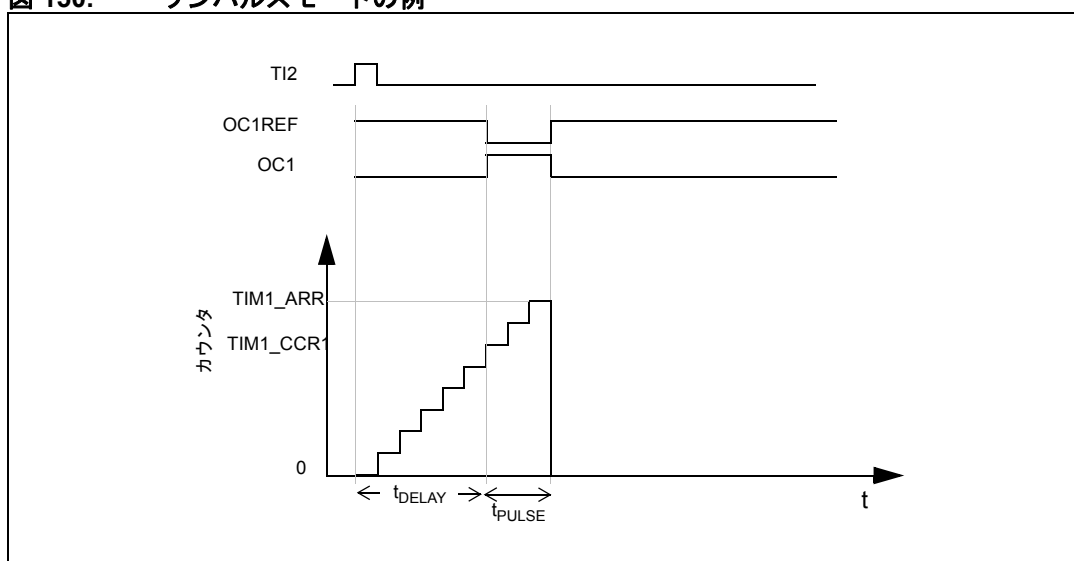
ワンパルスモード (OPM : One Pulse Mode) は、これまでに説明したモードの特殊ケースです。トリガに応じてカウンタを開始して、プログラム可能な遅延後にプログラム可能な長さのパルスを生成できます。

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。

$$\text{CNT} < \text{CCR}x \leq \text{ARR} \quad (\text{特に、} 0 < \text{CCR}x)$$

図 150. 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。

1. TIMx_CCMR1 レジスタに CC2S=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 レジスタで CC2P=0 と CC2NP=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=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 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 レジスタで 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

スタに OC1PE="1" を書き込み、TIMx_CR1 レジスタに ARPE 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。

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 ($t_{\text{DELAY min}}$) 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。

18.3.11 TIM9 外部トリガ同期

TIM9 タイマは、リセットモード、ゲートモード、トリガモードなど、いくつかのモードで外部トリガと同期することができます。

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。

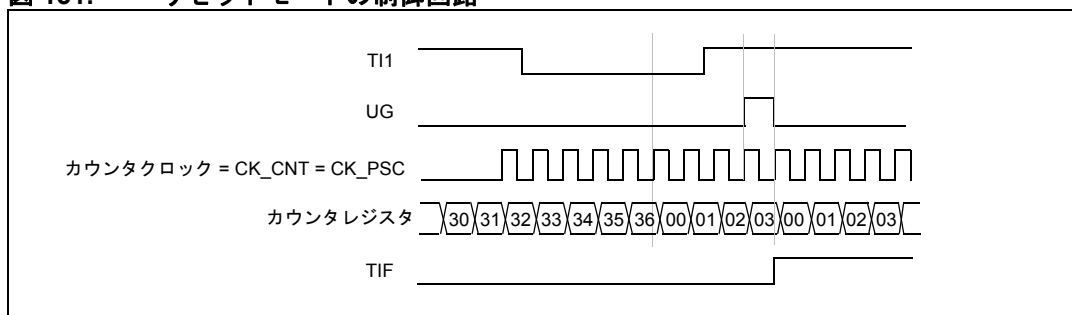
次の例では、TI1 入力の立ち上がりエッジに応じて、アップカウンタがクリアされます。

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 と CC1NP に "00" 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)。
2. 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。
3. TIMx_CR1 レジスタに CEN=1 を書き込んで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を送信できます (TIMx_DIER レジスタの TIE ビットに応じて)。

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 151. 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。

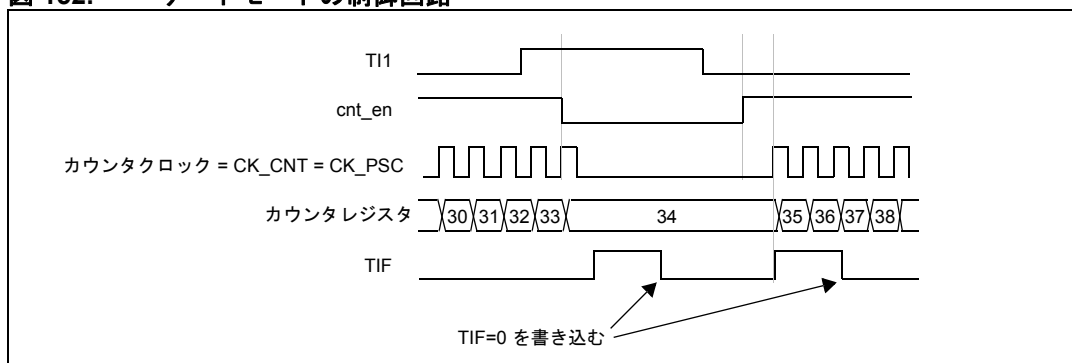
次の例では、アップカウンタは TI1 入力があるときだけカウントします。

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）。TIMx_CCER レジスタで CC1P = 1 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）。
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 レジスタの 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 がローである限り内部クロックでカウントを開始し、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 152. ゲートモードの制御回路



スレーブ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、カウンタが開始できます。

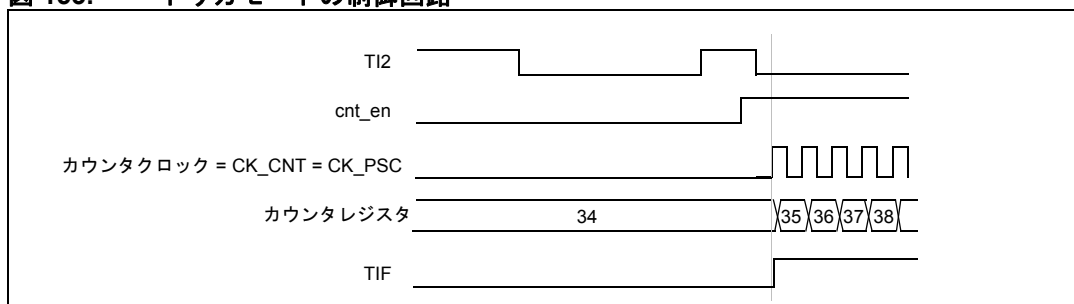
次の例では、アップカウンタは、TI2 入力の立ち上がりエッジに応じて開始します。

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アラはトリガには使用されないなので、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 レジスタで CC2P = 1 と CC2NP = 0 を書き込んで、極性を有効にします（そして、ロー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 153. トリガモードの制御回路



18.3.12 タイマ同期 (TIM9)

TIM タイマは、タイマの同期または連結のために、内部で互いに接続されます。詳細については、[セクション 17.3.15 : タイマの同期 \(426 ページ\)](#) を参照してください。

18.3.13 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®-M3 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 30.16.2 : 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。

18.3.14 エンコーダインタフェースモード (TIM9 の場合)

このセクションは Cat.3、Cat.4、Cat.5 デバイスにのみ適用されます。

[セクション 17.3.12 : エンコーダインタフェースモード](#) を参照してください。

18.4 TIM9 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。

18.4.1 TIM9 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
予約済み						CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
						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プリスケアラ比を示します。

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[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ンネル (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 (TIMx_CCMRx レジスタの CCxS=00) 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ンネル (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。

注 : これらのビットは、Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能です。

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。

注 : Cat.1 と Cat.2 デバイスでは、このビットは予約済みで、リセット値に維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 ビットをクリアします)。

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を発生します (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。UEV イベントは、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

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。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。

18.4.2 TIM9 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
予約済み									MMS[2:0]			予約済み			
									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[2:0]** : マスタモード選択

これらのビットは、スレーブタイマの同期のためにマスタモードで送られる情報 (TRGO) を選択するために使用されます。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、複数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ウィンドウ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生ずると同時に、CC1IF フラグがセットされる (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。



18.4.3 TIM9 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]			予約済み	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 と ETR のいずれをトリガ動作に使用するかを選択します。
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有効になります。
1 : ETR は反転され、ローレベルまたは立ち下がりエッジで有効になります。

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、外部クロックモード 1 で TRGI が ETRF に接続された状態 (SMS=111、TS=111) と同じ効果があります。
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 である必要があります。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} で実効
0001 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=2
0010 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=4
0011 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=8²
0100 : $f_{SAMPLING}=f_{DTS}/2$ 、N=6
0101 : $f_{SAMPLING}=f_{DTS}/2$ 、N=8
0110 : $f_{SAMPLING}=f_{DTS}/4$ 、N=6²
0111 : $f_{SAMPLING}=f_{DTS}/4$ 、N=8
1000 : $f_{SAMPLING}=f_{DTS}/8$ 、N=6
1001 : $f_{SAMPLING}=f_{DTS}/8$ 、N=8
1010 : $f_{SAMPLING}=f_{DTS}/16$ 、N=5
1011 : $f_{SAMPLING}=f_{DTS}/16$ 、N=6
1100 : $f_{SAMPLING}=f_{DTS}/16$ 、N=8
1101 : $f_{SAMPLING}=f_{DTS}/32$ 、N=5
1110 : $f_{SAMPLING}=f_{DTS}/32$ 、N=6
1111 : $f_{SAMPLING}=f_{DTS}/32$ 、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する場合に役立ちます。



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。

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 : 内部トリガ 2 (ITR2)
- 011 : 内部トリガ 3 (ITR3)
- 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

各タイマの ITRx の意味については、表 82 : TIMx 内部トリガ接続 (481 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、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、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。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 - TI1FP1 レベルに応じて TI2FP2 エッジでアップ/ダウンカウント (このモードは Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能)。
- 010 : エンコーダモード 2 - TI2FP2 レベルに応じて TI1FP1 エッジでアップ/ダウンカウント (このモードは Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能)。
- 011 : エンコーダモード 3 - 他の入力のレベルに応じて、TI1FP1 と TI2FP2 の両方のエッジでアップ/ダウンカウント (このモードは Cat.3、Cat.4、Cat.5 デバイスでのみ使用可能)。
- 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、トリガ入力 (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 82. TIMx 内部トリガ接続

スレーブ TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM9	TIM2	TIM3	TIM10	TIM11



18.4.4 TIM9 割り込みイネーブルレジスタ (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
予約済み									TIE	予約済み			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。



18.4.5 TIM9 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
予約済み					CC2OF	CC1OF	予約済み		TIF	予約済み			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、対応するチャン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。このビットは、ソフトウェアでクリアされます。
0 : トリガイイベントは発生していません。
1 : トリガ割り込みがペンディング中です。

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。このビットは、ソフトウェアでクリアされます。

0 : 一致なし。

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 が入力として構成されている場合 :

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。このビットは、ソフトウェアでクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みがペンディング中です。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 レジスタの URS=0 かつ UDIS=0 であり、トリガイベントによって 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)。

18.4.6 TIM イベント発生レジスタ (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
予約済み									TG	予約済み			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、関連する割り込みが発生します。

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ソフトウェアでセットされ、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、ソフトウェアでセットでき、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)。カウンタはクリアされます。

18.4.7 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
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 の有効レベルは、それぞれ CC1P および CC1NP ビットに依存します。

000 : 凍結 - 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、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 の場合は無効 (OC1REF="0") に、そうでない場合は有効 (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、比較結果が変化するとき、または出力比較モードが凍結モードから 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有効にしなくても使用できます。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、トリガ入力のイベントから CC 出力への影響の伝播を加速す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) で内部トリガ入力が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 : 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 : $f_{SAMPLING}=f_{DTS}/2$, 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

注： **ICx[F3:0]=1、2 または 3 のとき、式内で f_{DTS} が CK_INT で置き換えられることに注意してください。**

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。**

18.4.8 TIM9 キャプチャ／比較イネーブルレジスタ (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
予約済み								CC2NP	予約済み	CC2P	CC2E	CC1NP	予約済み	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 チャンネルを出力として設定 : CC1NP はクリアされたままにする必要があります。
CC1 チャンネルを入力として設定 : 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 の説明を参照してください)。

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、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 83. 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。

18.4.9 TIM9 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

18.4.10 TIM9 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、 $f_{CK\_PSC} / (PSC[15:0] + 1)$ に一致します。

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。

18.4.11 TIM9 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：0xFFFF

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。

ARR 更新動作の詳細については、[セクション 18.3.1：タイムベースユニット \(459 ページ\)](#) 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。

18.4.12 TIM9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、この値がロードされ固定されます。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。

CC1 チャンネル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。

18.4.13 TIM9キャプチャ／比較レジスタ 2 (TIMx_CCR1)

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

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、この値がロードされ固定されます。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。

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。

18.4.14 TIM9 オプションレジスタ 1 (TIM9_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
予約済み													ITR1_RMP	TIM9_OR[1:0]	
													rw	rw	

ビット 15:3 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 ITR1_RMP タイマ 9 ITR1 再配置

ソフトウェアによってセット/クリアされます。

0 : TIM9 ITR1 入力と TIM3_TGO 信号を接続

1 : TIM9 ITR1 入力とタッチ検出 I/O を接続 [図 31 : タイマモードアクイジションのロジック \(ページ\) 235](#) を参照

このビットは、Cat.3、Cat.4、Cat.5デバイスでのみ使用可能です。

ビット 1:0 TIM9_OR[1:0] タイマ 9 入力 1 再配置

ソフトウェアによってセット/クリアされます。

00 : TIM9 チャンネル 1 と GPIO を接続 : 共用機能との対応を参照してください。

01 : 測定のため LSE 内部クロックと TIM9_CH1 入力を接続

10 : TIM9 チャンネル 1 と GPIO を接続

11 : TIM9 チャンネル 1 と GPIO を接続



18.4.15 TIM9 レジスタマップ

TIM9レジスタは、次のように 16 ビット・アドレス可能レジスタとしてマップされます：

表 84. TIM9 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	TIMx_CR1	予約済み																						CKD[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN		
	0																							0	0	0	0	0	0	0					
0x04	TIMx_CR2	予約済み																						MMS[2:0]			予約済み								
	0																							0	0										
0x08	TIMx_SMCR	予約済み																ETP	ECE	ETPS[1:0]		ETF[3:0]		MSM	TS[2:0]			予約済み	SMS[2:0]						
	0																	0	0	0	0	0	0	0	0	0	0		0	0	0				
0x0C	TIMx_DIER	予約済み																						TIE	予約済み			CC2IE	CC1IE	UIE					
	0																							0				0	0						
0x10	TIMx_SR	予約済み																		CC2OF	CC1OF	予約済み	TIF	予約済み			CC2IF	CC1IF	UIF						
	0																			0	0		0				0	0							
0x14	TIMx_EGR	予約済み																						TG	予約済み			CC2G	CC1G	UG					
	0																							0				0	0	0					
0x18	TIMx_CCMR1 出力比較モード	予約済み																OC2M[2:0]		OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]		OC1PE	OC1FE	CC1S[1:0]						
	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	予約済み																IC2F[3:0]		IC2PSC[1:0]	CC2S[1:0]	IC1F[3:0]			IC1PSC[1:0]	CC1S[1:0]									
	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
0x1C		予約済み																																	
0x20	TIMx_CCER	予約済み																						CC2NP	予約済み	CC2P	CC2E	CC1NP	予約済み	CC1P	CC1E				
	0																							0		0	0	0		0	0				
0x24	TIMx_CNT	予約済み																CNT[15:0]																	
	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x28	TIMx_PSC	予約済み																PSC[15:0]																	
	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x2C	TIMx_ARR	予約済み																ARR[15:0]																	
	1																	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0x30		予約済み																																	
0x34	TIMx_CCR1	予約済み																CCR1[15:0]																	
	0																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

表 84. TIM9 の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
0x38	TIMx_CCR2	予約済み																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	予約済み																																
0x50	TIM9_OR	予約済み																										ITR1_RMP		TI1_RMP			
	リセット値																											0	0	0			

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

18.5 TIM10/11 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。

18.5.1 TIM10/11 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
予約済み						CKD[1:0]		ARPE	予約済み				URS	UDIS	CEN
						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 (ETR、Tix) によって使用されるサンプリングクロックとの間のプリスケアラ比を示します。

- 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 6:3 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 URS : 更新リクエストソース

このビットは、更新割り込み (UEV) 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 UEV 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 UEV 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、更新割り込み (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。UEV イベントは、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。
 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。UEV は生成されず、シャドウレジスタ (ARR、PSC、CCRx) 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。

18.5.2 TIM10/11 スレーブモード制御レジスタ 1 (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]				予約済み							
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 ETR のいずれをトリガ動作に使用するかを選択します。
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有効になります。
1 : ETR は反転され、ローレベルまたは立ち下がりエッジで有効になります。

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、外部クロックモード 1 で TRGI が ETRF に接続された状態 (SMS=111、TS=111) と同じ効果があります。
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 である必要があります。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} で実行
0001 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=2
0010 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=4
0011 : $f_{SAMPLING}=f_{CK\_INT}$ 、N=8²
0100 : $f_{SAMPLING}=f_{DTS}/2$ 、N=6
0101 : $f_{SAMPLING}=f_{DTS}/2$ 、N=8
0110 : $f_{SAMPLING}=f_{DTS}/4$ 、N=6²
0111 : $f_{SAMPLING}=f_{DTS}/4$ 、N=8
1000 : $f_{SAMPLING}=f_{DTS}/8$ 、N=6
1001 : $f_{SAMPLING}=f_{DTS}/8$ 、N=8
1010 : $f_{SAMPLING}=f_{DTS}/16$ 、N=5
1011 : $f_{SAMPLING}=f_{DTS}/16$ 、N=6
1100 : $f_{SAMPLING}=f_{DTS}/16$ 、N=8
1101 : $f_{SAMPLING}=f_{DTS}/32$ 、N=5
1110 : $f_{SAMPLING}=f_{DTS}/32$ 、N=6
1111 : $f_{SAMPLING}=f_{DTS}/32$ 、N=8

ビット 7:0 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 (株) 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) 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。

18.5.3 TIM10/11 割り込みイネーブルレジスタ (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
予約済み														CC1IE	UIE
														rw	rw

ビット 15:2 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 **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。

18.5.4 TIM10/11 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
予約済み						CC1OF	予約済み							CC1IF	UIF
						rc_w0								rc_w0	rc_w0

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 **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、対応するチャン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 8:2 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 **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。このビットは、ソフトウェアでクリアされます。

0 : 一致なし。

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 (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。このビットは、ソフトウェアでクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みがペンディング中です。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。

18.5.5 TIM10/11 イベント発生レジスタ (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
予約済み														CC1G	UG
														w	w

ビット 15:2 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 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 :

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)。カウンタはクリアされます。

18.5.6 TIM10/11 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示します。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
予約済み								OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
								IC1F[3:0]				IC1PSC[1:0]			
								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 15:8 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 **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 が派生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 のアクティブレベルは CC1P ビットに依存します。
- 000 : 凍結。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。
- 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 をイン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 - 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。
- 111 : PWM モード 2 - 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。

注 : PWM モード 1 または 2 では、比較結果が変化するとき、または出力比較モードが凍結モードから PWM モードに変更されたときに、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ず。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有効にしなくても使用できます。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、トリガ入力のイベントから CC 出力への影響の伝播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。その後、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 レジスタで 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 15:8 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 7:4 **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ために使用される周波数と、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 : 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 : $f_{SAMPLING}=f_{DTS}/2$, $N=6$	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プリスケアラ比を定義します。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 :
- 11 :

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。

18.5.7 TIM10/11 キャプチャ／比較イネーブルレジスタ (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
予約済み												CC1NP	予約済み	CC1P	CC1E
												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 **CC1NP** : キャプチャ／比較 1 相補出力極性

CC1 チャンネルを出力として設定 : CC1NP はクリアされたままにする必要があります。

CC1 チャンネルを入力として設定 : CC1NP ビットは、TI1FP1 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 アクティブハイ

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 :

CC1P ビットは、トリガ動作またはキャプチャ動作での 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 TI1FP1 の立ち上がりエッジにตอบสนองし (キャプチャモード)、TI1FP1 は反転されません。

01 : 反転/立ち下がりエッジ

回路は TI1FP1 の立ち下がりエッジにตอบสนองし (キャプチャモード)、TI1FP1 は反転されます。

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 TI1FP1 の立ち上がりエッジにตอบสนองし (キャプチャモード)、TI1FP1 は反転されません。

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 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。

18.5.8 TIM10/11 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

18.5.9 TIM10/11 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、 $f_{CK\_PSC} / (PSC[15:0] + 1)$ に一致します。

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。

18.5.10 TIM10/11 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 : 0xFFFF

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 18.3.1 : タイムベースユニット \(459 ページ\)](#) 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。

18.5.11 TIM10/11 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、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。

CC1 チャンネル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。

18.5.12 TIM10 オプションレジスタ 1 (TIM10_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
予約済み												T11_RMP_RI	ETR_RMP	T11_RMP[1:0]	
												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 **T11_RMP_RI** : ルーティングインタフェース (RI) のタイマ 10 入力 1 再配置
ソフトウェアによってセット/クリアされます。
0 : T11_RMP[1:0] ビット値に依存する TIM10 チャンネル 1 接続 (下記参照)
1 : TIM10 チャンネル 1 と RI を接続 (T11_RMP ビットの値によらず)。
このビットは、Cat.3、Cat.4、Cat.5デバイスでのみ使用可能です。
- ビット 2 **ETR_RMP** : タイマ 10 ETR 再配置
ソフトウェアによってセット/クリアされます。
0 : TIM10 ETR 入力と LSE クロックを接続
1 : TIM10 ETR 入力と TIM9_TGO を接続
このビットは、Cat.3、Cat.4、Cat.5デバイスでのみ使用可能です。
- ビット 1:0 **T11_RMP[1:0]** : TIM10 入力再配置機能
ソフトウェアによってセット/クリアされます。
00 : TIM10 チャンネル 1 と GPIO を接続 : 共用機能との対応を参照してください。
01 : 測定のため LSI 内部クロックと TIM10_CH1 入力を接続
10 : 測定のため LSE 内部クロックと TIM10_CH1 入力を接続
11 : 測定のため RTC ウェイクアップ割り込み信号と TIM10_CH1 入力を接続



18.5.13 TIM11 オプションレジスタ 1 (TIM11_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
予約済み												TI1_RM P_RI	ETR_RM P	TI1_RMP[1:0]	
												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 **TI1_RMP_RI** : ルーティングインタフェース (RI) のタイマ 11 入力 1 再配置
ソフトウェアによってセット/クリアされます。
0 : TI1_RMP[1:0] ビット値に依存する TIM11 チャンネル 1 接続 (下記参照)
1 : TIM11 チャンネル 1 と RI を接続 (TI1_RMP ビットの値によらず)。
このビットは、Cat.3、Cat.4、Cat.5デバイスでのみ使用可能です。
- ビット 2 **ETR_RMP** : タイマ 11 ETR 再配置
ソフトウェアによってセット/クリアされます。
0 : TIM11 ETR 入力と LSE クロックを接続
1 : TIM11 ETR 入力と TIM9_TGO を接続
このビットは、Cat.3、Cat.4、Cat.5デバイスでのみ使用可能です。
- ビット 1:0 **TI1_RMP[1:0]** : TIM11 入力 1 再配置機能
ソフトウェアによってセット/クリアされます。
00 : TIM11 チャンネル 1 を GPIO に接続 : データシートの代替機能配置表を参照してください。
01 : MSI の内部クロックは測定のために TIM11_CH1 入力へ接続
10 : HSE_RTC クロック (プログラマブルプリスケラによって分周された HSE) は測定のために TIM11_CH1 入力へ接続
11 : TIM11 チャンネル 1 と GPIO を接続

TIMx レジスタは、次の表のように、16 ビットのアドレス可能レジスタとしてマップされます。

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	予約済み																						CKD[1:0]		ARPE	予約済み			URS	UDIS	CEN	
	リセット値																							0	0								0
0x08	TIMx_SMCR	予約済み														ETP	ECE	ETPS[1:0]	ETF [3:0]			予約済み											
	リセット値																													0	0	0	0
0x0C	TIMx_DIER	予約済み																										CC1IE	UIE				
	リセット値																													0	0		
0x10	TIMx_SR	予約済み																		CC1OF	予約済み						CC1IF	UIF					
	リセット値																												0	0			
0x14	TIMx_EGR	予約済み																										CC1G	UG				
	リセット値																													0	0		
0x18	TIMx_CCMR1 出力比較モード	予約済み																		OC1CE	OC1M [2:0]				OC1PE	OC1FE	CC1S [1:0]						
	リセット値																											0	0	0	0	0	0
	TIMx_CCMR1 入力キャプチャモード	予約済み																		IC1F[3:0]				IC1PSC [1:0]	CC1S [1:0]								
	リセット値																									0	0	0	0	0	0	0	0
0x1C	予約済み																																
0x20	TIMx_CCER	予約済み																										CC1NP	予約済み		CC1P	CC1E	
	リセット値																																0
0x24	TIMx_CNT	予約済み														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	予約済み														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	予約済み														ARR[15:0]																	
	リセット値															1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		
0x30	予約済み																																
0x34	TIMx_CCR1	予約済み														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 から 0x4C	予約済み																																

表 86. TIM10/11の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	TIMx_OR	予約済み																												T11_RMP_RI	ETR_RMP	T11_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、表 5 (47 ページ) を参照してください。

19 基本タイマ (TIM6、TIM7)

19.1 TIM6 および TIM7 の概要

基本タイマ TIM6 および TIM7 は、プログラマブルなプリスケア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、これらのタイマは内部で DAC に接続されていて、トリガ出力を通じて駆動することができます。

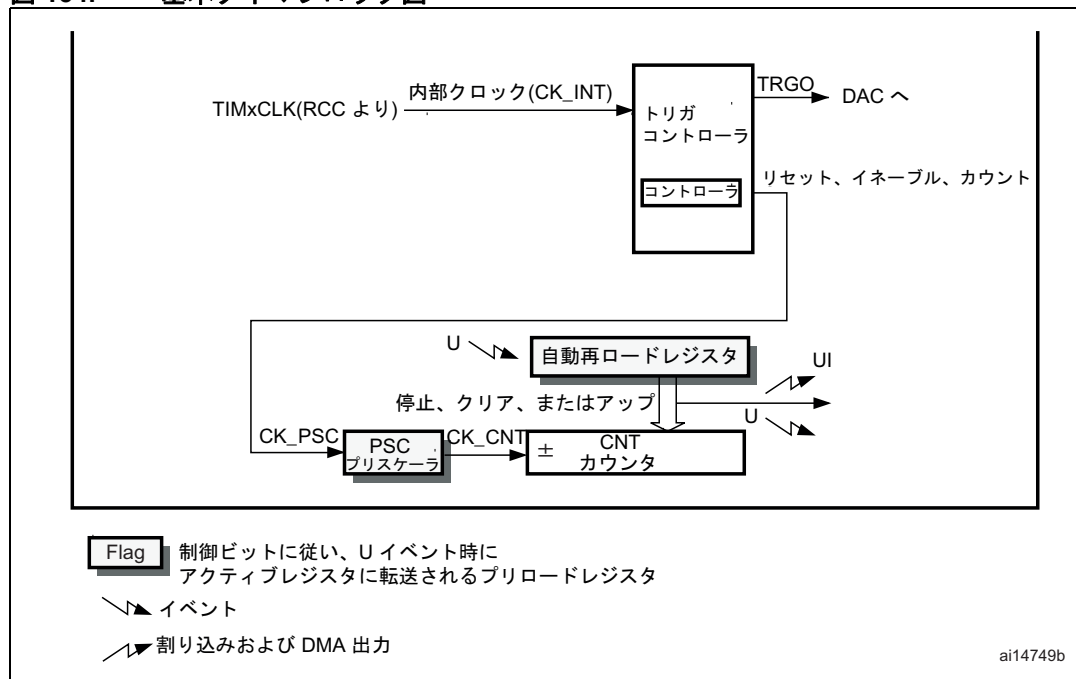
タイマは完全に独立していて、いかなるリソースも共有しません。

19.2 TIM6 および TIM7 の主な機能

基本タイマ (TIM6 および TIM7) 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
- カウンタクロック周波数を、1 から 65536 の間で分周するための 16 ビットプログラマブルプリスケアラ
- 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 154. 基本タイマブロック図



19.3 TIM6 および TIM7 の機能詳細

19.3.1 タイムベースユニット

プログラマブル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。カウンタクロックは、プリスケアラで分周することができます。

カウンタ、自動再ロードレジスタ、プリスケアラレジスタは、ソフトウェアから読み書き可能です。カウンタの動作中も同様です。

タイムベースユニットには、次が含まれます。

- 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が試みられるたびに、プリロードレジスタがアクセスされます。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移されます。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、ソフトウェアから生成することもできます。更新イベントの生成は、それぞれの構成について詳しく説明します。

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にのみ、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。

プリスケアラの詳細

プリスケアラは、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。プリスケアラは、16 ビットレジスタ (TIMx_PSC) を使って制御される 16 ビットカウンタに基づきます。TIMx_PSC 制御レジスタはバッファされるので、即座に変更できます。新しいプリスケアラ比は、次の更新イベントで有効になります。

図 155 と 図 156 は、プリスケアラ比がその場で変更されたときのカウンタの動作の例を示します。

図 155. プリスケアラ比が 1 から 2 に変化したときのカウンタのタイミング図

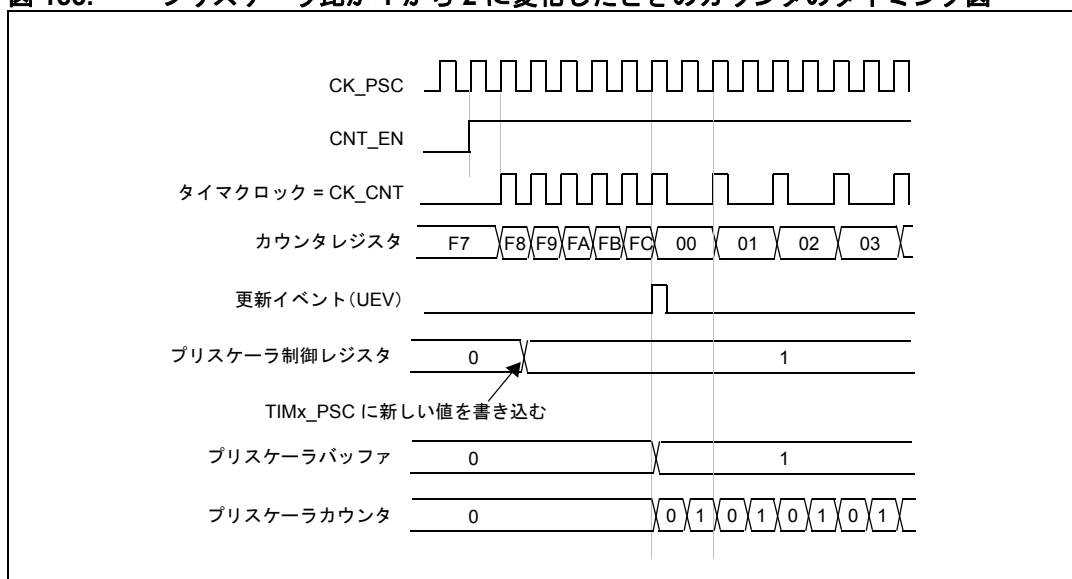
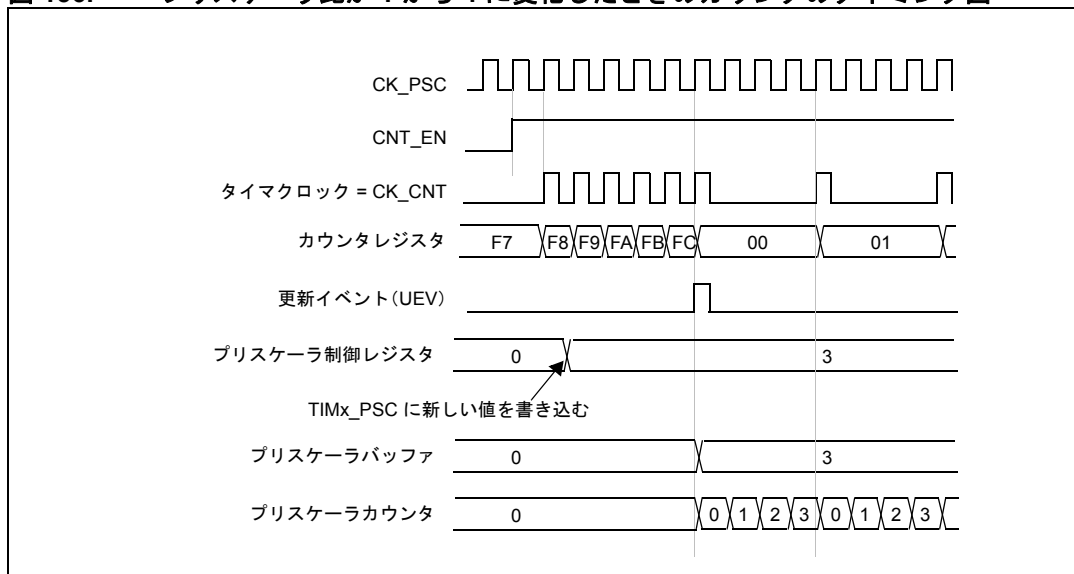


図 156. プリスケアラ比が 1 から 4 に変化したときのカウンタのタイミング図



19.3.2 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、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 (ソフトウェアによって、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 (Update Request Selection : 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 157. 内部クロック分周比が 1 の場合のカウンタタイミング図

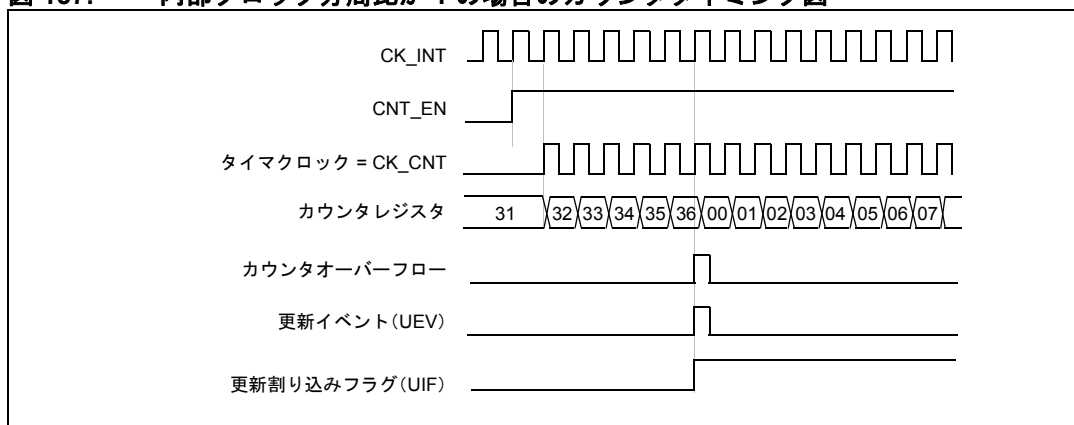


図 158. カウンタタイミング図、内部クロック 2 分周

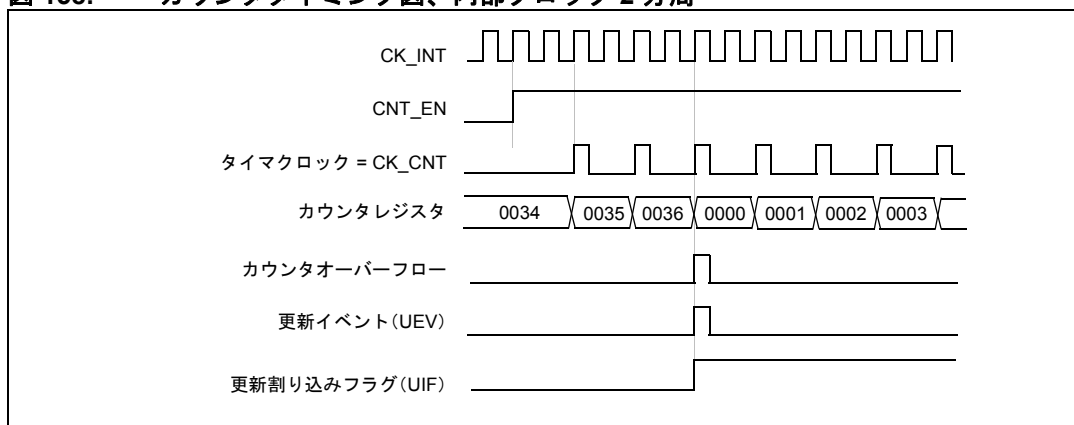


図 159. カウンタタイミング図、内部クロック 4 分周

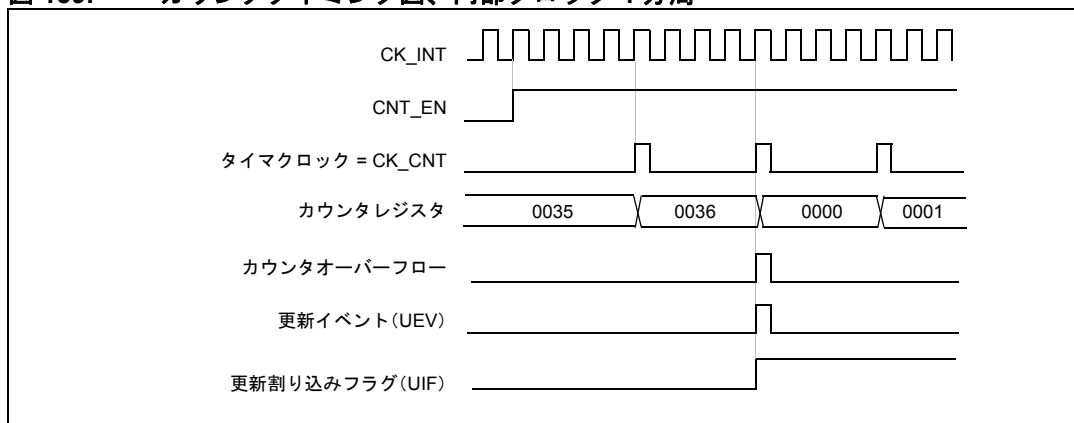


図 160. 内部クロックプリスケアラ比が N の場合のカウンタのタイミング図

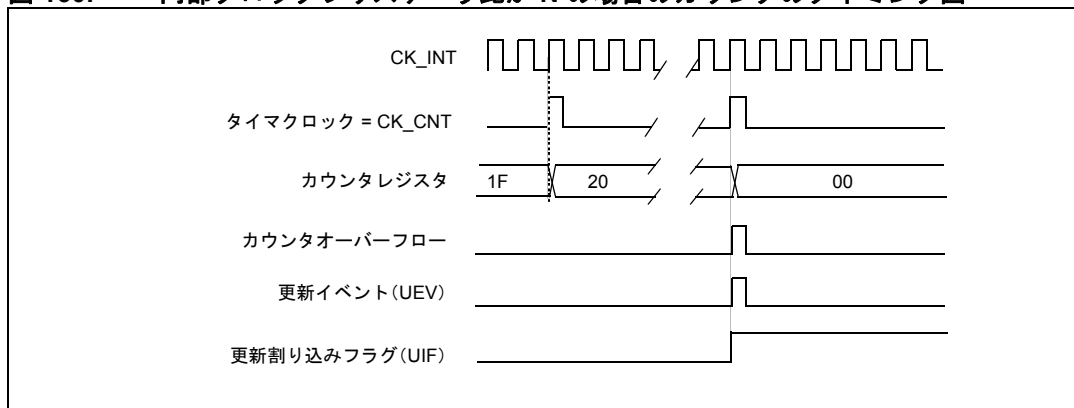


図 161. ARPE=0 (TIMx_ARR はプリロードされない) のときの更新イベント時のカウンタのタイミング図

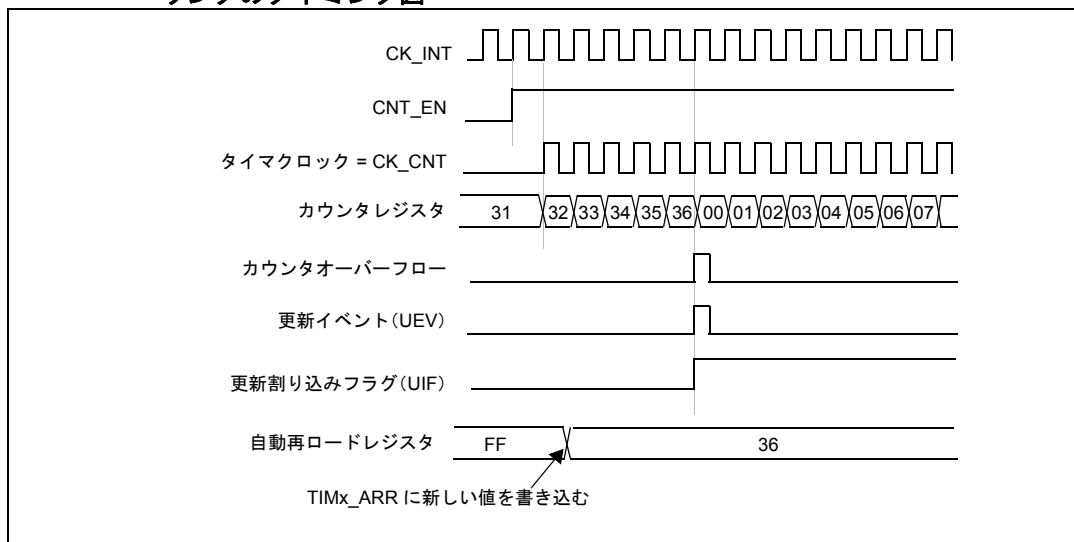
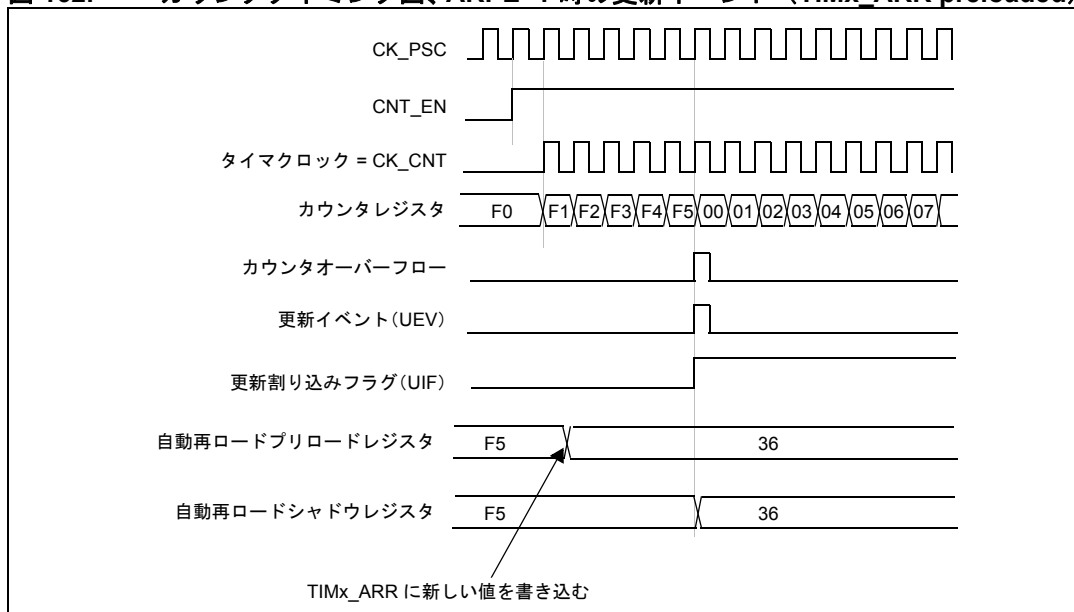


図 162. カウンタタイミング図、ARPE=1 時の更新イベント (TIMx_ARR preloaded)



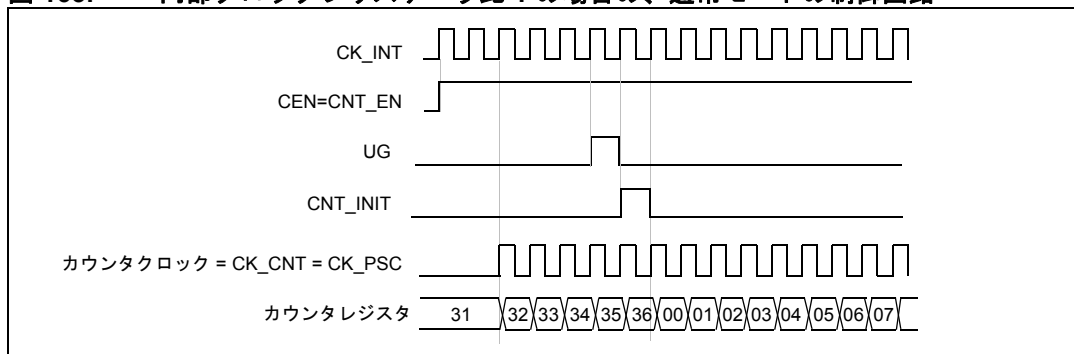
19.3.3 クロックソース

カウンタクロックは、内部クロック (CK_INT) ソースから供給されます。

TIMx_CR1 レジスタの CEN ビットと TIMx_EGR レジスタの UG ビットは実際の制御ビットであり、ソフトウェアによってのみ変更できます (ただし、UG は、自動的にクリアされたままになります)。CEN ビットに 1 が書き込まれると、プリスケアラに内部クロック CK_INT が供給されます。

図 163 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 163. 内部クロックプリスケアラ比 1 の場合の、通常モードの制御回路



19.3.4 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®-M3 コア停止)、DBG モジュールの DBG_TIMx_STOP 設定ビットに応じて、TIMx カウンタは通常通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 30.16.2: 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。

19.4 TIM6 および TIM7 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。

19.4.1 TIM6 および TIM7 制御レジスタ 1 (TIMx_CR1)

アドレスオフセット : 0x00

リセット値 : 0xFFFF

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
予約済み								ARPE	予約済み				OPM	URS	UDIS	CEN
								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 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 6:4 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 ビットをクリアします)。

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。更新イベントは発生されず、シャドウレジスタ (ARR、PSC) はそれぞれの値のままです。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によってセットされていた場合のみ動作します。ただし、トリガモードでは、CEN ビットはハードウェアによって自動的にセットできます。

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。

19.4.2 TIM6 および TIM7 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
予約済み									MMS[2:0]			予約済み			
									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[2:0]** : 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されます。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生成された場合 (スレーブモードコントローラがリセットモードに構成されているとき)、TRGO 信号は実際のリセットより遅れます。

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、複数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ウィンドウ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。

19.4.3 TIM6 および TIM7 DMA/割り込みイネーブルレジスタ (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
予約済み							UDE	予約済み							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。

19.4.4 TIM6 および TIM7 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
予約済み															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。このビットは、ソフトウェアでクリアされます。

0 : 更新は発生していません。

1 : 更新割り込みがベンディング中です。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 の場合、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、TIMx_EGR レジスタの UG ビットを使用して、ソフトウェアによって CNT が再初期化されたとき。

19.4.5 TIM6 および TIM7 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
予約済み															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 : 影響なし。

1 : カウンタを再初期化し、レジスタの更新が発生します。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）。

19.4.6 TIM6 および TIM7 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

19.4.7 TIM6 および TIM7 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、 $f_{CK\_PSC} / (PSC[15:0] + 1)$ に一致します。
 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。

19.4.8 TIM6 および TIM7 自動リロードレジスタ (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 19.3.1 : タイムベースユニット \(512 ページ\)](#) 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。

19.4.9 TIM6 および TIM7 レジスタマップ

TIMx レジスタは、次の表のように、16 ビットのアドレス可能レジスタとしてマップされます。

表 87. TIM6 および TIM7 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	TIMx_CR1	予約済み																								ARPE	予約済み			OPM	URS	UDIS	CEN
	リセット値																									0				0	0	0	
0x04	TIMx_CR2	予約済み																								MMS[2:0]			予約済み				
	リセット値																									0	0	0					
0x08	予約済み																																
0x0C	TIMx_DIER	予約済み																								UDE	予約済み			UIE			
	リセット値																									0				0			
0x10	TIMx_SR	予約済み																															UIF
	リセット値																																0
0x14	TIMx_EGR	予約済み																															UG
	リセット値																																0
0x18	予約済み																																
0x1C	予約済み																																
0x20	予約済み																																
0x24	TIMx_CNT	予約済み																CNT[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x28	TIMx_PSC	予約済み																PSC[15:0]															
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
0x2C	TIMx_ARR	予約済み																ARR[15:0]															
	リセット値																	1	1	1	1	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、表 5 (47 ページ) を参照してください。



20 リアルタイムクロック (RTC)

20.1 概要

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、定期的な割り込み機能を備えたプログラマブルウェイクアップフラグを搭載しています。RTC には、低電力モードを管理する自動ウェイクアップユニットも搭載されています。

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、時間 (12 時間または 24 時間形式)、曜日、日、月、年が含まれています。バイナリ形式では、サブセカンド値も利用できます (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)。

28、29 (うるう年)、30、31 日ある月の補正は、自動的に実施されます。サマータイムも補正できます。

もう 1 つの 32 ビットレジスタには、プログラム可能なアラームサブセカンド (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)、秒、分、時、曜日、日付が含まれています。

また、デジタル較正機能を利用すると、クリスタルオシレータ精度の偏差を補正できます。

電源オン時のリセット後、すべての RTC レジスタは不正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。

20.2 RTC の主な機能

RTC コントローラは次のような機能を持っています (図 164 : RTC のブロック図 (Cat.1 製品) を参照)。

- サブセカンド (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)、秒、分、時 (12 または 24 時間形式)、曜日、日、月、年のあるカレンダー。
- サマータイムの補正はソフトウェアによってプログラムされます。
- 割り込み機能を備えた 2 本のプログラム可能なアラーム。アラームは、どのカレンダー項目の組み合わせでもトリガできます。
- 自動ウェイクアップ割り込みをトリガする定期的なフラグを生成する自動ウェイクアップユニット。
- 基準クロック検出：より正確な秒ソースのクロック (50 または 60 Hz) を使用すると、カレンダーの精度を向上させることができます。
- サブセカンドのシフト機能を使用する外部クロックとの正確な同期 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)。
- マスク可能割り込み／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
- デジタル較正回路 (定期的なカウンタ補正)
 - － 5 ppm の精度 (Cat.1 製品)
 - － 0.95 ppm の精度 (Cat.2、Cat.3、Cat.4 および Cat.5 製品)、数秒の較正時間範囲で取得

- イベントを保存するタイムスタンプ機能 (イベント 1 つ)
- タンパ検出 :
 - エッジ検出時のタンパイベント 1 つ (Cat.1 製品)
 - 設定可能なフィルタおよび内部プルアップのある 3 つのタンパイベント (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)。
 - バリユーラインに 5 個のバックアップレジスタ (20 バイト)、Cat.1 および Cat.2 製品 (バリユーラインを除く) に 20 個のバックアップレジスタ (80 バイト) と、Cat.3 (バリユーラインを除く)、Cat.4 および Cat.5 製品に 32 個のレジスタ (128 バイト) があります。
- 代替機能出力 : 以下の 2 つの出力のうち 1 つを選択する RTC_OUT :
 - RTC_CALIB : 512 Hz または 1 Hz (1 Hz は Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ) のクロック出力 (LSE 周波数 32.768 kHz)。この出力を有効にするには、RTC_CR レジスタの COE ビットをセットします。デバイス RTC_AF1 機能に配線されています。
 - RTC_ALARM : アラーム A。この出力を選択するには、RTC_CR レジスタの OSEL[1:0] ビットを設定します。デバイス RTC_AF1 機能に配線されています。
- RTC 代替機能入力 :
 - RTC_TS : タイムスタンプイベント検出です。デバイス RTC_AF1 機能に配線されています。
 - RTC_TAMP1 : TAMPER1 イベント検出です。デバイス RTC_AF1 機能に配線されています。
 - RTC_TAMP2 : TAMPER2 イベント検出です。
 - RTC_TAMP3 : TAMPER3 イベント検出です。
 - RTC_REFIN : 基準クロック入力 (通常はメイン、50 または 60 Hz) です。

注 : **RTC 代替機能 (RTC_AF1) 選択方法の詳細については、[セクション 7.4.1 : GPIO ポートモードレジスタ \(GPIOx_MODER\) \(x = A..H\)](#) を参照してください。**

図 164. RTC のブロック図 (Cat.1 製品)

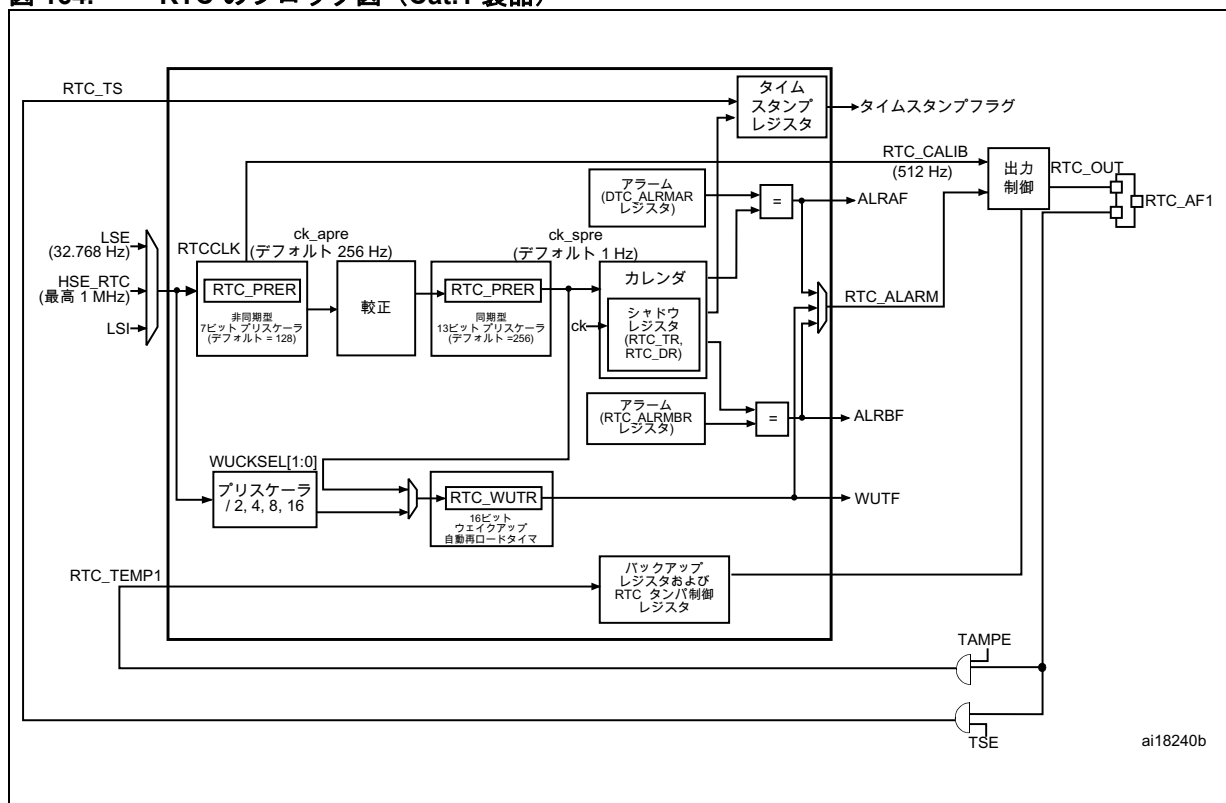
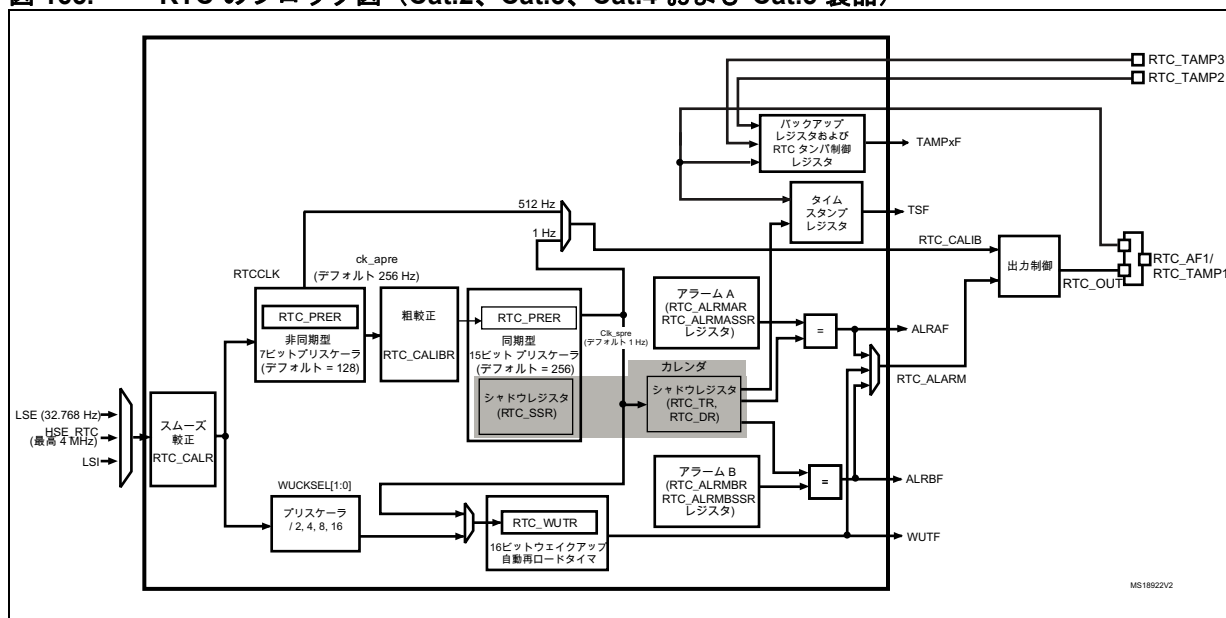


図 165. RTC のブロック図 (Cat.2、Cat.3、Cat.4 および Cat.5 製品)



20.3 RTC の機能詳細

20.3.1 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 クロックにあるクロックコントローラを通じて選択されます。RTC クロックソース設定の詳細については、[セクション 6 : 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が、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。消費電力を最小に抑えるため、プリスケアラはプログラム可能な 2 つのプリスケアラに分割されます ([図 164 : RTC のブロック図 \(Cat.1 製品\)](#) 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
- プリスケアラは、Cat.1 製品に 13 ビット、Cat.2、Cat.3、Cat.4 および Cat.5 製品に 15 ビットがあります。

注 : *両方のプリスケアラを使用する際は、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が推奨されます。*

LSE 周波数 32.768 kHz で 1 Hz (ck_spre) の内部クロック周波数を取得するよう、非同期プリスケアラ分周比は 128、同期分周比は 256 に設定されます。

最小分周比は Cat.1 製品で 2、Cat.2、Cat.3、Cat.4 および Cat.5 製品で 1 です。

最大分周比は Cat.1 製品で 2^{20} 、Cat.2、Cat.3、Cat.4 および Cat.5 製品で 2^{22} です。

これは Cat.1 製品で最大入力周波数約 1 MHz、Cat.2、Cat.3、Cat.4 および Cat.5 製品で 4 MHz に相当します。

f_{ck_apre} は、次の公式で与えられます。

$$f_{CK\_APRE} = \frac{f_{RTCCLK}}{PREDIV\_A + 1}$$

ck_apre クロックをバイナリ RTC_SSR のサブセカンドのダウンカウンタのクロックとして使用します。値がゼロになると、RTC_SSR が PREDIV_S の内容で再ロードされます。RTC_SSR は、Cat.2、Cat.3、Cat.4 および Cat.5 製品のみで利用できます。

f_{ck_spre} は、次の公式で与えられます。

$$f_{CK\_SPRE} = \frac{f_{RTCCLK}}{(PREDIV\_S + 1) \times (PREDIV\_A + 1)}$$

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取得するには、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 ビットの非同期プリスケアラで分周した RTCCLK で実行することもできます (詳細は [セクション 20.3.4 : 定期的な自動ウェイクアップ](#) を参照)。

20.3.2 リアルタイムクロックとカレンダー

RTC カレンダーの時間および日付レジスタには、PCLK1 (APB1 クロック) と同期するシャドウレジスタを通じてアクセスします。Cat.2、Cat.3、Cat.4 および Cat.5 製品では、同期持続の待ち時間を避けるため、直接アクセスすることもできます。

- サブセカンド用 RTC_SSR (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)
- 時間用 RTC_TR
- 日付用 RTC_DR

RTCCLK 2 周期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 20.6.4 参照)。STOP および STANDBY モードでは、コピーはされません。これらのモードを終了すると、RTCCLK 2 周期以内にシャドウレジスタが更新されます。

アプリケーションは、カレンダーレジスタを読み出す際にシャドウレジスタの内容にアクセスします。Cat.2、Cat.3、Cat.4 および Cat.5 製品では、RTC_CR レジスタの BYPSHAD 制御ビットをセットすると、カレンダーレジスタに直接アクセスできます。デフォルトによりこのビットはクリアされ、ユーザー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 モードで読み出す際は、APB クロックの周波数 (f_{APB}) が RTC クロック (f_{RTCCLK}) の周波数の 7 倍以上である必要があります。

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。

20.3.3 プログラム可能なアラーム

RTC ユニットは、2 本の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 を提供します。

プログラム可能なアラーム機能は、RTC_CR レジスタの ALRAIE および ALRBIE ビットで有効になります。ALRAF および ALRBF フラグは、カレンダーのサブセカンド (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)、秒、分、時、日または曜日がそれぞれアラームレジスタ RTC_ALRMASR / RTC_ALRMAR および RTC_ALRMBSSR / RTC_ALRMBR でプログラムされている値と一致すると 1 にセットされます。各カレンダーフィールドは、RTC_ALRMAR および RTC_ALRMBR レジスタの MSKx ビット、および RTC_ALRMASR および RTC_ALRMBSSR レジスタの MASKSSx ビットで個別に選択できます。アラーム割り込みは、RTC_CR レジスタの ALRAIE および ALRBIE ビットで有効になります。

アラーム A およびアラーム B (RTC_CR レジスタの OSEL[1:0] 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 の極性は、RTC_CR レジスタの POL ビットで設定できます。

注意： 秒フィールドが選択されている場合 (RTC_ALRMAR または RTC_ALRMBR で MSK0 ビットをリセット)、正しい動作が保証されるように 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が 3 以上である必要があります。

20.3.4 定期的な自動ウェイクアップ

定期的な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で生成されます。ウェイクアップタイマの範囲は 17ビットまで拡大できます。

ウェイクアップ機能は、RTC_CR レジスタの WUTE ビットで有効になります。

ウェイクアップタイマクロック入力には、次のようなものが考えられます。

- 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、これによって分解能を 61 μ s に下げてウェイクアップ割り込み周期を 122 μ s から 32 s に設定できるようになります。
- ck_spre (通常は 1 Hz の内部クロック)
ck_spre 周波数が 1Hz の場合、これによって 1 秒の分解能で 1 s からおよそ 36 時間のウェイクアップ時間が達成できるようになります。この大きなプログラム可能な時間範囲は、2 つの部分に分かれています。
 - WUCKSEL [2:1] = 10 の場合 1 から 18 時間、
 - そして 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 が追加されます。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ム \(530 ページ\)](#) 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になっている場合、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 レジスタで 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定期的な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。

定期的なウェイクアップフラグは、RTC_CR レジスタの OSEL[1:0] ビットで有効になっていれば RTC_ALARM 出力に配線できます。RTC_ALARM の極性は、RTC_CR レジスタの POL ビットで設定できます。

低電力モード (SLEEP、STOP、STANDBY) だけでなく、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。

20.3.5 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 つのウェイトステートを導入します。ただし、BYPHAD=0 の際のカレンダ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 電源制御レジスタ (PWR_CR) の DBP ビットで不正な書き込みアクセスから保護されます。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。

電源オン時のリセット後、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 にキーを書き込むと有効になります。

RTC_ISR[13:8]、RTC_TAFCR、RTC_BKPxR 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となります。

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など、初期時間と日付カレンダー値をプログラムするには、次のシーケンスが必要となります。

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、値を更新できます。
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 1 から 2 周期かかります (クロック同期のため)。
3. カレンダーカウンタに 1 Hz クロックを生成するには、まず RTC_PRER レジスタで同期プリスケアラ比をプログラムし、その後非同期プリスケアラ比をプログラムします。2 つのフィールドのうち 1 つだけを変更しなければならない場合でも、RTC_PRER レジスタには 2 つの別々の書き込みアクセスを実行する必要があります。
4. シャドウレジスタ (RTC_TR および RTC_DR) で初期時間と日付値をロードし、RTC_CR レジスタの FMT ビットで時間形式 (12 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。実際のカレンダーカウンタ値は、その後自動的にロードされ、RTCCLK クロック 4 周期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どうかを確認できるようになります。このフラグが 0 である場合、カレンダーは年フィールドが電源オン時のリセットデフォルト値 (0x00) にセットされて以来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、ソフトウェアがまず RTC_ISR レジスタで 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

サマータイムは、RTC_CR レジスタの SUB1H、ADD1H、BKP ビットで管理されます。

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なくても 1 度の操作でカレンダーから 1 時間引いたり、カレンダーに 1 時間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できます。

アラームのプログラム

プログラム可能なアラーム（アラーム A またはアラーム B）をプログラムまたは更新するには、同じような手順を実施する必要があります。

1. RTC_CR の ALRAE または ALRBIE をクリアしてアラーム A またはアラーム B を無効にします。
2. アラームレジスタへのアクセスが許可されていることを確認するようにセットされるまで、RTC_ISR で ALRAWF または ALRBWF をポーリングします。Cat.1 製品では、これに RTCCLK クロック 1 から 2 周期かかります（クロック同期のため）。Cat.2、Cat.3、Cat.4 および Cat.5 製品では、ALRAWF および ALRBWF が常にセットされますので、このステップは飛ばしてもかまいません。
3. アラーム A またはアラーム B レジスタ（RTC_ALRMASR/RTC_ALRMAR または RTC_ALRMBSSR/RTC_ALRMBR）をプログラムします。
4. RTC_CR レジスタで ALRAE または ALRBIE をセットしてアラーム A またはアラーム B 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 1 から 2 周期後に考慮されません。

ウェイクアップタイマのプログラム

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、次のシーケンスが必要です。

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 でセットされるまで WUTWF をポーリングし、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ることを確認します。これには RTCCLK クロック 1 から 2 周期かかります（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 をセットして再びタイマを有効にします。ウェイクアップタイマがカウントダウンを再開します。

20.3.6 カレンダの読み出し

Cat.1 製品の場合、または BYPSHAD 制御ビットが RTC_CR レジスタでクリアされるとき

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_{PCLK1}) が f_{RTCCLK} RTC クロック周波数の 7 倍以上である必要があります。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レジスタが 2 回読み込まれる必要があります。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実行する必要があります。どの場合も、APB1 クロック周波数は必ず RTC クロック周波数よりも大きい必要があります。

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、RTCCLK 2 周期ごとに実施されます。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、RTC_DR が読み出されるまで高位にあるカレンダーシャドウレジスタの値がロックされます。ソフトウェアが RTCCLK 2 周期未満の間隔でカレンダーに読み出しアクセスをしようとすると、最初のカレンダー読み出し後に RSF がソフトウェアによってクリアされ、その後ソフトウェアが RSF ビットがセットされるまで待たなければ、RTC_SSR、RTC_TR および RTC_DR レジスタを再び読み出すことはできません。

低電力モード (STOP または STANDBY) からのウェイクアップ後は、ソフトウェアによって RSF がクリアされる必要があります。その後ソフトウェアは、もう一度セットされるまで待たなければ、RTC_SSR、RTC_TR および RTC_DR レジスタを再び読み出すことはできません。

RSF ビットは、低電力モードに入る前ではなく、ウェイクアップ後にクリアされる必要があります。

注： 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たなければ、RTC_SSR、RTC_TR および RTC_DR レジスタを読み出すことはできません。実際、システムリセットはシャドウレジスタをデフォルト値にリセットします。

初期化 (カレンダーの初期化と設定 (529 ページ) 参照) 後、 ソフトウェアは RSF がセットされるまで待たなければ、RTC_SSR、RTC_TR および RTC_DR レジスタを読み出すことはできません。

同期 (セクション 20.3.8 : RTC 同期 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ) 参照) 後、 ソフトウェアは RSF がセットされるまで待たなければ、RTC_SSR、RTC_TR および RTC_DR レジスタを読み出すことはできません。

Cat.2、Cat.3、Cat.4 および Cat.5 製品の場合、BYPSHAD 制御ビットが RTC_CR レジスタ (バイパスシャドウレジスタ) でセットされるとき

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なくなります。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、これは低電力モード終了後は特に有用です。

レジスタへの 2 回の読み出しアクセス間で RTCCLK エッジが発生する場合は、BYPSHAD ビットが 1 にセットされると、さまざまなレジスタの結果がお互いに一致しないことがあります。さらに、読み出し操作中に RTCCLK エッジが発生すると、レジスタの 1 つの値が正しくないことがあります。ソフトウェアはすべてのレジスタを 2 回読み出し、結果を比較してデータが一貫して正しいことを確認する必要があります。または、ソフトウェアが最下位カレンダーレジスタの 2 回の結果を比較することもできます。

注： BYPSHAD=1 の間は、カレンダーレジスタを読み出す指示で APB をもう 1 周期余分に完了する必要があります。



20.3.7 RTC のリセット

カレンダーシャドウレジスタ (RTC_SSR、RTC_TR および RTC_DR) と RTC ステータスレジスタ (RTC_ISR) の一部のビットが、利用可能なすべてのシステムリセットソースによってデフォルト値にリセットされます。

逆に、次のレジスタは電源オン時のリセットによってデフォルト値にリセットされ、システムリセットの影響を受けません：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_CALIBR または 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 タンバおよび代替機能設定レジスタ (RTC_TAFCR)、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 およびアラーム B レジスタ (RTC_ALRMASR/RTC_ALRMAR および RTC_ALRMBSSR/RTC_ALRMBR)。

さらに、RTC はリセットソースが電源オン時リセットと異なっていれば、システムリセット中も実行を続けます。電源オン時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。

20.3.8 RTC 同期 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)

RTC は、高精度でリモートクロックと同期できます。サブセカンドフィールド (RTC_SSR または RTC_TSSSR) を読み出した後、リモートクロックと RTC によって維持される時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」して RTC を調整し、このオフセットを排除することができます。

RTC_SSR には、同期プリスケアラのカウンタの値が含まれています。これにより、RTC が維持した正確な時間を $1 / (\text{PREDIV}_S + 1)$ 秒までの正確さ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 ($\text{PREDIV}_S[14:0]$) を増加させて分解能を改善できます。 PREDIV_S を 0x7FFF にセットすると、許可されている最大分解能 (32768 Hz クロックで 30.52 μs) が得られます。

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 に維持するために PREDIV_A を減少させなければならないことを意味します。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 動的消費電力が増加することがあります。

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 に書き込むと、クロックを分解能 $1 / (\text{PREDIV}_S + 1)$ 秒で最大 1 秒シフトさせることができます (遅れるまたは進む)。シフト操作は、同期プリスケアラカウンタ $\text{SS}[15:0]$ に $\text{SUBFS}[14:0]$ 値を追加するという操作で構成されます。これでクロックが遅れます。同時に ADD1S ビットがセットされると、これは何分の 1 秒かを引くと同時に 1 秒追加することになるため、クロックが進みます。

注意： シフト操作を始める前に、ユーザーは $\text{SS}[15] = 0$ となっていることを確認し、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同時に、シフト操作が保留中であることを示すためにハードウェアによって SHPF フラグがセットされます。このビットは、シフト操作が完了すると、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、基準クロック検出機能とは互換性がありません。ファームウェアは、 $\text{REFCKON}=1$ のときは RTC_SHIFTR に書き込まないでください。

20.3.9 RTC 基準クロック検出

RTC カレンダ更新は、通常はメイン (50 または 60 Hz) である基準クロック RTC_REFIN に同期させることができます。RTC_REFIN 基準クロックには、32.768 kHz LSE クロックよりも高い精度が必要です。RTC_REFIN 検出が有効 (RTC_CR の REFCKON ビットが 1 にセット) の場合、カレンダーはまだ LSE によってクロック出力されており、RTC_REFIN はカレンダー更新周波数 (1 Hz) の不正確さを補償するのに使用します。

各 1 Hz クロックエッジは、一番近い基準クロックエッジ (所与の時間枠内にある場合) と比較されます。大半の場合、2 つのクロックエッジは正しく整列されています。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 のクロックを少しずらしてこの後の 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、カレンダーは基準クロックと同じくらい正確になるのです。

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、基準クロックソースがあるかどうかを検出します。検出は各カレンダー更新 (1 秒ごと) 程度の時間枠で実施されます。最初の基準クロックエッジを検出する際、時間枠は ck_apre 7 周期に等しくなります。その後のカレンダー更新では、もう少し小さい ck_apre 3 周期の時間枠が使用されます。

その時間枠で基準クロックが検出されるたびに、ck_apre クロックを出力する非同期プリスケアラが強制的に再ロードされます。これは、プリスケアラが同時に再ロードされているために基準クロックおよび 1 Hz のクロックが整列しているときには影響ありません。クロックが整列していない場合は、再ロードによって今後の 1 Hz クロックエッジが少しずれ、基準クロックと整列するようにします。

基準クロックが中断すると (ck_apre 3 周期の時間枠で基準クロックエッジが発生しない)、カレンダーは LSE クロックのみに基づいて連続的に更新されます。その後 RTC は ck_spre エッジ回りの大きな ck_apre 7 周期の検出ウィンドウを使って基準クロックを待ちます。

基準クロック検出が有効になると、PREDIV_A および PREDIV_S を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： 基準クロック検出は、**STANDBY モードではご利用いただけません。**

注意： 基準クロック検出機能は、簡易デジタル較正と併用することはできません。REFCKON = 1 の場合、RTC_CALIBR は 0x0000 0000 に保持する必要があります。

20.3.10 RTC の簡易デジタル校正

デジタル校正手法には、簡易校正とスムーズ校正の 2 種類があります。簡易校正を実施するには、[セクション 20.6.7 : RTC 校正レジスタ \(RTC_CALIBR\)](#) を参照してください。

スムーズなデジタル校正は、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。これら 2 種類の校正手法は合わせて使用することは想定されておらず、アプリケーションは 2 種類の操作方法のうち、いずれかを選択する必要があります。簡易校正は、互換性を確保するために搭載されています。スムーズ校正を実施するには、[セクション 20.3.11 : RTC の高精度デジタル校正 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) および [セクション 20.6.16 : RTC 校正レジスタ \(RTC_CALR\)](#) を参照してください。

簡易デジタル校正を使用すると、非同期プリスケアラ (ck_apre) の出力でクロックサイクルを追加 (正の校正) またはマスキング (負の校正) して、クリスタルの不正確性を補償することができます。

正および負の校正は、RTC_CALIBR レジスタの DCS ビットをそれぞれ "0" および "1" にセットして選択されます。

正の校正が有効 (DCS = "0") の場合、2XDC 分の間、毎分 (ck_apre 約 15360 周期) ck_apre 2 周期が追加されます。これによりカレンダーが早めに更新されるようになり、それによって事実上の RTC 周波数は少し高くなります。

負の校正が有効 (DCS = "1") の場合、2XDC 分の間、毎分 (ck_apre 約 15360 周期) ck_apre 1 周期が削除されます。これによりカレンダーが遅めに更新されるようになり、それによって事実上の RTC 周波数は少し低くなります。

DC は RTC_CALIBR レジスタの DC[4:0] ビットで設定されます。この数値は 0 から 31 の範囲にあり、0 から 62 にわたる時間間隔 (2xDC) に相当します。

簡易デジタル校正は、初期化モードでのみ設定でき、INIT ビットがクリアされると開始します。フル校正サイクルは、64 分続きます。64 分サイクルの最初の 2xDC 分は、今説明したように変更されます。

負の校正は、約 2 ppm の分解能で実施できますが、正の校正は約 4ppm の分解能で実施できます。最大校正範囲は、-63 ppm から 126 ppm です。

校正は LSE または HSE クロックのどちらに対しても実施可能です。

注意： デジタル校正は、PREDIV_A < 6 の場合には正しく動作しないことがあります。

RTCCLK=32.768 kHz および PREDIV_A+1=128 の場合

次の説明は、ck_apre 周波数が LSE クロック公称周波数 32.768 kHz、PREDIV_A を 127 (デフォルト値) にセットして得られた 256 Hz であることを前提としています。

ck_spre クロック周波数は、64 分サイクルの最初の 2xDC 分間でのみ変更されます。たとえば、DC が 1 である場合は、最初の 2 分のみが変更されます。これは、ck_apre の各周期が RTCCLK 128 周期であれば各 64 分サイクルの最初の 2xDC 分が、毎分一度だけ、RTCCLK 256 周期によって 1 秒短くなるか、RTCCLK 128 周期によって 1 秒長くなることを意味します (PREDIV_A+1=128)。

したがって、各校正ステップには、125829120 RTCCLK 周期 (64 分 x 60s/分 x 32768 周期/s) ごとにオシレータ 512 周期を追加またはオシレータ 256 周期を引くという効果があります。これは、校正ステップにつき +4.069 ppm または -2.035 ppm と同じです。その結果、校正分解能は毎月 +10.5 または -5.27 秒となり、総校正範囲は毎月 +5.45 から -2.72 分となります。

クロック偏差を測定するため、校正用に 512 Hz クロックが出力されます。[セクション 20.3.14 : 校正されたクロックの出力](#) を参照してください。

20.3.11 RTC の高精度デジタル較正 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)

Cat.2、Cat.3、Cat.4 および Cat.5 製品では、RTC 周波数は、-487.1 ppm から +488.5 ppm の範囲で、分解能約 0.954 ppm でデジタル較正できます。周波数は、微調整 (個々の RTCCLK パルスの加算と減算) の連続で補正されます。この調整は、短い時間で観測されたものであっても RTC が十分に較正されるようになり分散しています。

スムーズデジタル較正は、入力周波数が 32768 Hz のときに、RTCCLK 約 2^{20} パルスの周期または 32 秒で実施されます。この周期は、20ビットカウンタであり、RTCCLK がクロック供給する `cal_cnt[19:0]` で維持されます。

スムーズな較正レジスタ (RTC_CALR) は、32 秒周期でマスクされなければならない RTCCLK クロックサイクル数を指定します。

- ビット CALM[0] を 1 にセットすると、32 秒間のサイクルでちょうど 1 パルスがマスクされます。
- ビット CALM[1] 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。
- ビット CALM[2] 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] が 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_CALRx) は、32 秒周期でマスクされなければならない RTCCLK パルス数を指定します。cal_cnt[19:0] が 0x80000 のときにビット CALM[0] を "1" にセットすると、32 秒間のサイクルでちょうど 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ます (cal_cnt が 0x40000 および 0xC0000)。CALM[2]=1 ではさらに 4 サイクルがマスクされます (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ます (cal_cnt = 0xXX800)。

CALM では高分解能で RTC 周波数を最大 487.1 ppm 低下させられるのに対し、ビット CALP では周波数を 488.5 ppm 増加することができます。CALP を "1" にセットすると、RTCCLK 2^{11} 周期毎に余分な RTCCLK パルスが効果的に挿入され、これは各 32 秒間サイクルの間に 512 クロックが追加されることを意味します。

CALM を CALP と合わせて使用すると、32 秒間のサイクルで RTCCLK -511 から +512 周期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 ppm から +488.5 ppm となります。

入力周波数 (FRTCCLK) が所与の場合に有効較正周波数 (FCAL) を計算する公式は、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 未満の場合に 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 にセットされているときに PREDIV_A ビットが 3 未満の値にセットされると、CALP が無視され、較正は CALP が 0 であるかのように動作します。

PREDIV_A が 3 未満の状態では較正を実施するには、各秒が RTCCLK 8 クロック周期加速されるように同期プリスケアラ値 (PREDIV_S) を小さくします。これは 32 秒毎に 256 クロック周期を追加するのと同じです。その結果、CALM ビットのみを使用して 32 秒周期の間に 255 クロックパルスから 256 クロックパルス (243.3 から 244.1 ppm の較正範囲に相当) を効果的に追加することができます。

公称 RTCCLK 周波数 32768 Hz では、PREDIV_A が 1 (分周比 2) の時に PREDIV_S を 16383 ではなく 16379 (4 少ない) にセットします。それ以外で興味深いケースとしては、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ることくらいです。

PREDIV_S をこのように減少させると、較正入力クロックの有効周波数を使った公式は次のようになります。

$$F_{\text{CAL}} = F_{\text{RTCCLK}} \times [1 + (256 - \text{CALM}) / (2^{20} + \text{CALM} - 256)]$$

この場合、RTCCLK が正確に 32768.00 Hz であれば、CALM[7:0] が 0x100 (CALM 範囲の中心点) と等しくなるのが正しい設定です。

RTC 較正值のベリファイ

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 値を計算して実施します。Cat.2、Cat.3、Cat.4 および Cat.5 製品で オプションの 1 Hz 出力が提供されると、アプリケーションは RTC 精度を測定して確認できます。

制限された間隔で RTC の正確な周波数を測定すると、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、測定誤差は測定周期で最大 RTCCLK 2 クロック周期になります。

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。

- デフォルトにより、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、測定が 0.477 ppm (較正分解能が制限されているため、32 秒で RTCCLK 0.5 周期) 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 RTCCLK 0.5 周期) で 16 秒間測定できます。ただし、較正分解能が低下しているため、長期的な RTC 精度もまた 0.954 ppm に低下します。CALM[0] ビットは、CALW16 が 1 にセットされると 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 RTCCLK 0.5 周期) で 8 秒で測定できます。長期的な RTC 精度もまた 1.907 ppm に低下します。CALM[1:0] ビットは、CALW8 が 1 にセットされると 00 のままとなります。

オンザフライ再較正

次の処理を使用することにより、RTC_ISR/INITF=0 の間でも、較正レジスタ (RTC_CALR) をオンザフライで更新できます。

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。RECALPF が自動的に 1 にセットされます。
3. RTC_CALR への書き込み操作後 ck_apre 3 周期内に、新しい較正設定が有効になります。

20.3.12 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 にセットすると有効になります。

TIMESTAMP オルタネート機能が割り付けられている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) でカレンダーが保存されます。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 TSE ビットをセットすると、タイムスタンプイベント発生時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ると、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) がその前のイベントの結果を維持します。

注： *同期処理があるため、TSF はタイムスタンプイベント発生後 ck_apre 2 周期でセットされます。*

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が近い場合、TSF がまだ"0"であっても TSOVF が"1"と考慮される可能性があることを意味します。結果として、TSOVF のポーリングは TSF がセットされてから行うことが推奨されます。

注意： TSF ビットがクリアされるはずのときにすぐにタイムスタンプイベントが発生すると、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、アプリケーションは TSF がすでに"1"と読み出されていない限り"0"を書き込まないようにしてください。

オプションで、タンパイ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 20.6.17: RTC タンパおよび代替機能設定レジスタ \(RTC_TAFCR\)](#) を参照してください。タイムスタンプイベントがフィルタモードで設定されているタンパイイベントと同じピン上にある場合 (TAMPFLT がゼロ以外の値にセットされている場合)、RTC_TAFCR レジスタの TAMPTS を"1"にセットして、タンパ検出イベントモード上のタイムスタンプを選択する必要があります。

TIMESTAMP オルタネート機能

TIMESTAMP オルタネート機能 (RTC_TS) は、RTC_AF1 に割り付けられます。

20.3.13 タンパ検出

Cat.1 製品では、エッジ検出で 1 つのタンパ検出入力が使用できます。

Cat.2、Cat.3、Cat.4 および Cat.5 製品では、3 個のタンパ検出入力が利用できます。この入力、フィルタリングで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。

RTC バックアップレジスタ

バックアップレジスタ (RTC_BKPxR) は、20 バイトのユーザアプリケーションデータを保存するバリュールライン製品の 5 個の 32 ビットレジスタ、80 バイトのユーザアプリケーションデータを保存する Cat.1 および Cat.2 製品 (バリュールラインを除く) の 20 個の 32 ビットレジスタ、128 バイトのユーザアプリケーションデータを保存する Cat.3 (バリュールラインを除く)、Cat.4 および Cat.5 製品の 32 個の 32 ビットレジスタであり、V_{DD} ドメインに実装されています。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ず、電源オン時リセットによってリセットされます。

バックアップレジスタは、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 20.6.20: 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 \(538 ページ\)](#) 参照)。

タンパ検出の初期化

各タンパ検出入力は、RTC_ISR2 レジスタのフラグ TAMP1F/TAMP2F/TAMP3F フラグに関連付けられています。各入力は、該当する TAMP1E/TAMP2E/TAMP3E ビットを RTC_TAFCR レジスタで 1 にセットすると有効にできます。

タンパ検出イベントは、すべてのバックアップレジスタ (RTC_BKPxR) をリセットします。

RTC_TAFCR レジスタの TAMPIE ビットをセットすると、タンパ検出イベント発生時に割り込みが生成されます。

タンパイベント上のタイムスタンプ

TAMPTS を "1" にセット (高位製品のみ) すると、すべてのタンパ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、通常のタイムイベント発生時と同じように 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 がセットされると、同時に影響されるタンパフラグレジスタ (TAMP1F、TAMP2F、または TAMP3F) 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 ビットにより、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 TAMPER ピンがタンパ検出イベント (RTC_TAMP[3:1]) を生成します。エッジ検出を選択すると、TAMPER 入力上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の見落としを防ぐため、エッジ検出に使用される信号は TAMPxE と論理積をとって、TAMPERx ピンが有効になる前にすでに発生していたタンパ検出イベントも検出します。

- TAMPxTRG = 0 の場合、タンパ検出が有効になる前に TAMPERx 代替機能がすでにハイであれば (TAMPxE ビットが 1 にセットされている)、TAMPxE がセットされた後に TAMPERx に立ち上がりエッジがなくても TAMPERx が有効になると同時にタンパ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 TAMPERx 代替機能がすでにローであれば、TAMPxE がセットされた後に TAMPERx に立ち下がりエッジがなくても TAMPERx が有効になると同時にタンパイベントが検出されます。

タンパイベントが検出され、クリアされた後は、バックアップレジスタ (RTC_BKPxR) を再プログラムする前に TAMPERx 代替機能を無効にしてから再度有効にします (TAMPxE を 1 にセット)。これによって、TAMPERx の値がまだタンパ検出を示しているときに、アプリケーションがバックアップレジスタに書き込むのを防ぎます。これは、TAMPERx 代替機能でのレベル検出と同じことです。

注： タンパ検出は、 V_{DD} 電源がオフのときでも有効です。バックアップレジスタの不用意なリセットを避けるには、TAMPER 代替機能が配置されているピンを外部で適当な信号レベルに接続しておく必要があります。

タンパ入力でのフィルタを使ったレベル検出 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ)

フィルタリングを使ったレベル検出は、TAMPFLT をゼロ以外の値にセットして実施します。タンパ検出イベントは、2、4 または 8 (TAMPFLT による) 個の連続サンプルが TAMPxTRG ビット (TAMP1TRG/TAMP2TRG/TAMP3TRG) で指定するレベルで観測されると生成されます。

TAMPER 入力 は、TAMPPUDIS を 1 にセットして無効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 内部プルアップ抵抗で事前に充電されています。事前充電の持続時間は TAMPPRCH ビットによって決定され、タンパ入力でより大きな容量を可能にしています。

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 でレベル検出のサンプリング周波数を決定して最適化できます。

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。

TAMPER オルタネート機能検出

TAMPER1 代替機能 (RTC_TAMP1) は、RTC_AF1 に配置されます。TAMPER 2 および TAMPER 3 代替機能は、それぞれ RTC_TAMP2 および RTC_TAMP3 に配置されます。

20.3.14 較正されたクロックの出力

RTC_CR レジスタで COE ビットが 1 にセットされると、RTC_CALIB デバイス出力で基準クロックが提供されます。RTC_CR レジスタの COSEL ビットがリセットされ、PREDIV_A = 0x7F である場合、RTC_CALIB 周波数は $f_{RTCCLK}/64$ です。これは 32.768 kHz における RTCCLK 周波数に対する 512 Hz での較正出力に相当します。

RTC_CALIB 出力は、RTC_CALIBR レジスタでプログラムされる較正值に影響されません。RTC_CALIB デューティサイクルは不規則です。立ち下がりがエッジには、軽いジッタがあります。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 (Cat.2、Cat.3、Cat.4 および Cat.5 製品) され、「PREDIV_S+1」がゼロではない 256 の倍数である場合 (PREDIV_S[7:0] = 0xFF など)、RTC_CALIB 周波数は $f_{RTCCLK}/(256 * (PREDIV_A+1))$ となります。これは、RTCCLK 周波数が 32.768 kHz の場合のプリスケアラデフォルト値 (PREDIV_A = 0x7F、PREDIV_S = 0xFF) の 1 Hz における較正出力に相当します。

較正代替機能出力

RTC_CR レジスタの COE ビットが 1 にセットされると、較正代替機能 (RTC_CALIB) が RTC_AF1 で有効になります。

注： RTC_CALIB または RTC_ALARM を選択すると、RTC_AF1 が代替出力機能で自動的に設定されます。

20.3.15 アラーム出力

アラーム出力では、ALRAF、ALRBF、WUTF の 3 つの機能が選択できます。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 レジスタの OSEL[1:0] 制御ビットを使用して RTC_AF1 でアラーム代替機能出力 (RTC_ALARM) をアクティブにし、RTC_ALARM で出力される機能を選択します。

出力の極性は、POL が 1 にセットされるときに選択されたフラグビットの反対が出力されるよう、RTC_CR の POL 制御ビットによって決定されます。

アラーム代替機能出力

RTC_ALARM は、RTC_TAFCR レジスタの制御ビット ALARMOUTTYPE を使って出力オープンドレインまたは出力プッシュプルで設定できます。

注： *RTC_CALIB を無効にします (COE ビットをクリアにしておきます)。*

RTC_CALIB または RTC_ALARM を選択すると、RTC_AF1 が代替出力機能で自動的に選択されます。

20.4 RTC および低電力モード

表 88. 低電力モードが RTC に与える影響

モード	説明
スリープ	影響なし RTC 割込みによって、デバイスは SLEEP モードを終了します。
ストップ	RTC クロックソースが LSE または LSI のときは、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、デバイスが STOP モードを終了します。
スタンバイ	RTC クロックソースが LSE または LSI のときは、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、デバイスが STANDBY モードを終了します。

20.5 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。

RTC アラーム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 EXTI ライン 17 を設定して有効にし、立ち上りエッジ検出を選択します。
2. NVIC で RTC_Alarm IRQ チャンネルを設定し、有効にします。
3. RTC が RTC アラーム (アラーム A またはアラーム B) を発生させるように設定します。

RTC ウェイクアップ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 EXTI ライン 20 を設定して有効にし、立ち上がりエッジ検出を選択します。
2. NVIC で RTC_WKUP IRQ チャンネルを設定し、有効にします。
3. RTC が RTC ウェイクアップタイミントを発生させるように設定します。

RTC タンパ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 EXTI ライン 19 を設定して有効にし、立ち上がりエッジ検出を選択します。
2. NVIC で TAMP_STAMP IRQ チャンネルを設定し、有効にします。
3. RTC が RTC タンパイベントを検出するように設定します。

RTC タイムスタンプ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 EXTI ライン 19 を設定して有効にし、立ち上がりエッジ検出を選択します。
2. NVIC で TAMP_STAMP IRQ チャンネルを設定し、有効にします。
3. RTC が RTC タイムスタンプイベントを検出するように設定します。

表 89. 割り込み制御ビット

割り込みイベント	イベント フラグ	有効 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	○	○ ⁽¹⁾	○ ⁽¹⁾
アラーム B	ALRBF	ALRBIE	○	○ ⁽¹⁾	○ ⁽¹⁾
ウェイクアップ	WUTF	WUTIE	○	○ ⁽¹⁾	○ ⁽¹⁾
タイムスタンプ	TSF	TSIE	○	○ ⁽¹⁾	○ ⁽¹⁾
Tamper1 検出	TAMP1F	TAMPIE	○	○ ⁽¹⁾	○ ⁽¹⁾
Tamper2 検出 ⁽²⁾	TAMP2F	TAMPIE	○	○ ⁽¹⁾	○ ⁽¹⁾
Tamper3 検出 ⁽³⁾	TAMP3F	TAMPIE	○	○ ⁽¹⁾	○ ⁽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。
2. Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ (RTC_TAMPER2 ピンがある場合) デバイスデータシートのピン配置を参照してください。
3. Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ (RTC_TAMPER3 ピンがある場合) デバイスデータシートのピン配置を参照してください。

20.6 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](#) 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。

20.6.1 RTC 時間レジスタ (RTC_TR)

RTC_TR は、カレンダー時間シャドウレジスタです。このレジスタは、初期化モード以外では書き込みません。 [カレンダーの初期化と設定 \(529 ページ\)](#) および [カレンダーの読み出し \(531 ページ\)](#) を参照してください。

アドレスオフセット : 0x00

電源オン時のリセット値 : 0x0000 0000

システムリセット : 0x0000 0000 (BYP SHAD = 0) BYP SHAD = 1 のときには何も影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み									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
予約済み	MNT[2:0]			MNU[3:0]				予約済み	ST[2:0]			SU[3:0]			
	RW	RW	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW	RW

ビット 31-24 予約済み

ビット 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 **HT[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 **MNU[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 **ST[3:0]** : BCD 形式での秒の一の位

注 : 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 \(529 ページ\)](#) で説明されています。

20.6.2 RTC データレジスタ (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、初期化モード以外では書き込めません。 [カレンダーの初期化と設定 \(529 ページ\)](#) および [カレンダーの読み出し \(531 ページ\)](#) を参照してください。

アドレスオフセット : 0x04

電源オン時のリセット値 : 0x0000_2101

システムリセット : 0x0000 2101 (BYP SHAD = 0) BYP SHAD = 1 のときには何も影響を受けません。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み								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]				予約済み		DT[1:0]		DU[3:0]		
rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw

ビット 31-24 予約済み

ビット 23:20 **YT[3:0]** : BCD 形式での年の十の位

ビット 19:16 **YT[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

注 : 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 \(529 ページ\)](#) で説明されています。



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。

20.6.3 RTC 制御レジスタ(RTC_CR)

アドレスオフセット : 0x08

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み								COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
								rW	rW	rW	rW	rW	rW	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	DCE	FMT	BYPSHAD	REFCKON	TSEDGE	WUCKSEL[2:0]		
rW	rW	rW	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 **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 出力に配線さ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ピンがハイになります (OSEL[1:0] による)

1 : ALRAF/ALRBF/WUTF がアサートされるとピンがローになります (OSEL[1:0] による)

ビット 19 **COSEL** : 較正出力選択

このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

COE=1 のとき、このビットが RTC_CALIB でどの信号を出力するかを選択します。

0 : 較正出力は 512 Hz です。

1 : 較正出力は 1 Hz です。

この周波数は、RTCCLK 32.768 kHz に、またデフォルト値でのプリスケアラに有効です (PREDIV_A=127 および PREDIV_S=255)。セクション 20.3.14: 較正されたクロックの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、ユーザーがサマータイムの変更を実施したかどうかを記憶しておくために書き込みます。

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でセットすると、現在時間が 0 でない場合にカレンダー時間から 1 時間を引きます。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 : 影響ありません

1 : 現在時間から 1 時間引きます。これは、冬時間変更で使用できます。

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でセットすると、現在時間が 0 でない場合にカレンダー時間に 1 時間足します。このビットは常に 0 として読み出されます。

0 : 影響ありません

1 : 現在時間に 1 時間足します。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 DCE : 簡易デジタル較正イネーブル

0 : デジタル較正は無効です。

1 : デジタル構成は有効です。

PREDIV_A は 6 以上である必要があります。

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 : シャドウレジスタバイパス

このビットは Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

0 : 2 RTCCLK サイクル毎に更新されるシャドウレジスタからカレンダー値を取得します (RTC_SSR、RTC_TR、RTC_DR からの読み出し時)。

1 : カレンダーカウンタから直接カレンダー値を取得します (RTC_SSR、RTC_TR、RTC_DR からの読み出し時)。

注 : APB1 クロックの周波数が RTCCLK の周波数の 7 倍よりも低い場合は、BYP SHAD を "1" にセットする必要があります。

ビット 4 **REFCKON** : 基準クロック検出イネーブル (50 または 60 Hz)

0 : 基準クロック検出は無効です。

1 : 基準クロック検出は有効です。

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 : タイムスタンプ立ち上がりエッジがタイムスタンプイベントを生成します。

1 : タイムスタンプ立ち下がりエッジがタイムスタンプイベントを生成します。

TSE は、TSEDGE が変更されたらリセットし、TSF の不要な設定を回避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 クロックを選択します

001 : RTC/8 クロックを選択します

010 : RTC/4 クロックを選択します

011 : RTC/2 クロックを選択します

10x : ck_spre (通常は 1 Hz) クロックを選択します

11x : ck_spre (通常は 1 Hz) クロックを選択し、 2^{16} が WUT カウンタ値に追加されます (下記注参照)

注 : **WUT = ウェイクアップユニットカウンタ値**
WUT = (0x0000 から 0xFFFF) + 0x10000
(WUCKSEL[2:1 = 11] の場合追加されます。)

初期化モードの場合 (RTC_ISR/INITF = 1) のみ、このレジスタのビット 7、6、4 が書き込めます。
このレジスタのビット 2 から 0 は、RTC_CR WUTE ビット = 0 および RTC_ISR WUTWF ビット = 1 の場合にのみ書き込めます。

カレンダー時増加中に時間を変更するとカレンダー時の増加がマスクされる可能性があるため、増加中には時間を変更しないことが推奨されます。

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 (529 ページ) で説明されています。

20.6.4 RTC 初期化およびステータスレジスタ (RTC_ISR)

アドレスオフセット : 0x0C

電源オン時のリセット値 : 0x0000 0007

システムリセット値 : 0 にクリアされる INIT、INITF、RSF 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	
予約済み															RECALPF	
															r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TAMP 3F	TAMP 2F	TAMP 1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	INITS	SHPF	WUT WF	ALRB WF	ALRA WF	
rc_w0	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

ビット 16 **RECALPF** : 再較正保留フラグ

RECALPF ステータスフラグは、ソフトウェアが RTC_CALR レジスタがブロックされていることを示す"1"を RTC_CALR レジスタに書き込むと自動的にセットされます。新しい較正設定が考慮されると、このビットは"0"に戻ります。詳細は [セクション : オンザフライ再較正](#) を参照してください。

ビット 15 **TAMP3F** : TAMPER3 検出フラグ

このフラグは、タンパ入力3でタンパ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。
ソフトウェアが 0 を書き込むと、クリアされます。

ビット 14 **TAMP2F** : TAMPER2 検出フラグ

このフラグは、タンパ入力2でタンパ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。
ソフトウェアが 0 を書き込むと、クリアされます。

ビット 13 **TAMP1F** : タンパ検出フラグ

このフラグは、タンパ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。
ソフトウェアが 0 を書き込むと、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 ビットをクリアしてからチェックし、クリアすることを推奨します。そうでない場合、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すると、オーバーフローが通知されない可能性があります。

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 になったときに、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によってクリアされる必要があります。



ビット 9 ALRBF : アラーム B フラグ

このフラグは、時間/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) に一致する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、時間/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) に一致する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 : 時間および日付レジスタ (RTC_TR および RTC_DR)、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、時間、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 (BYPHAD=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、ソフトウェアによってもクリアできます。

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、カレンダ年フィールドが 0 ではないとき (電源オン時のリセット値状態) にハードウェアによってセットされます。

0 : カレンダが初期化されていません。

1 : カレンダが初期化されています。

ビット 3 SHPF : シフト操作保留中

0 : シフト操作は保留されていません。

1 : シフト操作が保留されています。

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するとハードウェアによってセットされ、該当するシフト操作が実行されるとクリアされます。SHPF へ書き込んでも何も起きません。

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 **WUTWF** : ウェイクアップタイム書き込みフラグ

このビットは、WUTE ビットが RTC_CR で 0 にセットされた後にウェイクアップタイム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ム設定を更新できません。

1 : ウェイクアップタイム設定を更新できます。

ビット 1 **ALRBWF** : アラーム B 書き込みフラグ

このビットは、ALRBIE ビットが RTC_CR で 0 にセットされた後にアラーム 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 を更新できません。

1 : アラーム B を更新できます。

ビット 0 **ALRAWF** : アラーム A 書き込みフラグ

このビットは、ALRAE ビットが RTC_CR で 0 にセットされた後にアラーム 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 を更新できません。

1 : アラーム A を更新できます。

注 : **ALRAF、ALRBF、WUTF および TSF ビットは、0 にプログラムされてから APB クロック 2 周期後にクリアされます。**

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 (529 ページ) で説明されています。

20.6.5 RTC プリスケアラレジスタ (RTC_PRER)

アドレスオフセット : 0x10

電源オン時のリセット値 : 0x007F 00FF

システムリセット : 影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み									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 31:24 予約済み

ビット 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。

$ck_{spre} \text{ 周波数} = ck_{apre} \text{ 周波数} / (PREDIV\_S + 1)$

注 : **PREDIV[14:13] は Cat.1 製品で予約されます。**



注： このレジスタは、初期化モード以外では書き込めません。初期化は、2 つの別々の書き込みアクセスで実施する必要があります。[カレンダーの初期化と設定 \(529 ページ\)](#) 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 \(529 ページ\)](#) で説明されています。

20.6.6 RTC ウェイクアップタイマレジスタ (RTC_WUTR)

アドレスオフセット : 0x14

電源オン時のリセット値 : 0x0000 FFFF

システムリセット : 影響されません

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
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

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)のときは、WUTF フラグが(WUT[15:0] + 1) ck_wut 周期毎にセットされます。ck_wut 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。

注： WUTF の最初のアサートは、WUTE がセットされてから (WUT+1) ck_wut 周期後に発生します。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。

注： このレジスタは、WUTWF が RTC_ISR で 1 にセットされるとき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 \(529 ページ\)](#) で説明されています。

20.6.7 RTC 較正レジスタ (RTC_CALIBR)

アドレスオフセット : 0x18

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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								DCS	予約済み		DC[4:0]				
								rW			rW	rW	rW	rW	rW

ビット 31:8 予約済み

ビット 7 **DCS** : デジタル較正符号

0 : 正較正 : カレンダ更新頻度が上昇します。

1 : 負較正 : カレンダ更新頻度が低下します。

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 4:0 **DC[4:0]** : デジタル較正

DCS = 0 (正較正)

00000 : + 0 ppm

00001 : + 4 ppm (概数値)

00010 : + 8 ppm (概数値)

..

11111 : + 126 ppm (概数値)

DCS = 1 (負較正)

00000 : - 0 ppm

00001 : - 2 ppm (概数値)

00010 : - 4 ppm (概数値)

..

11111 : - 63 ppm (概数値)

正確なステップ値については、[RTCCLK=32.768 kHz および PREDIV_A+1=128 の場合 \(534 ページ\)](#)を参照してください。

注 : このレジスタは、初期化モード (**RTC_ISR/INITF="1"**) の場合のみ書き込みます。

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 \(529 ページ\)](#) で説明されています。

20.6.8 RTC アラーム A レジスタ (RTC_ALRMAR)

アドレスオフセット : 0x1C

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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 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] は無視します。

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 時間マスク
0 : 時間が一致すると、アラーム A がセットされます。
1 : アラーム A の比較では時間を無視します。

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 **HT[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 **MNU[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 **ST[3:0]** : BCD 形式での秒の一の位

注 : このレジスタは、**ALRAWF** が **RTC_ISR** で 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 \(529 ページ\)](#) で説明されています。

20.6.9 RTC アラーム B レジスタ (RTC_ALRMBR)

アドレスオフセット : 0x20

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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 : 日付と曜日が一致すると、アラーム B がセットされます。
- 1 : アラーム B の比較では日付と曜日を無視します。

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] は無視します。

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 時間マスク

- 0 : 時間が一致すると、アラーム B がセットされます。
- 1 : アラーム B の比較では時間を無視します。

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 **HT[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 **MNU[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 B 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 **ST[3:0]** : BCD 形式での秒の一の位

注 : このレジスタは、ALRBWF が RTC_ISR で 1 にセットされる場合、または初期化モードの場合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 (529 ページ) で説明されています。

20.6.10 RTC 書き込み保護レジスタ (RTC_WPR)

アドレスオフセット : 0x24

電源オン時のリセット値 : 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								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によって書き込まれます。
 このバイトは、読み出すと必ず 0x00 を返します。
 RTC レジスタの書き込み保護を解除する方法については、[RTC レジスタ書き込み保護](#)を参照してください。

20.6.11 RTC サブセカンドレジスタ (RTC_SSR)

このレジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット : 0x28

電源オン時のリセット値 : 0x0000 0000

システムリセット : 0x0000 0000 (BYP SHAD = 0) BYP SHAD = 1 のときには何も影響を受けません。

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
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

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の値です。1 秒の分数値は、以下の式で与えられます。

$$\text{サブセカンド} = (\text{PREDIV\_S} - \text{SS}) / (\text{PREDIV\_S} + 1)$$

注 : **SS が PREDIV_S よりも大きくなる可能性があるのは、シフト操作の後のみです。その場合、正しい時間/日付は RTC_TR/RTC_DR の示す値よりも 1 秒遅いことになります。**

20.6.12 RTC シフト制御レジスタ (RTC_SHIFTR)

RTC_SHIFTR レジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット：0x2C

電源オン時のリセット値：0x0000 0000

システムリセット：影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADD1S	予約済み														
w	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.	SUBFS[14:0]														
r	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

ビット 31 **ADD1S**：1 秒追加

0：影響ありません。

1：クロック／カレンダーに 1 秒追加します。

このビットは書き込み専用であり、常にゼロとして読み出されます。シフト操作が保留中に (RTC_ISR で SHPF = 1 の場合) このビットに書き込んでも何の影響もありません。

この機能は、アトミックな操作でクロックに効果的にサブセカンドを追加できるように、SUBFS (下記説明を参照) と合わせて使うことを目的としています。

ビット 30:15 予約済み

ビット 14:0 **SUBFS**：サブセカンドの差し引き

このビットは書き込み専用であり、常にゼロとして読み出されます。シフト操作が保留中に (RTC_ISR で SHPF = 1 の場合) このビットに書き込んでも何の影響もありません。

SUBFS に書き込まれる値が同期プリスケアラのカウンタに追加されます。このカウンタはカウントダウンしていくため、この操作によってクロックを次のように効果的に遅らせることができます。

遅らせる (秒) = $SUBFS / (PREDIV\_S + 1)$

SUBFS と合わせて ADD1S 機能を使用すると、サブセカンドを効果的にクロックに追加 (クロックを進める) でき、次のようにして効果的にクロックを進めます。

進める (秒) = $(1 - (SUBFS / (PREDIV\_S + 1)))$

注： **SUBFS に書き込むと RSF がクリアされます。ソフトウェアは、シャドウレジスタがシフトされた時間で確実に更新されるように、RSF = 1 となるまで待つことができます。**

[セクション 20.3.8：RTC 同期 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) を参照してください。

注： **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、次で説明されています：RTC レジスタ書き込み保護 (529 ページ)**

20.6.13 タイムスタンプ時刻レジスタ (RTC_TSTR)

アドレスオフセット : 0x30

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み									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
予約済み	MNT[2:0]			MNU[3:0]				予約済み	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 **HT[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 **MNU[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 **ST[3:0]** : BCD 形式での秒の一の位

注 : このレジスタの内容は、TSF が RTC_ISR で 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。

20.6.14 RTC タイムスタンプ日付レジスタ (RTC_TSDR)

アドレスオフセット : 0x34

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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
WDU[1:0]			MT	MU[3:0]				予約済み			DT[1:0]		DU[3:0]		
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

注 : このレジスタの内容は、TSF が RTC_ISR で 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。

20.6.15 RTC タイムスタンプサブセカンドレジスタ (RTC_TSSSR)

RTC_TSSSR レジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット : 0x38

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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
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

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発生時の同期プリスケアラのカウンタの値です。

注 : 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。

20.6.16 RTC 校正レジスタ (RTC_CALR)

RTC_CALR レジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット：0x3C

電源オン時のリセット値：0x0000 0000

システムリセット：影響されません

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
CALP	CALW8	CALW16	予約済み				CALM[8:0]								
rW	rW	rW	r	r	r	r	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 31:16 予約済み

ビット 15 CALP：RTC の周波数を 488.5 ppm 増加

0：RTCCLK パルスを追加しません。

1：2¹¹ パルスごとに RTCCLK パルスを 1 つ効果的に挿入します (488.5 ppm で増加した周波数)。

この機能は、カレンダーの周波数を高分解能で低下させる CALM と合わせて使用することを目的としています。入力周波数が 32768 Hz の場合、32 秒の間に追加される RTCCLK パルス数は (512 * CALP) - CALM で計算されます。

[セクション 20.3.11：RTC の高精度デジタル校正 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) を参照してください。

ビット 14 CALW8：8 秒校正サイクル周期の使用

CALW8 を“1”にセットすると、8 秒校正サイクル周期が選択されます。

CALW8 = “1”のとき、CALM[1:0] は「00」で止まります。

[セクション 20.3.11：RTC の高精度デジタル校正 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) を参照してください。

ビット 13 CALW16：16 秒校正サイクル周期の使用

CALW16 を“1”にセットすると、16 秒校正サイクル周期が選択されます。このビットは、CALW8 = 1 のときは“1”にセットしないでください。

注： CALW16 = “1”のとき、CALM[1:0] は“0”で止まります。

[セクション 20.3.11：RTC の高精度デジタル校正 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) を参照してください。

ビット 12:9 予約済み

ビット 8:0 CALM[8:0]：校正マイナス

2²⁰ 個の RTCCLK パルス (入力周波数 32768 Hz の 32 秒間) の中から CALM をマスクしてカレンダーの周波数を下げます。これによってカレンダーの周波数が分解能 0.9537 ppm で低下します。

カレンダーの周波数を増加させるには、この機能を CALP と合わせて使用します。

[セクション 20.3.11：RTC の高精度デジタル校正 \(Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ\)](#) (535 ページ) を参照してください。

注： 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、次で説明されています：[RTC レジスタ書き込み保護 \(529 ページ\)](#)



20.6.17 RTC タンパおよび代替機能設定レジスタ (RTC_TAFCR)

アドレスオフセット : 0x40

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み													ALARMOUTTYPE	予約済み	
													rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TAMP-PUDIS	TAMP-PRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]			TAMP TS	TAMP 3TRG	TAMP 3E	TAMP 2TRG	TAMP 2E	TAMPIE	TAMP 1TRG	TAMP 1E
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。予約済み、常に 0 が読み出されます。

ビット 18 **ALARMOUTTYPE** : RTC_ALARM 出力タイプ

0 : ALARM_AF0 はオープンドレイン出力です。

1 : ALARM_AF0 はプッシュプル出力です。

ビット 17:16 予約済み、常に 0 として読み出されます。

ビット 15 **TAMP-PUDIS** : タンパブルアップディセーブル

このビットは、各タンパピンをサンプリング前に事前充電するかどうかを判定します。

0 : サンプリング前にタンパピンを充電します (内部ブルアップが有効です)。

1 : タンパピンの事前充電は無効です。

注： このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ利用可能です。ビット 14:13 **TAMP-PRCH[1:0]** : タンパ事前充電持続時間

このビットは、各サンプリング前にブルアップがアクティブとなる持続時間を決定します。

TAMP-PRCH は各タンパ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

注： このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。ビット 12:11 **TAMPFLT[1:0]** : タンパフィルタカウント

このビットは、タンパイベントをアクティブにするのに必要な指定レベルにおける連続サンプリング回数 (TAMP*TRG) を決定します。

0x0 : タンパはアクティブレベルへのタンパ入力遷移エッジでアクティブになります (タンパ入力での内部ブルアップなし)。

0x1 : タンパはアクティブレベルでの連続 2 回のサンプリング後にアクティブになります。

0x2 : タンパはアクティブレベルでの連続 4 回のサンプリング後にアクティブになります。

0x3 : タンパはアクティブレベルでの連続 8 回のサンプリング後にアクティブになります。

注： このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 10:8 **TAMPFREQ[2:0]** : タンパサンプリング周波数

各タンパ入力をサンプリングする周波数を決定します。

- 0x0 : RTCCLK / 32768 (RTCCLK = 32768 Hz の時に 1 Hz)
- 0x1 : RTCCLK / 16384 (RTCCLK = 32768 Hz の時に 2 Hz)
- 0x2 : RTCCLK / 8192 (RTCCLK = 32768 Hz の時に 4 Hz)
- 0x3 : RTCCLK / 4096 (RTCCLK = 32768 Hz の時に 8 Hz)
- 0x4 : RTCCLK / 2048 (RTCCLK = 32768 Hz の時に 16 Hz)
- 0x5 : RTCCLK / 1024 (RTCCLK = 32768 Hz の時に 32 Hz)
- 0x6 : RTCCLK / 512 (RTCCLK = 32768 Hz の時に 64 Hz)
- 0x7 : RTCCLK / 256 (RTCCLK = 32768 Hz の時に 128 Hz)

注: これらのビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 7 **TAMPTS** : タンパ検出イベントでタイムスタンプをアクティブにする

- 0 : タンパ検出イベントでタイムスタンプを保存しません。
 - 1 : タンパ検出イベントでタイムスタンプを保存します。
- TAMPTS は、RTC_CR レジスタで TSE=0 でも有効です。

注: このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 6 **TAMP3TRG** : タンパ 2 のアクティブレベル

TAMPFLT != 00 (Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ) の場合:

- 0 : TAMPER3 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 : TAMPER3 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 : TAMPER3 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 : TAMPER3 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 5 **TAMP3E** : タンパ 3 検出イネーブル

- 0 : タンパ 3 検出は無効です。
- 1 : タンパ 3 検出は有効です。

ビット 4 **TAMP2TRG** : タンパ 2 のアクティブレベル

TAMPFLT != 00 (Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ) の場合:

- 0 : TAMPER2 がローのままだとタンパ検出イベントがトリガされます。
- 1 : TAMPER2 がハイのままだとタンパ検出イベントがトリガされます。

TAMPFLT = 00 の場合:

- 0 : TAMPER2 立ち上がりエッジがタンパ検出イベントをトリガします。
- 1 : TAMPER2 立ち下がりエッジがタンパ検出イベントをトリガします。

注: このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

ビット 3 **TAMP2E** : タンパ 2 検出イネーブル

- 0 : タンパ 2 検出は無効です。
- 1 : タンパ 2 検出は有効です。

注: このビットは、Cat.2、Cat.3、Cat.4 および Cat.5 製品でのみ使用可能です。

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。

ビット 1 **TAMP1TRG** : タンパ 1 のアクティブレベル

TAMPFLT != 00 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ) の場合

0 : TAMPER1 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 : TAMPER1 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 : TAMPER1 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 : TAMPER1 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。

注意 : **TAMPFLT = 0 の場合、TAMP1TRG を変更するときは TAMP1E をリセットして TAMP1F を誤ってセットしないようにします。**

ビット 0 **TAMP1E** : タンパ 1 検出イネーブル

0 : タンパ 1 検出は無効です。

1 : タンパ 1 検出は有効です。

20.6.18 RTC アラーム A サブセカンドレジスタ (RTC_ALRMASR)

RTC_ALRMASR レジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット : 0x44

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み				MASKSS[3:0]				予約済み							
r	r	r	r	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
予約済み	SS[14:0]														
r	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	W	rW

ビット 31:28 予約済み

ビット 27:24 **MASKSS[3:0]** : このビットで開始する最上位ビットをマスク

0 : アラーム A のサブセカンドを比較しません。アラームは秒単位が増加するとセットされます(フィールドの残りが一致すると想定)。

1 : アラーム A の比較では SS[14:1] を無視し、SS[0] のみ比較します。

2 : アラーム A の比較では SS[14:2] を無視し、SS[1:0] のみ比較します。

3 : アラーム A の比較では SS[14:3] を無視し、SS[2:0] のみ比較します。

...

12 : アラーム A の比較では SS[14:12] を無視し、SS[11:0] のみ比較します。

13 : アラーム A の比較では SS[14:13] を無視し、SS[12:0] のみ比較します。

14 : アラーム A の比較では SS[14:14] を無視し、SS[13:0] のみ比較します。

15 : 15 個の SS ビットすべてを比較し、一致する場合にアラームをアクティブにします。

同期カウンタのオーバーフロービット (ビット 15) は絶対に比較しません。このビットは、シフト操作後でなければ 0 以外の値にはなりません。

ビット 23:15 予約済み

ビット 14:0 **SS[14:0]** : サブセカンド値

この値を同期プリスケアラのカウンタの内容と比較し、アラーム A をアクティブにするかどうかを決定します。比較されるのはビット 0 から MASKSS-1 までです。

注： このレジスタは、ALRAE が RTC_CR レジスタでリセットされる場合、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、次で説明されています：[RTC レジスタ書き込み保護 \(529 ページ\)](#)

20.6.19 RTC アラーム B サブセカンドレジスタ (RTC_ALRMBSSR)

RTC_ALRMBSSR レジスタは、Cat.2、Cat.3、Cat.4 および Cat.5 製品で使用可能です。

アドレスオフセット：0x48

電源オン時のリセット値：0x0000 0000

システムリセット：影響されません

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み				MASKSS[3:0]				予約済み							
r	r	r	r	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
予約済み	SS[14:0]														
r	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	W	rW

ビット 31:28 予約済み

ビット 27:24 **MASKSS[3:0]**：このビットで開始する最上位ビットをマスク

0x0：アラーム B のサブセカンドを比較しません。アラームは秒単位が増加するとセットされます（フィールドの残りが一致すると想定）。

0x1：アラーム B の比較では SS[14:1] を無視し、SS[0] のみ比較します。

0x2：アラーム B の比較では SS[14:2] を無視し、SS[1:0] のみ比較します。

0x3：アラーム B の比較では SS[14:3] を無視し、SS[2:0] のみ比較します。

...

0xC：アラーム B の比較では SS[14:12] を無視し、SS[11:0] のみ比較します。

0xD：アラーム B の比較では SS[14:13] を無視し、SS[12:0] のみ比較します。

0xE：アラーム B の比較では SS[14:14] を無視し、SS[13:0] のみ比較します。

0xF：15 個の SS ビットすべてを比較し、一致する場合にアラームをアクティブにします。

同期カウンタのオーバーフロービット（ビット 15）は絶対に比較しません。このビットは、シフト操作後でなければ 0 以外の値にはなりません。

ビット 23:15 予約済み

ビット 14:0 **SS[14:0]**：サブセカンド値

この値を同期プリスケアラのカウンタの内容と比較し、アラーム B をアクティブにするかどうかを決定します。比較されるのはビット 0 から MASKSS-1 までです。

注： このレジスタは、ALRBIE が RTC_CR レジスタでリセットされる場合、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。書き込みアクセス手順は、次で説明されています：[セクション：RTC レジスタ書き込み保護](#)



20.6.20 RTC バックアップレジスタ (RTC_BKPxR)

アドレスオフセット : 0x50 から 0x60 (バリュールライン製品)

アドレスオフセット : 0x50 から 0x9C (Cat.1 および Cat.2 製品)

アドレスオフセット : 0x50 から 0xCC (Cat.3、Cat.4 および Cat.5 製品)

電源オン時のリセット値 : 0x0000 0000

システムリセット : 影響されません

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、このレジスタにデータを書き込んだり、レジスタからデータを読み出すことができます。レジスタは、V_{DD} がオフになるとオンになりません。また、システムリセットによってリセットされず、デバイスが低電力モードで動作するときはレジスタの内容はそのまま有効です。このレジスタは、TAMPx_F=1 である限り、またはフラッシュ読み出し保護が無効になるとタンパ検出イベントでリセットされます。

20.6.21 RTC レジスタマップ

表 90. 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	予約済み										PM	HT [1:0]		HU[3:0]			予約済み	MNT[2:0]			MNU[3:0]			予約済み	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
0x04	RTC_DR	予約済み										YT[3:0]			YU[3:0]			WDU[2:0]			MT	MU[3:0]			予約済み	DT [1:0]		DU[3:0]						
	リセット値																				0	0	1	0		0	0	0	1	0	0	0	0	1
0x08	RTC_CR	予約済み										COE	OSEL [1:0]		POL	COSEL	BKP	SUB1H	ADD1H	TSIE	WUTIE	ALRBIE	ALRAIE	TSE	WUTE	ALRBE	ALRAE	DCE	FMT	BYPHAD	REFCKON	TSEDGE	WCKSEL [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	0
0x0C	RTC_ISR	予約済み																	TAMP3F	TAMP2F	TAMP1F	TSOVF	TSF	WUTF	ALRBF	ALRAF	INIT	INITF	RSF	SHPF	WUTFW	ALRBWF	ALRAWF	
	リセット値																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x10	RTC_PRER	予約済み										PREDIV_A[6:0]						予約済み	PREDIV_S[14:0]															
	リセット値											1	1	1	1	1	1		1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1
0x14	RTC_WUTR	予約済み																	WUT[15:0]															
	リセット値																		1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0x18	RTC_CALIBR	予約済み																							DCS	予約済み			DC[4:0]					
	0																								0							0	0	0
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		
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	予約済み																							KEY[7:0]									
	0																															0	0	0
0x28	RTC_SSR	予約済み																	SS[15:0]															
	0																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x2C	RTC_SHIFTR	ADD1S	予約済み																	SUBFS[14:0]														
	0	0																		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x30	RTC_TSTR	予約済み										PM	HT [1:0]		HU[3:0]			予約済み	MNT[2:0]			MNU[3:0]			予約済み	ST[2:0]			SU[3:0]					
	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	予約済み																	WDU[2:0]			MT	MU[3:0]			予約済み	DT [1:0]		DU[3:0]					
	0																		0	0	0	0	0	0	0		0	0	0	0	0	0	0	0



表 90. 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									
0x38	RTC_TSSSR	予約済み																SS[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
0x3C	RTC_CALR	予約済み																CALP	CALW8	CALW16	予約済み				CALM[8:0]																	
	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	
0x40	RTC_TAFCR	予約済み														ALARMOUTTYPE	予約済み	TAMPPUDIS	TAMPPRCH[1:0]		TAMPFLT[1:0]		TAMPFREQ[2:0]		TAMPTS	TAMP3TRG	TAMP3E	TAMP2TRG	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	0	0	0	0		
0x44	RTC_ALRMASSR	予約済み				MASKSS[3:0]				予約済み								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	0	0	0	0	0	0	0	0	0	0
0x48	RTC_ALRMBSSR	予約済み				MASKSS[3:0]				予約済み								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	0	0	0	0	0	0	0	0	0	0
0x50 から 0xCC	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	0								
	から RTC_BKP31R	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	0								

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

注意：

21 独立型ウォッチドッグ (IWDG)

21.1 IWDG の概要

STM32L1xxxx は、2 つの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。2 つのウォッチドッグペリフェラル（独立型とウィンドウ型）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、システムリセットまたは割り込み（ウィンドウ型ウォッチドッグのみ）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有効なままです。ウィンドウ型ウォッチドッグ (WWDG) クロックは、APB1 クロックからプリスケールされ、設定可能な時間枠 (time-window) を持つので、異常に遅いアプリケーション動作や異常に速いアプリケーション動作を検出するようにプログラミングできます。

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。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。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 22 \(573 ページ\)](#) を参照してください。

21.2 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 オシレータからクロック供給 (STANDBY および STOP モードで動作可能)
- ダウンカウンタが 0x000 に達したときにリセット (ウォッチドッグが有効な場合)

21.3 IWDG の機能詳細

[図 166](#) 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。

キーレジスタ (IWDG_KR) に値 0xCCCC 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 0xAAAA が書き込まれると、IWDG_RLR レジスタ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。

21.3.1 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通じて「ハードウェアウォッチドッグ」機能が有効な場合、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の書き込みが行われない限り、リセットを生成します。

21.3.2 レジスタのアクセス保護

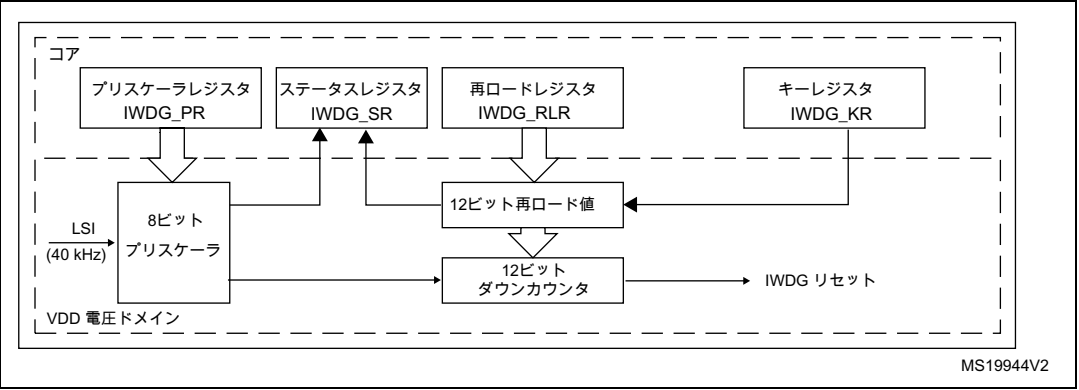
IWDG_PR および IWDG_RLR レジスタへの書き込みアクセスは保護されます。これらを変更するには、まず、IWDG_KR レジスタにコード 0x5555 を書き込む必要があります。このレジスタに別の値を書き込むと、シーケンスはブレイクされ、レジスタへのアクセスが再び保護されます。これは、再ロード操作 (0xAAAA の書き込み) であることを意味します。

ステータスレジスタは、プリスケアラやダウンカウンタ再ロード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。

21.3.3 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 (Cortex®-M3コアは停止)、IWDG カウンタは、DBG モジュールの DWG_IWDG_STOP 設定ビットに応じて、通常通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 30.16.2 : 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。

図 166. 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、V_{DD} 電圧ドメインで実装され、STOP モードおよびSTANDBY モードでも機能します。

表 91. 37 kHz での最小/最大 IWDG タイムアウト期間 (LSI)⁽¹⁾

プリスケアラ 分周回路	PR[2:0] ビット	タイムアウト最小値 (ms) RL[11:0]= 0x000	タイムアウト最大値 (ms) RL[11:0]= 0xFFFF
/4	0	0.108	442.81
/8	1	0.216	885.621
/16	2	0.432	1771.243
/32	3	0.864	3542.486
/64	4	1.729	7084.972
/128	5	3.459	14169.945
/256	6	6.918	28339.891

1. これらのタイミングは、37 kHz クロックが供給されたときの値ですが、マイクロコントローラの内蔵 RC 周波数は 30 kHz~60 kHz で変化することがあります。さらに、RC オシレータ周波数が正確だとしても、タイミング精度は、APB インタフェースクロックと LSI クロックの位相によっても左右されるので、最大で RC クロック 1 周期分の誤差が生じます。

21.4 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、を参照してください。

ペリフェラルレジスタは、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。

21.4.1 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 (スタンバイモードによるリセット)

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
予約済み																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、読み出しは 0000h）

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 AAAAh が書き込まれなければなりません。さもなければ、カウンタが 0 に達すると、ウォッチドッグがリセットを生成します。

キー値 5555h を書き込むことによって、IWDG_PR および IWDG_RLR レジスタへのアクセスが可能になります（[セクション 21.3.2](#) を参照）。

キー値 CCCCh を書き込むと、ウォッチドッグが開始されます（ハードウェアウォッチドッグオプションを選択した場合は除きます）。

21.4.2 プリスケーラレジスタ (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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み																													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]** : プリスケーラ分周比

これらのビットは、書き込みアクセス保護がなされています ([セクション 21.3.2](#) を参照)。カウンタクロックを供給するプリスケーラ分周回路を選択するために、ソフトウェアによって書き込まれます。プリスケーラ分周回路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ーラ値が返されます。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です。

21.4.3 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 (スタンバイモードによるリセット)

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
予約済み																				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がなされています ([セクション 21.3.2](#) を参照)。IWDG_KR レジスタに値 AAAAh 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ラによって決まります。[表 91](#) 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です。

21.4.4 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 (スタンバイ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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み																													RVU	PVU	
																													r	r	

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 **RVU** : ウォッチドッグカウンタ再ロード値更新

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動作が完了したときに、ハードウェアからリセットされます (最大 5 RC 40 kHz サイクルかかります)。

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 サイクルかかります)。

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 : 複数の再ロード値またはプリスケアラ値がアプリケーションで使用される場合は、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必要があります、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必要があります。ただし、プリスケアラ値や再ロード値を更新した後は、RVU または PVU 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でも、書き込み操作は実行され、完了します)。

21.4.5 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 92. 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	予約済み																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	予約済み																											PR[2:0]				
	リセット値																												0	0	0		
0x08	IWDG_RLR	予約済み															RL[11:0]																
	リセット値																1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
0x0C	IWDG_SR	予約済み																											RVU	PVU			
	リセット値																												0	0			

レジスタ境界アドレスについては、[表 5 \(47 ページ\)](#) を参照してください。

22 ウィンドウ型ウォッチドッグ (WWDG)

22.1 WWDG の概要

ウィンドウ型ウォッチドッグは、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更新しない限り、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）が更新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）の間にカウンタが更新されなければならないことを意味します。

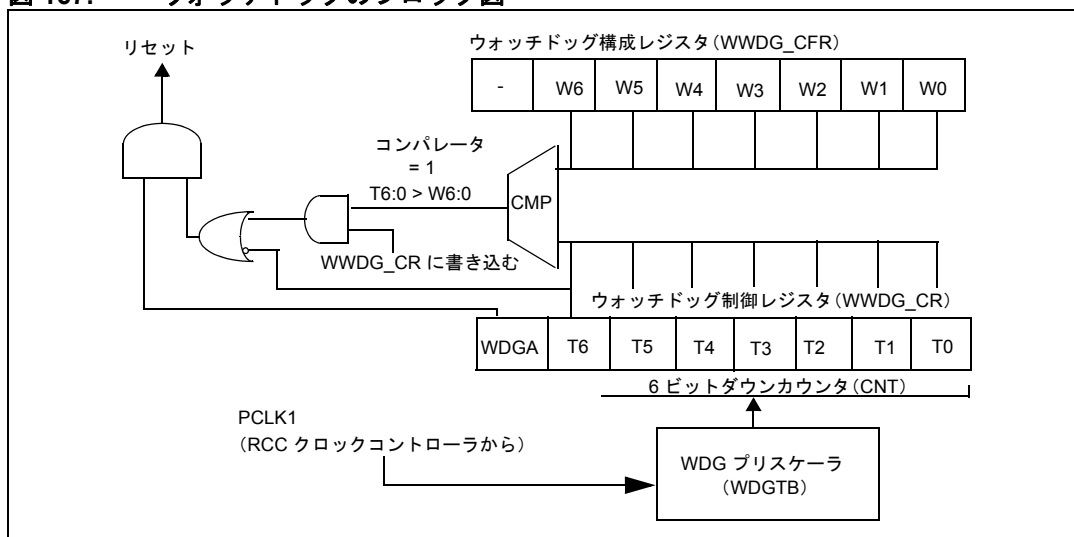
22.2 WWDG の主な機能

- 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 168 を参照）。
- アーリー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）

22.3 WWDG の機能詳細

ウォッチドッグが有効であり（WWDG_CR レジスタの WDGA ビットがセット）、7 ビットのダウンカウンタ（T[6:0] ビット）が 0x40 から 0x3F にロールオーバーすると（T6 がクリア）、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 167. 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~0xC0 の範囲である必要があります。

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化されています。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。

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、リセットの生成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 レジスタに書き込むときのプリスケアラの状態は不明なので、タイミングは最小値から最大値まで変化します (図 168 を参照)。設定レジスタ (WWDG_CFR) はウィンドウの上限値を格納しています。リセットを防ぐには、ダウンカウンタの再ロードは、カウンタの値がウィンドウレジスタの値より小さく、0x3F より大きいときに行う必要があります。図 168 に、ウィンドウ型ウォッチドッグ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ます)。

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、アーリー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/迅速な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 カウンタをリロードし、WWDG リセットを回避してから必要な操作をトリガする必要があります。

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。

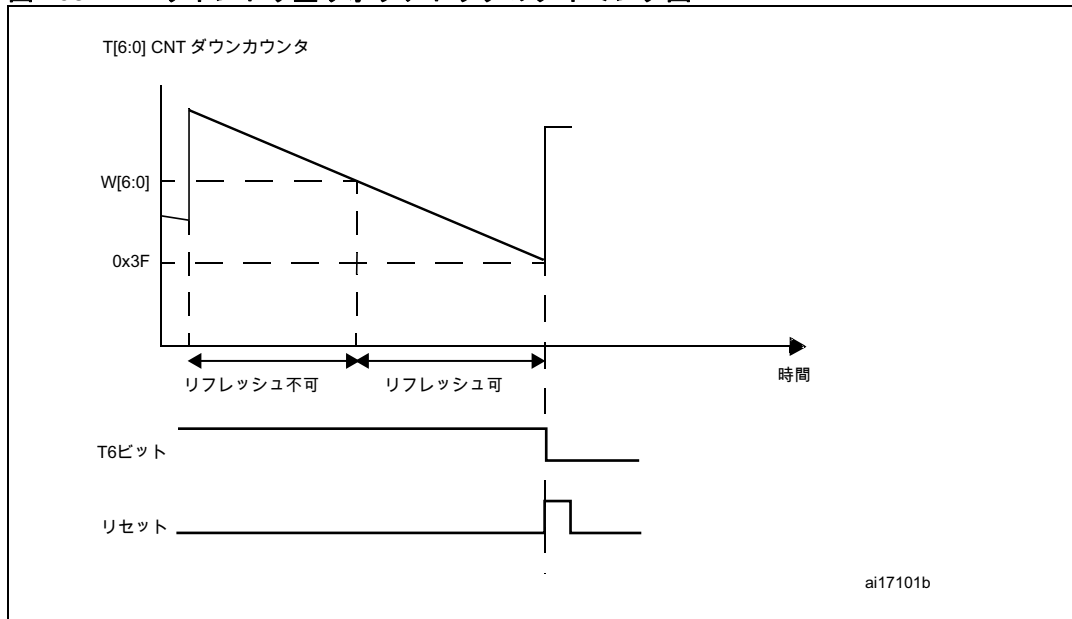
注： 優先度の高いタスクにおけるシステムロックなどによって EWI 割り込みが使用できない場合、最終的には WWDG リセットが生成されます。

22.4 ウォッチドッグタイムアウトをプログラム方法

図 168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： ただちにリセットされるのを防ぐために、WWDG_CR レジスタに書き込むときには、常に T6 ビットに 1 を書き込んでください。

図 168. ウィンドウ型ウォッチドッグのタイミング図



タイムアウト値を計算するための式：

$$t_{\text{WWDG}} = t_{\text{PCLK1}} \times 4096 \times 2^{\text{WDGTB}} \times (t[5:0] + 1) \quad (\text{ms})$$

ここで、

t_{WWDG} : WWDG タイムアウト

t_{PCLK1} : APB1 クロック周期 (ms)

T_{WWDG} の最小値と最大値については、下の表を参照してください。

表 93. タイムアウトの最小値と最大値 @32 MHz (f_{PCLK1})

プリスケアラ	WDGTB	最小タイムアウト値	最大タイムアウト値
1	0	128 μs	8.19 ms
2	1	256 μs	16.38 ms
4	2	512 μs	32.67 ms
8	3	1024 μs	65.54 ms

22.5 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入ると（Cortex®-M3コアは停止）、WWDG カウンタは、DBG モジュールの DWG_IWDG_STOP 設定ビットに応じて、通常通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 30.16.2 : 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。

22.6 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。

22.6.1 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み								WDGA	T[6:0]						
								rs	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 **WDGA** : 有効化ビット

このビットは、ソフトウェアによってセットされ、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~LSB)

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}) PCLK1 サイクルごとにデクリメントされます。0x40 から 0x3F にロールオーバーすると (T6 がクリアされると)、リセットが生成されます。

22.6.2 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み							EWI		WDGTB[1:0]			W[6:0]			
							rs		rw			rw			

ビット 31:10 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 **EWI** : アーリー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からのみクリアされます。

ビット 8:7 **WDGTB[1:0]** : タイムベース

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 (PCLK1 ÷ 4096) ÷ 1

01 : CK カウンタクロック (PCLK1 ÷ 4096) ÷ 2

10 : CK カウンタクロック (PCLK1 ÷ 4096) ÷ 4

11 : CK カウンタクロック (PCLK1 ÷ 4096) ÷ 8

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。

22.6.3 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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み														EWIF	
														rc_w0	

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 **EWIF** : アーリー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”を書き込んだときには、ビットの値に影響を与えません。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。

22.6.4 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	予約済み																								WDGA	T[6:0]								
	リセット値																									0	1	1	1	1	1	1	1		
0x04	WWDG_CFR	予約済み																						EWI	WDGTB1	WDGTB0	W[6:0]								
	リセット値																							0	0	0	1	1	1	1	1	1	1		
0x08	WWDG_SR	予約済み																																	EWIF
	リセット値																																		0

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

23 高度暗号化標準ハードウェアアクセラレータ (AES)

このセクションは、STM32L162xx デバイスにのみ適用されます。

23.1 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データの暗号化と復号化の両方に使用されます。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)。

このアクセラレータは、128 ビット長のキーを使用して 128 ビットのブロックの暗号化と復号化を行います。キーの派生を行うこともできます。暗号化または復号化キーは、同じキーを使って複数のデータブロックを処理する際に CPU または DMA による書き込み操作を最小限に抑えるために内部レジスタに保存されます。

デフォルトでは、ECB (電子コードブック) モードが選択されています。ハードウェアでは、CBC (Cipher Block Chaining) または CTR (Counter) モード連結アルゴリズムもサポートしています。

AES は、受信データと送信データの両方で DMA 転送をサポートします (2 つの DMA チャンネルが必要)。

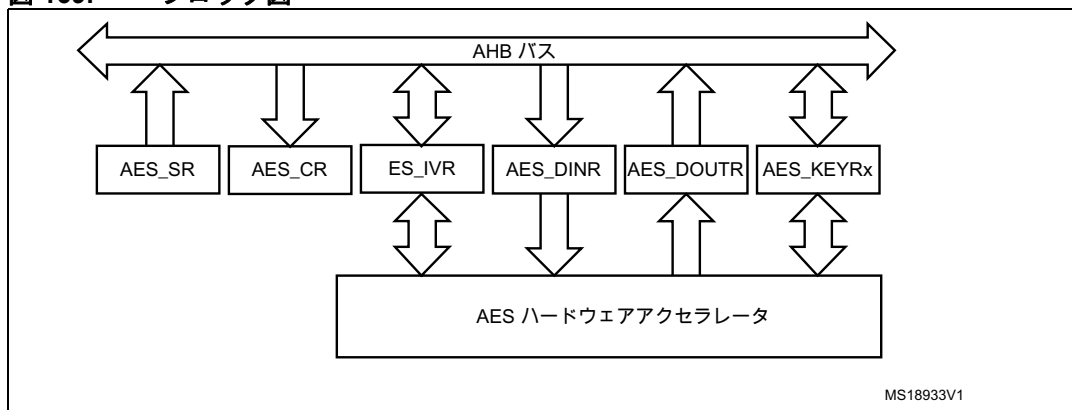
23.2 AES の主な機能

- 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
- 暗号化または復号化キーを保存する内部 128ビットレジスタ (32ビットレジスタ 4 個)
- ECB (電子コードブック)、CBC (Cipher Block Chaining)、CTR (Counter)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
- 213 クロック周期で 1 つの 128ビットブロック (入出力フェーズを含む) を暗号化または復号化
- 32ビットの入力バッファ 1 個と 32ビットの出力バッファ 1 個
- 32ビットデータ幅のみをサポートするレジスタアクセス
- AES を CBC モードで設定する際の初期化ベクタ用または CTR モードが選択されたときの 32 ビットカウンタ初期化用の 128ビットレジスタ 1 個
- 2 つのチャンネル (受信データ用と送信データ用) を使用したダイレクトメモリアクセス (DMA) による自動データフロー制御

23.3 AES の機能詳細

図 169 に、AES アクセラレータのブロック図を示します。

図 169. ブロック図



AES アクセラレータは、CBC または CTR 連結モードが選択されたときに 128 ビット長のキーと初期化ベクタを使用し、128ビット（4ワード）のデータブロックを処理します。

動作モードは 4 種類あります。

- モード 1：AES_KEYRx レジスタに保存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 レジスタに保存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保存されるキー派生。このモードは AES 連結モード選択とは独立しています。
- モード 3：AES_KEYRx レジスタに保存された所与の（事前に計算された）復号化キーを使用した復号。
- モード 4：キー派生と、AES_KEYRx レジスタに保存された暗号化キーを使った復号化（連結アルゴリズムを実行するために AES がカウンタモードで設定されるときには使用しない）。

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。モードは、AES が無効なとき（AES_CR レジスタの EN ビット=0）のみ変更できます。KEY レジスタ（AES_KEYRx）は、AES を有効にする前に保存する必要があります。

暗号ソリューションに使用する ECB、CBC または CTR モードのいずれかを選択するには、AES が無効になったときに（AES_CR レジスタの EN=0）、AES_CR レジスタの CHMOD[1:0] ビットおよび AES_IVR レジスタ（CBC および CTR 連結モードにのみ使用）に書き込まなければなりません。

有効になると（EN ビット=1）、AES は入力フェーズになり、モード 1、3 または 4 に対してソフトウェアが入力データワードを AES_DINR（4ワード）に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。ウェイトサイクルが AES_DINR レジスタへの 2 つの連続した書き込みの間に自動的に挿入され、データとインターリーブされたキーを AES プロセッサに送信します。

モード 2 の場合は、AES_CR レジスタの EN ビットがセットされた後すぐにキー派生処理が開始されます。AES を有効にする前に、AES_KEYRx レジスタに暗号化された KEY をロードしておく必要があります。キー派生処理（CCF フラグをセット）の最後に AES_KEYRx レジスタで派生キーが利用可能になり、AES はハードウェアによって無効になります。このモードでは、ハードウェアによって CCF フラグが 1 にセットされるまでは、AES が有効なときに AES_KEYRx レジスタを読み出さないようにしてください。

計算フェーズが完了すると、AES_SR レジスタのステータスフラグ CCF（計算完了フラグ）がセットされます。AES_CR レジスタの CCFIE ビット=1 の場合、割り込みを生成できます。その後ソフト

ウェアは、AES_DOUTR レジスタ（モード 1、3、4）または AES_KEYRx レジスタ（モード 2 が選択されている場合）からデータを読み出すことができます。

フラグ CCF は、AES_CR レジスタの DMAOUTEN=1 のときには意味を持ちません。これは、AES_DOUTR レジスタの読み出し動作は、ソフトウェアが計算フェーズの最後に介入しなくても、DMAによって自動的に管理されるからです。

動作は出力フェーズで終了します。モード 1、3、または 4 では、このフェーズでソフトウェアが AES_DOUTR レジスタから出力データワード 4 つを連続して読み出します。モード 2（キー派生モード）では、データが自動的に AES_KEYRx レジスタに保存され、AES がハードウェアによって無効になります。その後、ソフトウェアは AES が有効になる前にモード 3（復号化モード）を選択し、この派生キーを使って復号を開始することができます。

入力フェーズと出力フェーズの間、ソフトウェアは、データバイトの読み出しまたは書き込みを連続して行う必要があります（モード 2 を除く）、AES は各読み出しまたは書き込み動作の間に発生する遅延に対して耐性があります（この時点で別の割り込みを処理する場合など）。

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動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ると割り込みを生成できます。AES は、エラー検出後も無効にならず、通常どおり処理を続行します。

汎用 DMA を使用して、入力ワードを書き込んだり、出力ワードを読み出したりすることも可能です（図 184 および図 185 を参照）。

AES は、AES_CR レジスタの EN ビットをリセットすることによって、いつでも再初期化できます。その場合、AES は、EN=1 をセットすることによって始めから再スタートでき、最初の入力データバイトが書き込まれるのを待ちます（EN ビットがセットされるとすぐに AES_KEYRx レジスタに保存されている値からキー派生処理が開始となるモード 2 の場合を除く）。

23.4 暗号化キーと復号化キー

AES_KEYRx レジスタを使用して暗号化または復号化キーを保存します。これらの 4 つのレジスタは、リトルエンディアン設定で整理されています。レジスタ AES_KEYR0 には、キーの 32 ビットの最下位バイト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 ビットの最上位バイトをロードする必要があります。

暗号化または復号化用のキーは、AES が無効なとき（AES_CR レジスタで EN=0）にこれらのレジスタに保存される必要があります。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。計算フェーズの最後に派生キーが自動的に AES_KEYRx レジスタに保存され、その前の暗号化キーを上書きします。派生キーが利用可能になると、AES がハードウェアによって無効になります。ソフトウェアで 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、内容が派生キー（以前にモード 2 で計算されている）に一致するのであれば AES_KEYRx レジスタに書き込む必要はありません。

モード 4（キー派生と復号化）では、AES_KEYRx には暗号化キーしか入っていません。派生キーは、これらのレジスタに書き込むことなく内部で計算されます。

23.5 AES 連結アルゴリズム

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ており、AES が無効なとき (ビット EN=0) に、AES_CR レジスタの CHMOD[1:0] ビットで選択できます。

- ECB (電子コードブック)
- CBC (Cipher Block Chaining)
- CTR (カウンタモード)

23.5.1 ECB (電子コードブック)

これはデフォルトのモードです。このモードでは、AES_IVR レジスタを使用しません。また、連結動作もありません。メッセージはブロックに分割され、各ブロックは別々に暗号化されます。

図 170 および 図 171 は、暗号化および復号化の ECB アルゴリズムの原理をそれぞれ説明しています。

図 170. ECB 暗号化モード

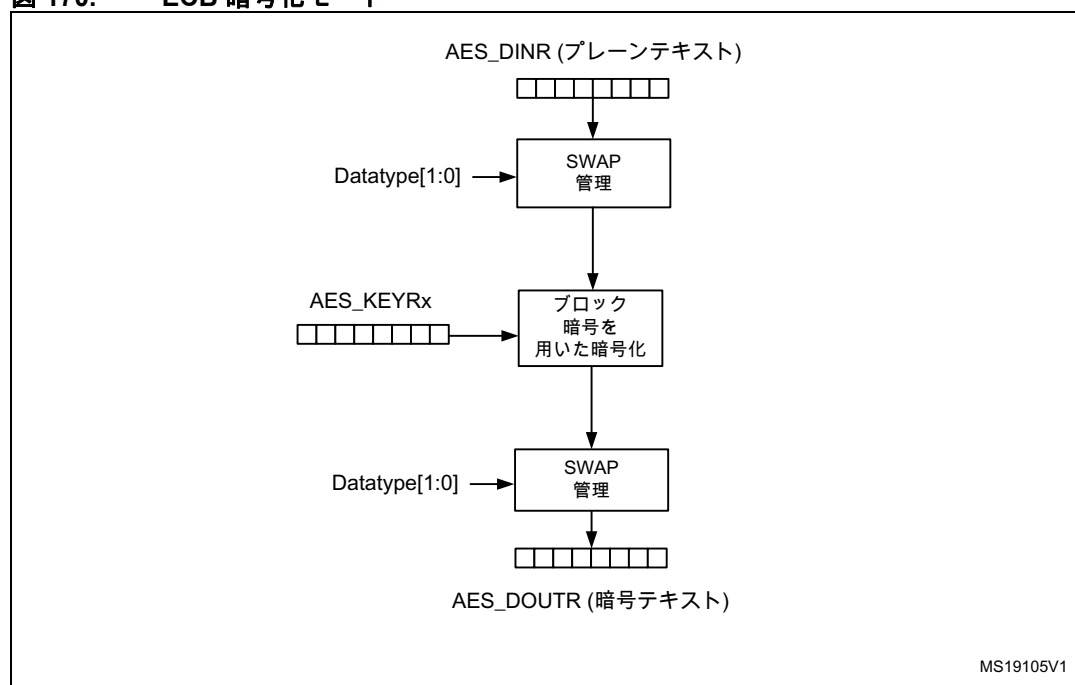
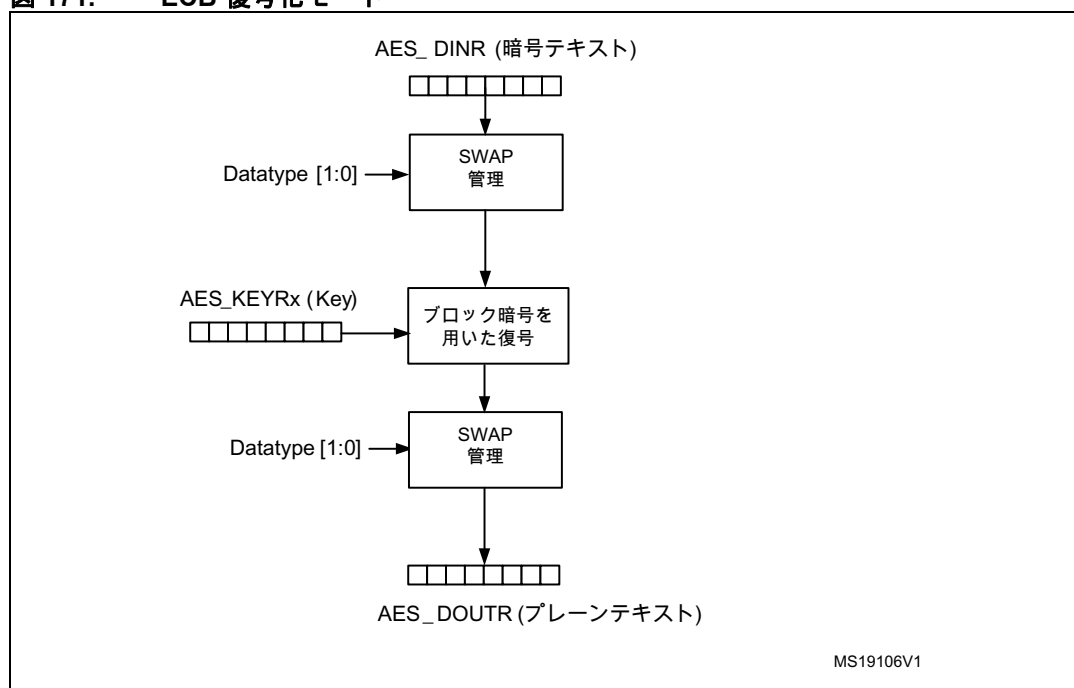


図 171. ECB 復号化モード



23.5.2 CBC (Cipher Block Chaining)

CBC モードでは、プレーンテキストの各ブロックが暗号化される前にその前の暗号テキストブロックとの排他的論理和を取ります。各メッセージが一意となるようにするため、最初のブロックの処理中に初期化ベクタ (AES_IVRx) を使用します。

暗号化モードで管理ブロックを交換した後、復号モードに入る前に初期化ベクタとの排他論理和を取ります (図 172 および 図 173 参照)。

図 172. CBC モードの暗号化

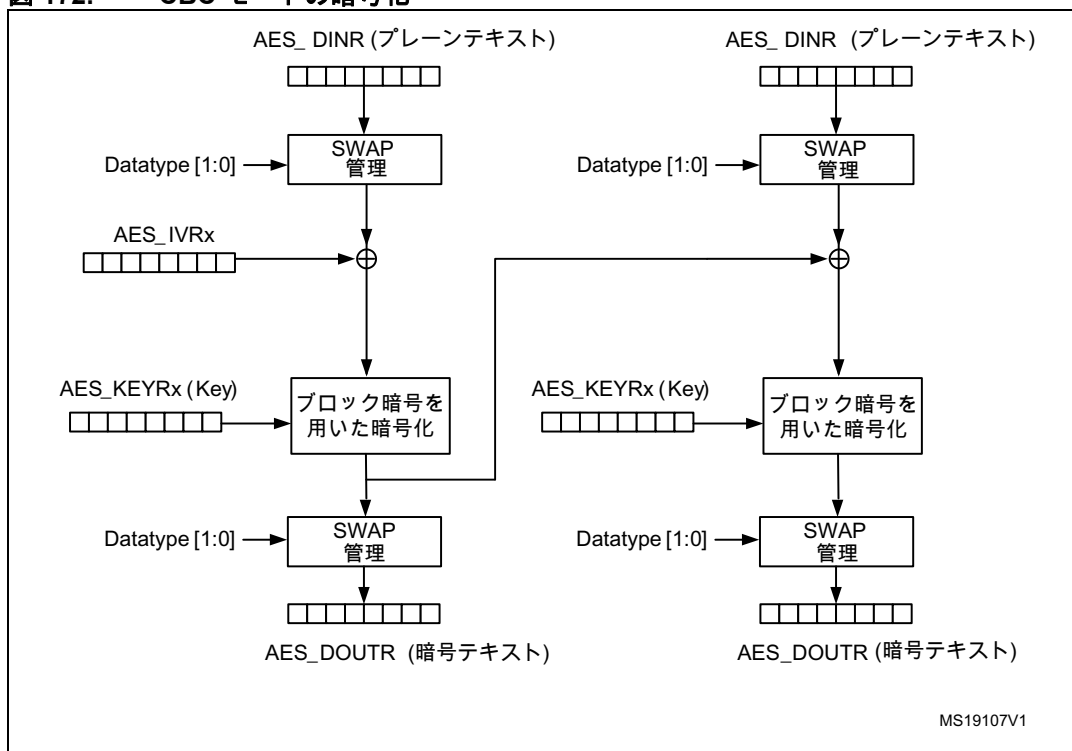
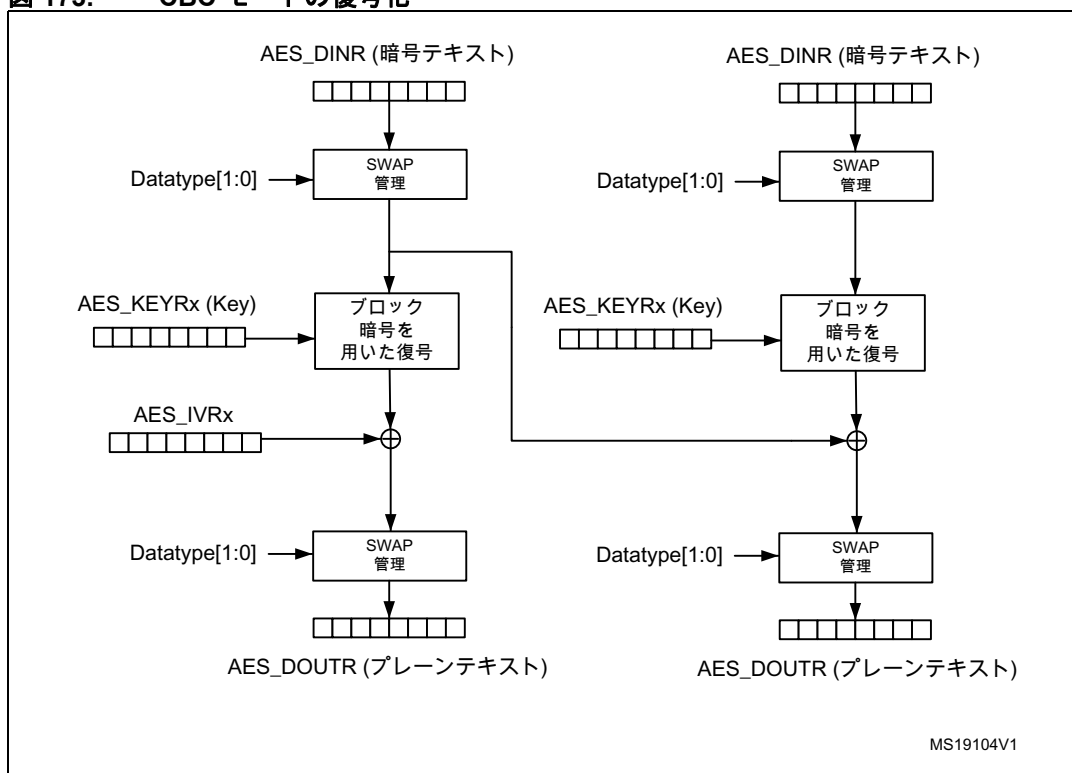


図 173. CBC モードの復号化



注: AES が有効なときに AES_IVR を読み出すと、値 0x00000000 が返ります。

所与のメッセージのサスペンドモード

優先度の高い別のメッセージを処理しなければならない場合には、メッセージをサスペンドすることができます。この優先度の高いメッセージを送信した後、暗号化または復号化モードの両方でサスペンドされたメッセージを再開できます。この機能は、CPU アクセスによって AES_DOUTR および AES_DINR レジスタへのデータ転送が実施される場合にのみ利用できます。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ようにしてください。

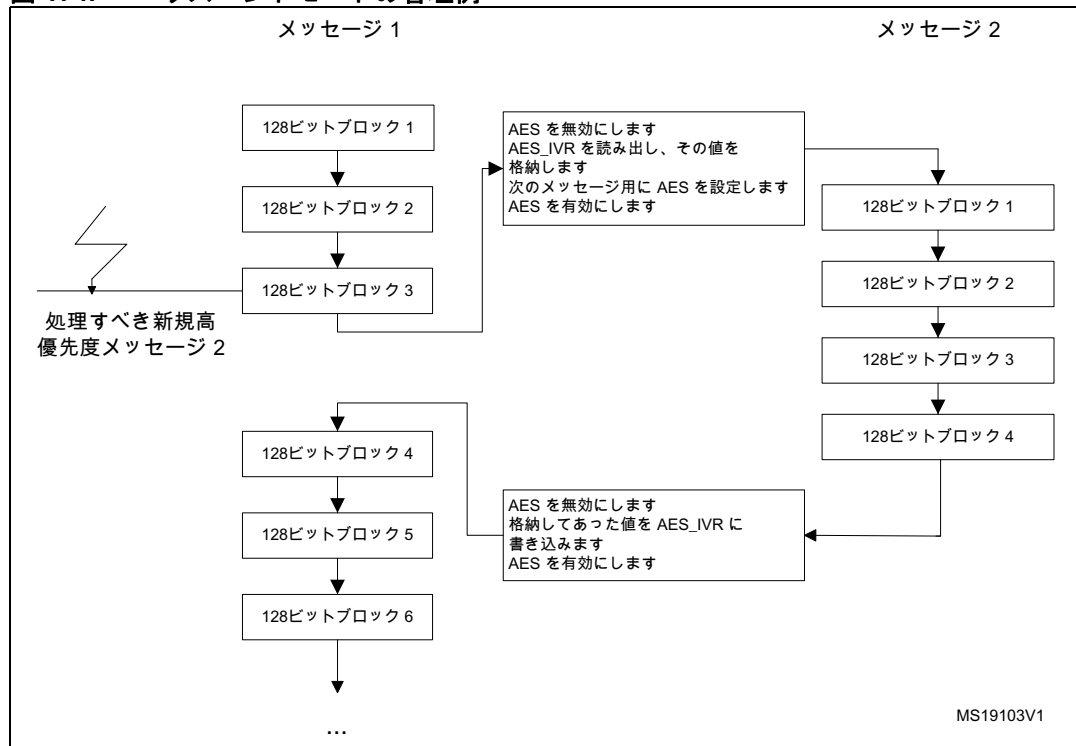
正しく動作させるためには、ブロックの処理終了時 (AES_DOUTR レジスタの 4 回目の読み出しの後で、処理する次のブロックの入力に対応する次の AES_DINR 書き込みアクセスの前) にメッセージをサスペンドする必要があります。

AES_CR レジスタの EN ビット=0 を書き込んで、AES を無効にします。ソフトウェアは、最新の値を含む AES_IVRx を読み出し、メッセージの割り込み前に XOR 演算の連結に使用します。この値は、割り込まれたメッセージが再開しなければならなくなったら (AES が無効になったら) すぐに AES_IVRx レジスタに書き込んで再使用できるように保存する必要があります。これによって連結動作が中断することはなく、次の 128 ビットのデータブロックを送信するために、再び AES が有効になるとメッセージの処理を再開できます。

この振る舞いは、AES の設定 (暗号化モードまたは復号化モード) に関わらず有効です。

図 174 は、メッセージ 1 より優先度が高く、短いメッセージ 2 を送信するためにサスペンドされたメッセージ 1 の例を示しています。128 ビットのブロック処理の最後に、AES が無効になります。AES_IVR レジスタを読み出して保存し、連結動作を中断しないように後でメッセージが再開したら取得します。その後、AES はメッセージ 2 を送信するように設定され、処理が開始されるように有効になります。メッセージ 2 の処理の最後に、AES をもう一度無効にし、メッセージ 1 が割り込まれたときに AES_IVRx レジスタに前に保存した値をロードする必要があります。その後、AES が有効になってメッセージ 1 を再開できるようになったら、すぐにブロック 4 に該当する入力値からソフトウェアを再起動する必要があります。

図 174. サスペンドモードの管理例



23.5.3 CTR (カウンタモード)

カウンタモードでは、暗号テキストまたはプレーンテキストとの XOR 演算用のノンス値に加え、32 ビットのカウンタを使用します (図 175 および 図 176 参照)。

図 175. CTR モードの暗号化

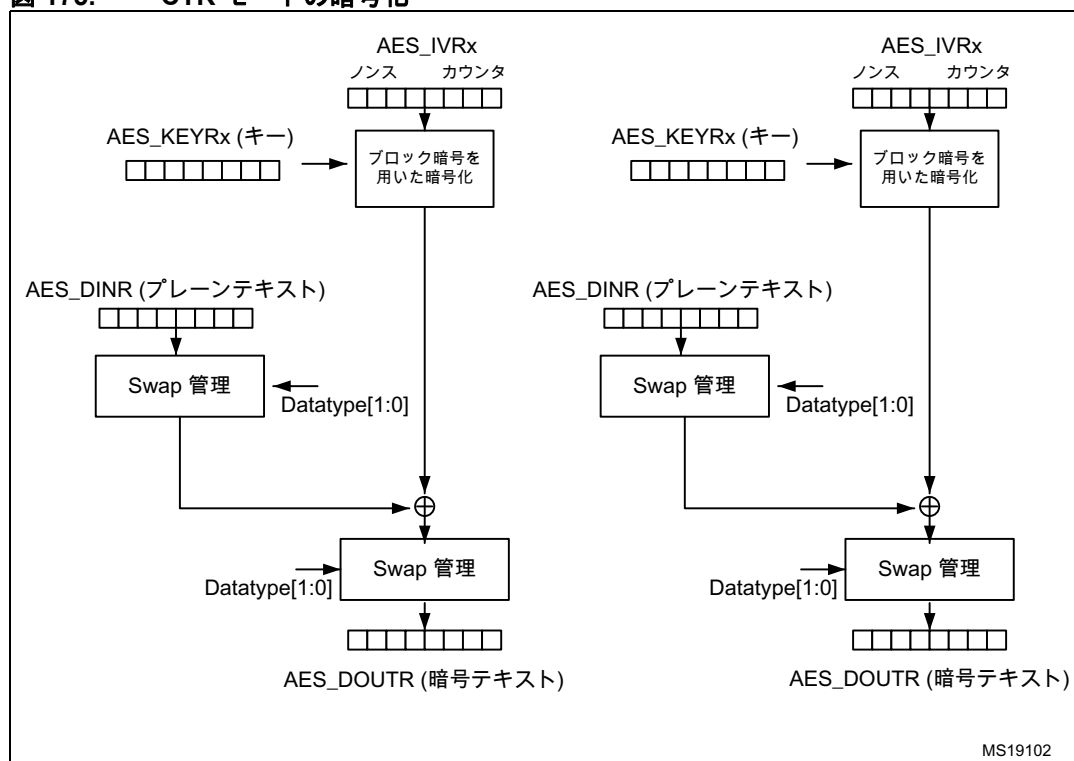
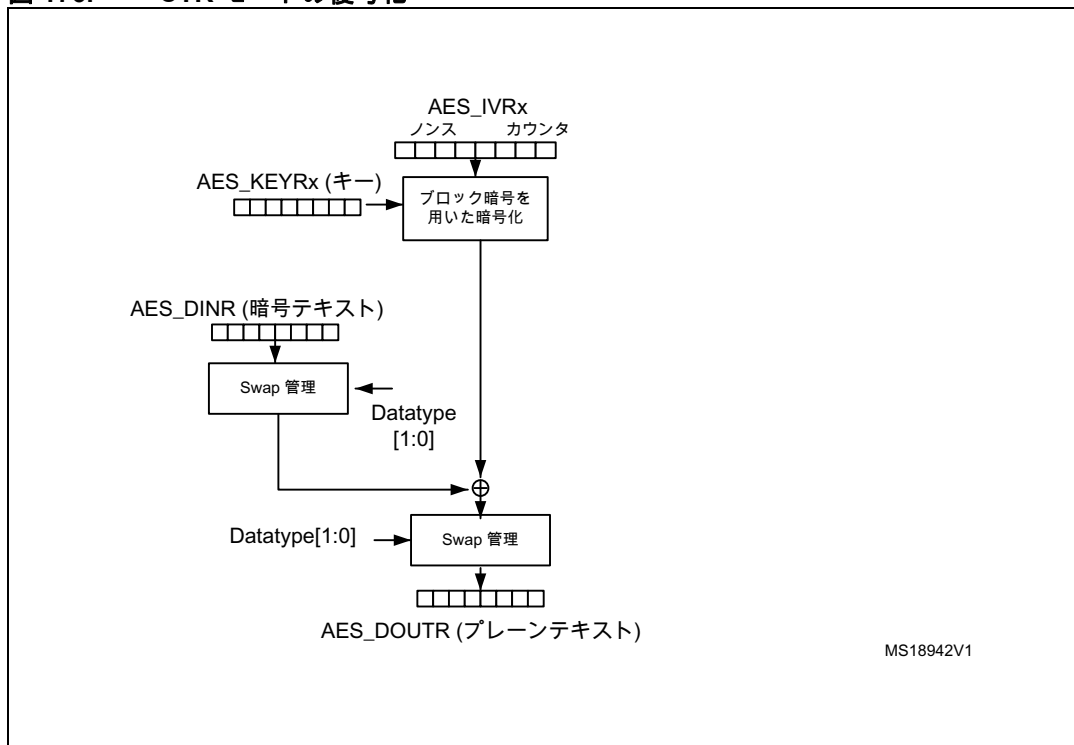
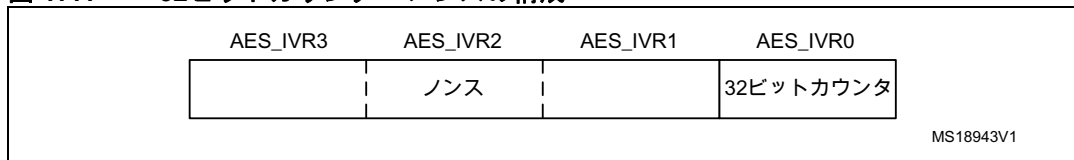


図 176. CTR モードの復号化



ノンス値と 32ビットのカウンタは、AES_IVRx レジスタからアクセスでき、図 177 に示すように構成されています。

図 177. 32ビットカウンタ + ノンスの構成



カウンタモードでは、一意のシーケンスを保証するため、カウンタは処理される各ブロックの初期化された値からインクリメントします。このシーケンスは長くは繰り返されません。これは 32ビットカウンタであり、つまりノンスメッセージは AES が無効になったら保存されている初期化値に維持されます。カウンタを表すのは、128ビットの初期化ベクタの 32ビットの最下位バイトのみです。最初のデータブロックを処理する際に AES_IVRx レジスタを 1 度だけ使用する CBC モードとは反対に、カウンタモードでは各データブロックの処理に AES_IVRx レジスタを使用します。

カウンタモードでは、キー派生と復号化モードは適用できません。

注： AES_IVRx レジスタは、AES が無効なとき (EN ビット = 0) にのみ書き込んで良好な AES 動作を保証する必要があります。

AES が有効なときに読み出すと、値 0x00000000 が返ります。

AES が無効なときに読み出すと、最新のカウンタ値が返ります (サスペンドモードの管理に有用)。

CTR モードでは、キー派生と復号化は何の目的も果たしません。結果として、AES_CR レジスタで **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すると強制的に **MODE[1:0] = 10** (CTR モード復号化に該当) となります。これは AES プロセッサの暗号化ブロックを使用し、図 176 に示すように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 モードの場合と同様、メッセージに割り込み、優先度の高いメッセージを送信して割り込まれたメッセージを再開することができます。サスペンドモード機能についての詳細は、[図 174](#) および [章 23.5.2](#) を参照してください。

23.6 データタイプ

データは、AES_DINR レジスタに書き込むことで AES プロセッサの 32ビットに 1 つ (ワード) ずつ入力されます。AES は 128ビットのデータブロックを処理します。1 つの 128ビットのデータブロックを処理するには、まず最上位バイトから AES_DINR または AES_DOUTR レジスタを 4 回読み出し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。使用されるデータタイプに関わらず (ビット、バイト、16ビットハーフワード、32ビットワード)、最下位データが最下位アドレス位置を占有します。

したがって、システムメモリから AES_DINR に書き込まれるデータに対し、AES プロセッサに入る前にビット、バイト、ハーフワードの交換操作を実施する必要があります。また、暗号化または復号化されるデータの種類により、同じ交換操作を AES_DOUTR レジスタからシステムメモリに読み出される AES データにも実施する必要があります。

AES_CR レジスタの DATATYPE ビットは、AES プロセッサに送信される前に AES_DINR レジスタに適用され、プロセッサから出力されるデータに対して AES_DOUTR レジスタに適用されるさまざまな交換モードを提供しています ([図 178](#) 参照)。

注： 交換操作は、AES_DOUTR および AES_DINR レジスタのみに関与します。AES_KEYRx および AES_IVRx レジスタは、選択される交換モードに影響されません。これらのレジスタには、固定のリトルエンディアン設定があります ([セクション 23.4](#) および [セクション 23.12](#) 参照)。

図 178. データタイプに準じた 128ビットブロックの構成

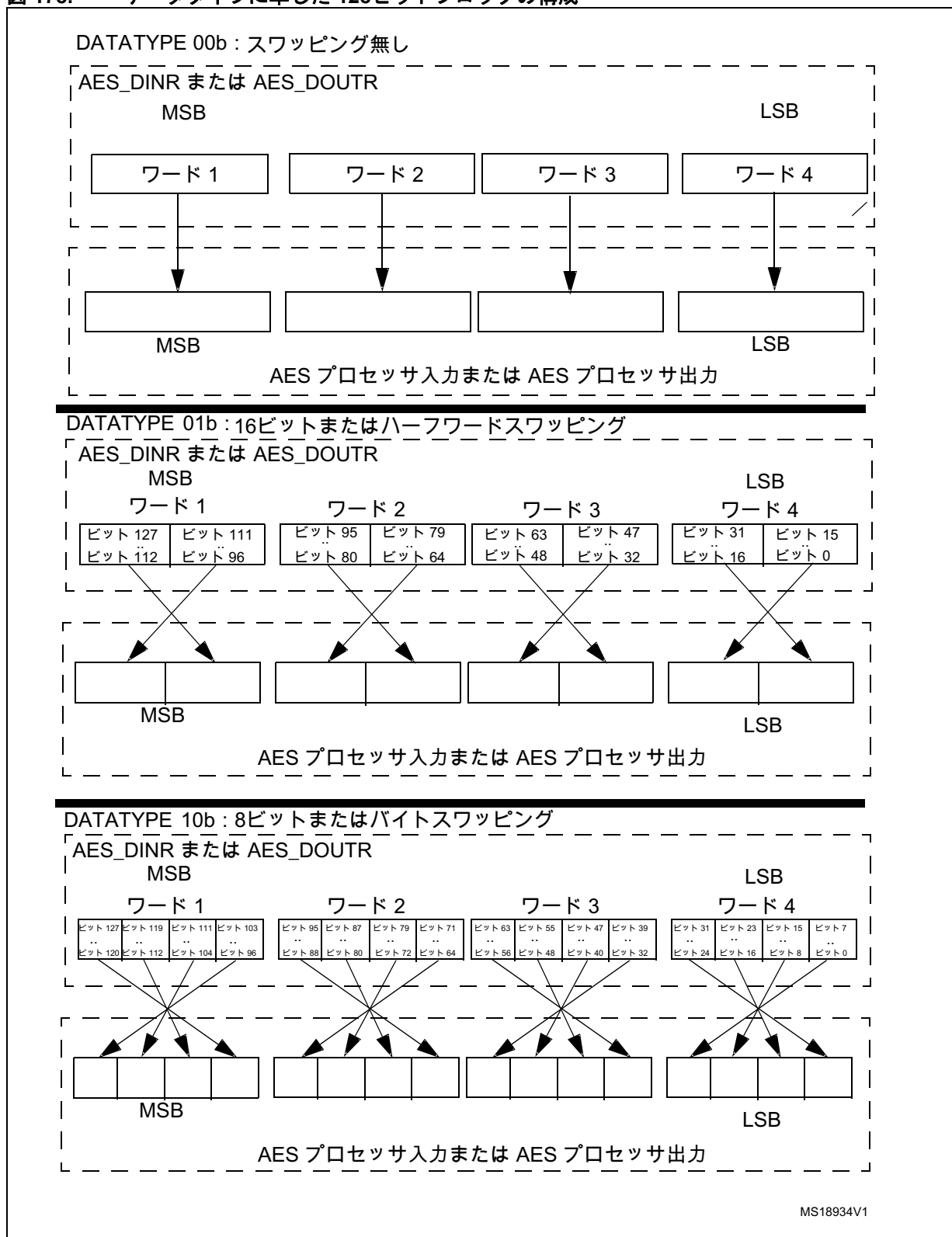
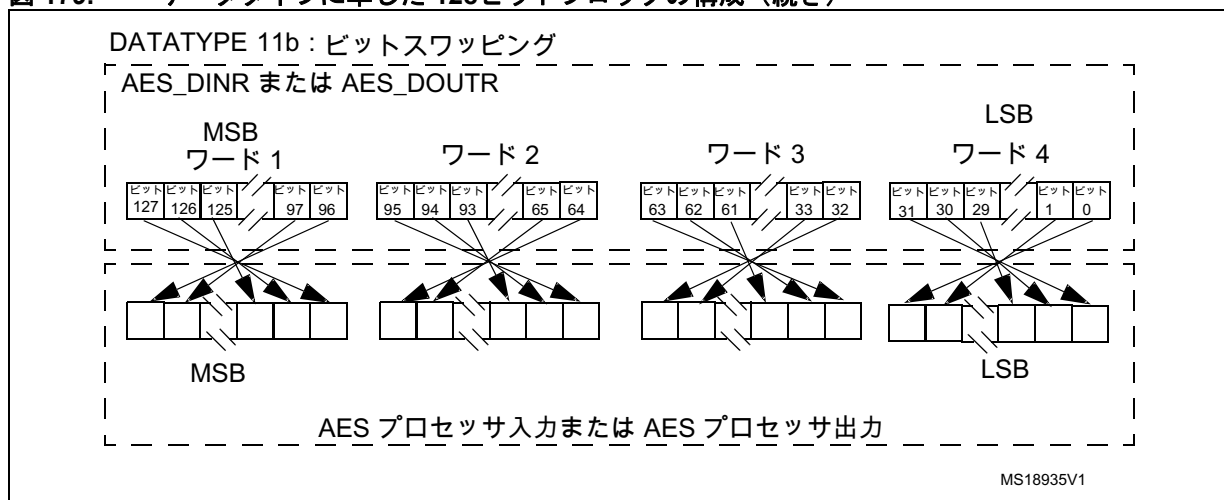


図 179. データタイプに準じた 128ビットブロックの構成 (続き)

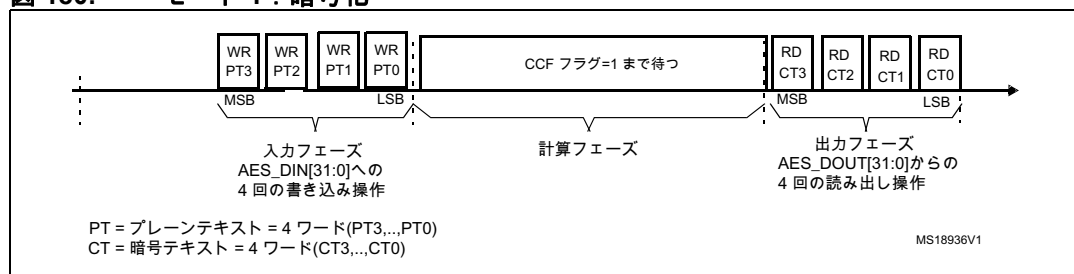


23.7 動作モード

23.7.1 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 をプログラムしてモード 1 を設定し、CHMOD[1:0] ビットをプログラムしてどのタイプの連結モードを実施しなければならないかを選択します。
3. AES_KEYRx レジスタ (128ビットの暗号化キー) および CTR または CBC モードが選択されている場合には AES_IVRx レジスタに書き込みます。EBC モードでは、AES_IVRx レジスタは使用しません。
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_DINR レジスタに 4 回書き込み、[図 180 : モード 1 : 暗号化 \(592 ページ\)](#) に示すようにプレーンテキスト (最上位バイトから) を入力します。
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. AES_DOUTR レジスタを 4 回読み出し、[図 180 : モード 1 : 暗号化 \(592 ページ\)](#) に示すように暗号テキスト (最上位バイトから) を取得します。
8. ステップ 5、6、7 を繰り返し、すべてのブロックを同じ暗号化キーで処理します。

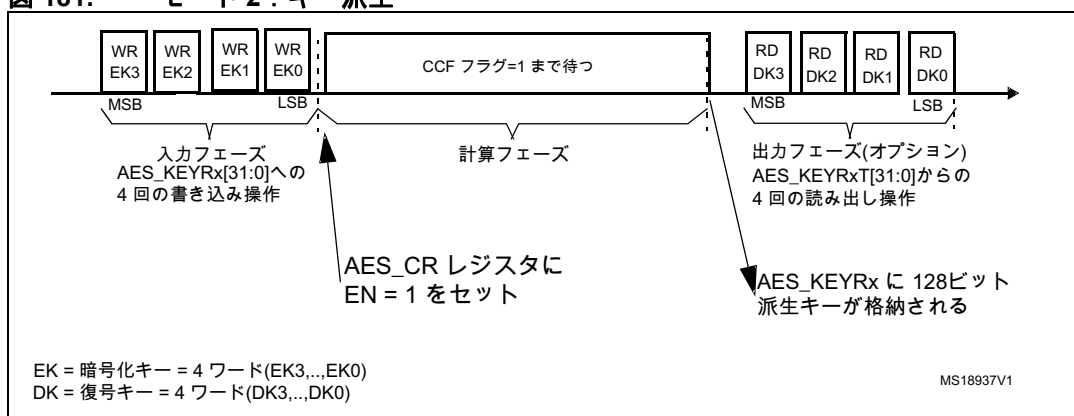
図 180. モード 1 : 暗号化



23.7.2 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ている連結アルゴリズムとは独立しているため、CHMOD[1:0] ビットはこの場合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み、派生キーを取得します。AES_IVRx への書き込みは影響しません。
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. 派生キーが自動的に AES_KEYRx レジスタに入ります。AES_KEYRx レジスタを読み出し、必要に応じて復号化キーを取得します。AES がハードウェアによって無効になります。派生キーの計算を再度開始するには、ステップ 3、4、5、6 を繰り返します。

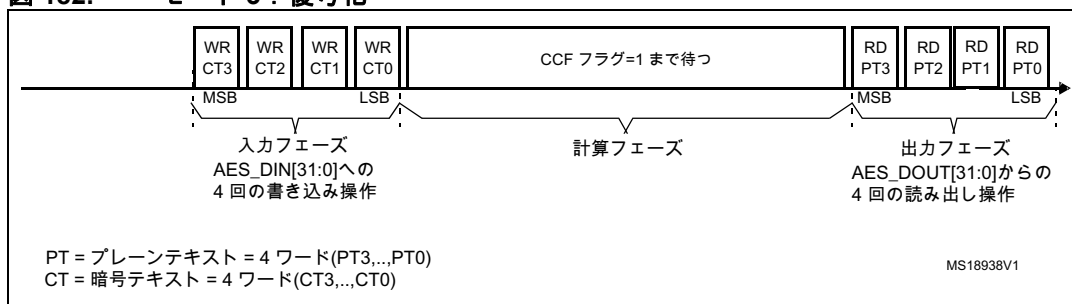
図 181. モード 2 : キー派生



23.7.3 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 をプログラムしてモード 3 を設定し、CHMOD[1:0] ビットをプログラムしてどのタイプの連結モードを実施しなければならないかを選択します。
3. AES_KEYRx レジスタに復号化キーを書き込みます (モード 2: キー派生を使って派生キーがすでに AES_KEYRx レジスタに保存されている場合には、このステップは回避できます)。CTR または CBC モードが選択されている場合は、AES_IVRx レジスタに書き込みます。EBC モードでは、AES_IVRx レジスタは使用しません。
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_DINR レジスタに 4 回書き込み、[図 182 : モード 3 : 復号化 \(594 ページ\)](#) に示すように暗号テキスト (最上位バイトから) を入力します。
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. AES_DOUTR レジスタを 4 回読み出し、[図 182 : モード 3 : 復号化 \(594 ページ\)](#) に示すようにプレーンテキスト (最上位バイトから) を取得します。
8. ステップ 5、6、7 を繰り返し、すべてのブロックを AES_KEYRx レジスタに保存されている同じ派生キーで処理します。

図 182. モード 3 : 復号化

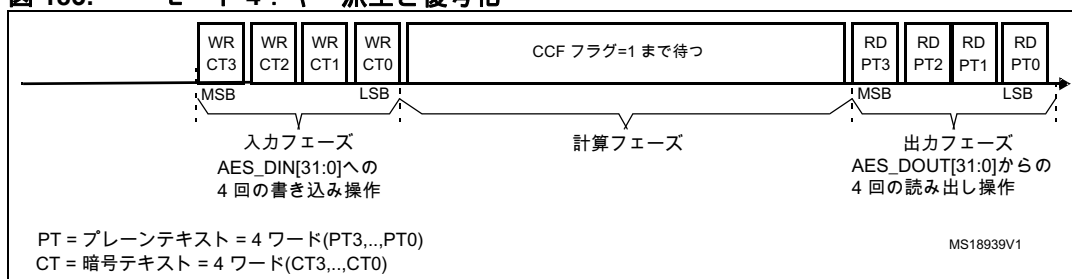


23.7.4 モード 4 : キー派生と復号化

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。このモードは、AES が CTR モードで設定される際には禁止されます。ソフトウェアが MODE[1:0] = 11 および CHMOD[1:0] = 10 を書き込むと、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 モードが選択されている場合は、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. AES_DINR レジスタに 4 回書き込み、[図 183 : モード 4 : キー派生と復号化 \(594 ページ\)](#) に示すように暗号テキスト（最上位バイトから）を入力します。
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. AES_DOUTR レジスタを 4 回読み出し、[図 183 : モード 4 : キー派生と復号化 \(594 ページ\)](#) に示すようにプレーンテキスト（最上位バイトから）を取得します。
8. ステップ 5、6、7 を繰り返し、すべてのブロックを同じ暗号化キーで処理します。

注 : AES_KEYRx レジスタには、処理のあらゆるフェーズにおける暗号化キーが含まれています。これらのレジスタには、派生キーは保存されません。暗号化キーから始まる派生キーは、コピーを AES_KEYRx レジスタに保存することなく内部で AES に保存されます。

図 183. モード 4 : キー派生と復号化



23.8 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ためのインタフェースを提供します。

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、AES_DINR レジスタにワードを書き込む必要が生じるたびに、入力フェーズ中に DMA リクエスト (AES_IN) を開始します。DMA チャネルは、32 ビットのデータサイズでメモリからペリフェラル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 ビットが有効になると、AES は、AES_DOUTR レジスタからワードを読み出す必要が生じるたびに、出力フェーズ中に DMA リクエスト (AES_OUT) を開始します。DMA チャネルは、32 ビットに等しいデータサイズでペリフェラルからメモリモードに設定する必要があります。

各フェーズに対しては 4 つの DMA リクエストがアサートされます。これを [図 184](#) および [図 185](#) で説明します。

DMA リクエストは、AES が無効になるまで生成されます。したがって、1 つの 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れば自動的にそのブロックのための新しいデータ入力フェーズに切り替わります。

注： モード 2 (キー派生) の場合、AES_KEYRx レジスタには CPU を使ってソフトウェアでアクセスできません。この目的に提供されている DMA チャネルはありません。結果として、AES_CR レジスタの DMAINEN ビットおよび DMAOUTEN ビットは、このモードでは一切影響しません。

CCF フラグは DMAOUTEN = 1 のときは関連せず、この場合ソフトウェアは読み出しを行う必要はありません。このビットはハイのままでもよく、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、データ入力フェーズまたはデータ出力フェーズに対して CPU アクセスを使用する場合はソフトウェアによってクリアされる必要があります。

図 184. 入力フェーズ中の DMA リクエストとデータ転送 (AES_IN)

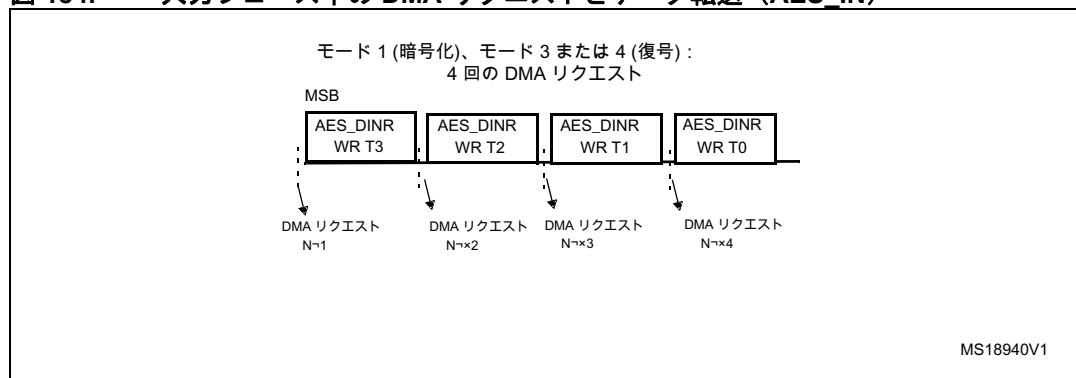
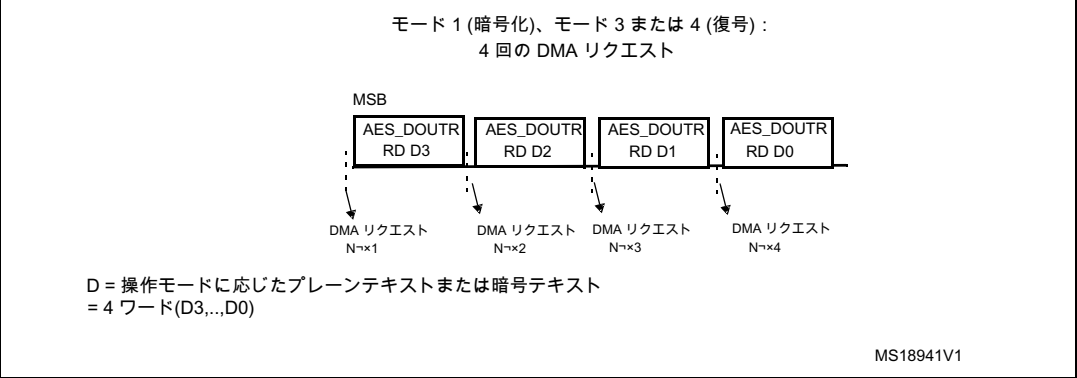


図 185. 出力フェーズ中の DMA リクエスト (AES_OUT)



23.9 エラーフラグ

AES_SR レジスタの RDERR フラグは、計算フェーズ中または入力フェーズ中に予期しない読み出し動作が検出された場合にセットされます。

AES_SR レジスタの WRERR フラグは、出力フェーズ中または計算フェーズ中に予期しない書き込み動作が検出された場合にセットされます。

フラグがクリアされて、AES_CR レジスタの各ビットがセットされることもあります (CCFC ビットで CCF フラグをクリアし、ERRC ビットで WERR および RDERR フラグをクリア)。

AES_CR レジスタの ERRIE ビットがあらかじめセットされていた場合、エラーフラグのいずれか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は無効にされず、通常どおりに処理を続行します。

23.10 処理時間

この表は、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 95. 処理時間 (クロックサイクル単位)

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 : キー派生と復号化	8	276	4	288

23.11 AES 割り込み

表 96. 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	有効制御ビット	WAIT 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

23.12 AES レジスタ

23.12.1 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 として読み出されます。
- ビット 12 **DMAOUTEN**：データ出力フェーズの DMA 管理イネーブル
- 0：DMA（データ出力フェーズ中）は無効です。
 - 1：DMA（データ出力フェーズ中）は有効です。
- DMAOUTEN ビットがセットされる場合、モード 1、3、4 では出力データフェーズに対して DMA リクエストが生成されます。このビットは、モード 2（キー派生）では影響はありません。
- ビット 11 **DMAINEN**：データ入力フェーズの DMA 管理イネーブル
- 0：DMA（データ入力フェーズ中）は無効です。
 - 1：DMA（データ入力フェーズ中）は有効です。
- DMAINEN ビットがセットされる場合、モード 1、3、4 ではデータ入力フェーズに対して DMA リクエストが生成されます。このビットは、モード 2（キー派生）では何の動作も実施しません。
- ビット 10 **ERRIE**：エラー割り込みイネーブル
- RDERR と WRERR の両フラグのいずれかがセットされると、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 フラグ割り込みイネーブル
- 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 割り込みは無効です。
 - 1：CCF 割り込みは有効です。
- 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 連結モード
- 00：EBC（電子コードブック）
 - 01：CBD（Cipher-Block Chaining）
 - 10：CTR（カウンタモード）
 - 11：予約済み
- AES 連結モードは、AES が無効なときのみ変更できます。AES が有効なときにこれらのビットに書き込むことは、予測できない AES 動作を防ぐために禁止されています。

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 : キー派生と復号化

動作モードは、AES が無効なときのみ変更できます。AES が有効なときにこれらのビットに書き込むことは、予測できない AES 動作を防ぐために禁止されています。

CTR モードが選択されている場合、モード 4 は禁止されています。それでもソフトウェアがこの CTR モード設定に対してモード 4 をセットしようとすると、強制的にモード 3 になります。

ビット 2:1 **DATATYPE[1:0]** : データタイプの選択 (暗号ブロックに入力されるデータおよび暗号ブロックから出力されるデータ)

- 00 : 32 ビットデータです。交換なし。
 - 01 : 16 ビットデータまたはハーフワードです。ワードでは、各ハーフワードが交換されます。たとえば、AES_DINR レジスタに書き込まれる 4 個の 32 ビットデータの 1 つが 0x764356AB である場合、暗号ブロックに与えられる値は 0x56AB7643 となります。
 - 10 : 8 ビットデータまたはバイトです。ワードでは、すべてのバイトが交換されます。たとえば、AES_DINR レジスタに書き込まれる 4 個の 32 ビットデータの 1 つが 0x764356AB である場合、暗号ブロックに与えられる値は 0xAB564376 となります。
 - 11 : ビットデータです。ワードでは、すべてのビットが交換されます。たとえば、AES_DINR レジスタに書き込まれる 4 個の 32 ビットデータの 1 つが 0x764356AB である場合、暗号ブロックに与えられる値は 0xD56AC26E となります。
- データタイプの選択は、AES が無効なときのみ変更できます。AES が有効なときにこれらのビットに書き込むことは、予測できない AES 動作を防ぐために禁止されています。

ビット 0 **EN** : AES イネーブル

- 0 : AES は無効です。
- 1 : AES は有効です。

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。AES は、次に EN がセットされると、新しいブロックの処理を開始できます。

このビットは、AES 計算がモード 2 (キー派生) で終了するとハードウェアによってクリアされます。

23.12.2 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 として読み出されます。

ビット 2 WRERR : 書き込みエラーフラグ

このビットは、AES_DINR レジスタへの予期しない書き込み動作が検出されたとき（計算またはデータ出力フェーズ中）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 ビットがあらかじめセットされていた場合、割り込みが生成されます。WERR がセットされていても、このフラグは AES には一切影響しません。AES はそのまま実行を続けます。

AES_CR レジスタの ERRC 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。

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、AES_DOUTR レジスタからの予期しない読み出し動作が検出されたとき（計算またはデータ入力フェーズ中）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 ビットがあらかじめセットされていた場合、割り込みが生成されます。RDERR がセットされていても、このフラグは AES には一切影響しません。AES はそのまま実行を続けます。

AES_CR レジスタの ERRC 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。

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 ビットがあらかじめセットされていた場合、割り込みが生成されます。

AES_CR レジスタの CCFC 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。

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、DAMOUTEN = 0 のときにのみ意味を持ちます。DMA_EN = 1 の場合はハイ状態を維持します。



23.12.3 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の間に 4 回書き込まれなければなりません。

- モード 1 (暗号化) では、プレーンテキストを表す 4ワード (MSB から LSB) が書き込まれなければなりません。
- モード 2 (キー派生) は、AES_KEYRx レジスタから開始する派生キー計算のみに関連するため、このレジスタはこのモードでは使用しません。
- モード 3 (復号化) およびモード 4 (キー派生と復号化) では、暗号テキストを表す 4ワード (MSB から LSB) が書き込まれなければなりません。

注 : このレジスタは、32ビットのデータ幅でアクセスする必要があります。

23.12.4 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) がセットされると、このデータレジスタからの読み出し 4 回で 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、暗号テキストを表す 4ワード (MSB から LSB) を読み出さなければなりません。
- モード 2 (キー派生) では、AES_KEYRx レジスタに派生キーがあります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 (キー派生と復号化) では、プレーンテキストを表す 4ワード (MSB から LSB) を読み出さなければなりません。

注 : このレジスタは、32ビットのデータ幅でアクセスする必要があります。



23.12.5 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)、モード 4 (キー派生と復号化) では、書き込まれる値は最下位ビットからの暗号化キー、つまりキー [31:0] を表します。
モード 3 (復号化) では、書き込まれる値は最下位ビットからの復号化キー、つまりキー [31:0] を表します。復号化モードでレジスタが暗号化キーを使って書き込まれるとき、AES が有効になる前にそれを読み出すと暗号化の値が返されます。CCF フラグがセットされてから読み出すと、派生キーが返されます。
AES が有効な間にこのレジスタを読み出すと、予期できない値が返されます。

注 : このレジスタには、モード 4 (派生キーと復号化) での派生キーは含まれていません。このレジスタには、常に暗号化キーの値が含まれています。

23.12.6 AES キーレジスタ 1 (AES_KEYR0) (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。



23.12.7 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。

23.12.8 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。

23.12.9 AES 初期化ベクタレジスタ 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]** : 初期化ベクタレジスタ（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。
- このレジスタ値は、次のような場合には意味を持ちません。
 - EBC（電子コードブック）モードが選択されている場合。
 - キー派生の他に CTR または CBC モードが選択されている場合。
- CTR（カウンタ）モードでは、このレジスタには 32 ビットのカウンタ値が含まれています。
- AES が有効な間にこのレジスタを読み出すと、値 0x00000000 が返されます。

23.12.10 AES 初期化ベクタレジスタ 1（AES_IVR0）（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]** : 初期化ベクタレジスタ（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。
- このレジスタ値は、次のような場合には意味を持ちません。
 - EBC（電子コードブック）モードが選択されている場合。
 - キー派生やキー派生と復号化モードの他に CTR または CBC モードが選択されている場合。
- CTR（カウンタ）モードでは、このレジスタにはノンス値が含まれています。
- AES が有効な間にこのレジスタを読み出すと、値 0x00000000 が返されます。



23.12.11 AES 初期化ベクタレジスタ 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]** : 初期化ベクタレジスタ (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。
- このレジスタ値は、次のような場合には意味を持ちません。
 - EBC (電子コードブック) モードが選択されている場合。
 - キー派生やキー派生と復号化モードの他に CTR または CBC モードが選択されている場合。
- CTR (カウンタ) モードでは、このレジスタにはノンス値が含まれています。
- AES が有効な間にこのレジスタを読み出すと、値 0x00000000 が返されます。

23.12.12 AES 初期化ベクタレジスタ 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]** : 初期化ベクタレジスタ (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。
- このレジスタ値は、次のような場合には意味を持ちません。
 - EBC (電子コードブック) モードが選択されている場合。
 - キー派生やキー派生と復号化モードの他に CTR または CBC モードが選択されている場合。
- CTR (カウンタ) モードでは、このレジスタにはノンス値が含まれています。
- AES が有効な間にこのレジスタを読み出すと、値 0x00000000 が返されます。



23.12.13 AES レジスタマップ

表 97. AES レジスタマップ

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	AES_CR	予約済み																				DMAOUTEN	DMAINEN	ERRIE	CCFIE	ERRC	CCFC	CHMOD[1:0]		MODE[1:0]		DATATYPE[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		
0x0004	AES_SR	予約済み																												WRERR	RDERR	CCF			
	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		
0x0008	AES_DINR	AES_DINR[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		
0x000C	AES_DOUTR	AES_DOUTR[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		
0x0010	AES_KEYR0	AES_KEYR0[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		
0x0014	AES_KEYR1	AES_KEYR1[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		
0x0018	AES_KEYR2	AES_KEYR2[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		
0x001C	AES_KEYR3	AES_KEYR3[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		
0x0020	AES_IVR0	AES_IVR0[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		
0x0024	AES_IVR1	AES_IVR1[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		
0x0028	AES_IVR2	AES_IVR2[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		
0x002C	AES_IVR3	AES_IVR3[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、表 5 (47 ページ) を参照してください。



24 ユニバーサルシリアルバス（USB）フルスピードデバイスインタフェース

24.1 USB の概要

USB peripheral は、フルスピード USB 2.0 バスと APB1 バスとのインタフェースを実現します。

USB のサスペンド／レジューム操作がサポートされており、消費電力削減のためにデバイスのクロックを停止することができます。

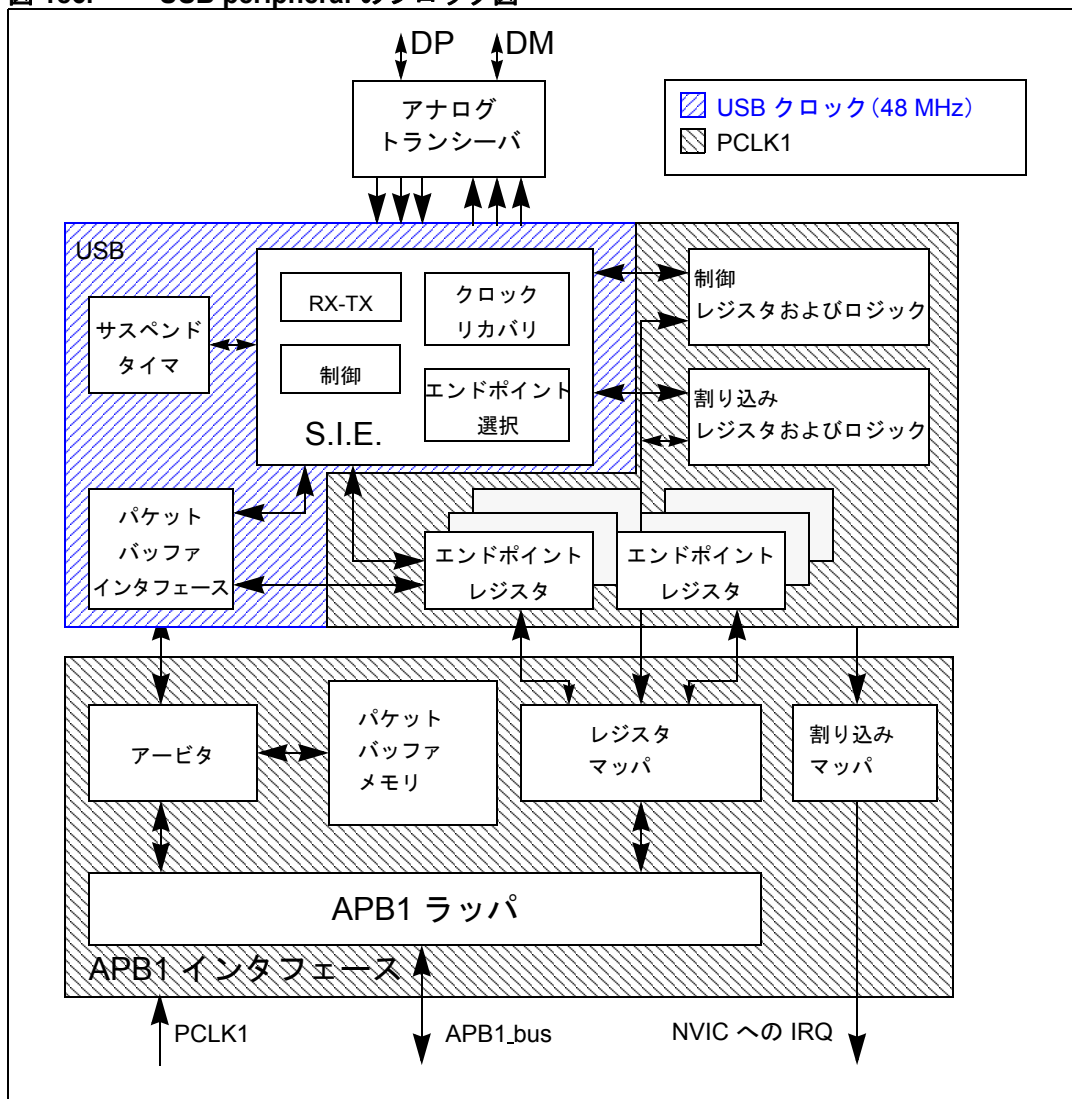
24.2 USB の主な機能

- USB 仕様バージョン 2.0 フルスピードに準拠
- 1 から 8 個の間でエンドポイント数を設定可能
- 巡回冗長検査（CRC）の生成／検査、非ゼロ復帰反転方式（NRZI）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のサポート
- ダブルバッファ方式のバルク/アイソクロナスエンドポイントのサポート
- USB サスペンド／レジューム操作
- フレームに同期したクロックパルスの生成
- USB データ+（DP）ライン上の内部プルアップ抵抗（1.5 k）との USB 内部接続／切断機能（システムコンフィギュレーションレジスタが制御）

24.3 USB の機能詳細

図 186 に、USB peripheral のブロック図を示します。

図 186. USB peripheral のブロック図



USB peripheral は、マイクロコントローラによって実装される機能とホスト PC との間に USB 準拠の接続を提供します。ホスト PC とシステムメモリとの間のデータ転送は、USB peripheral によって直接アクセスされる専用のパケットバッファメモリを介して行われます。この専用バッファメモリのサイズは、使用されるエンドポイントの数と最大パケットサイズに基づいて決める必要があります。この専用メモリのサイズは 512 bytes となっており、最大 16 個の単方向または 8 個の双方向エンドポイントが使用できます。USB peripheral は USB ホストとインタフェース接続し、USB 規格の要求に従ってトークンパケットを検出してデータ送受信を処理し、ハンドシェイクパケットを処理します。CRC の生成と検査など、トランザクションの書式設定はハードウェアによって行われます。

各エンドポイントは、エンドポイントに関連するメモリ領域の位置、そのサイズ、および転送すべきバイト数を示す、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に対するトークンが USB peripheral によって認識されると、関連するデータ転送が行われます（必要であり、かつエンドポイントが設定されている場合）。USB peripheral によってバッファされたデータが内部 16 ビットレジスタにロードされ、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基づいて、USB を介した適切なハンドシェイクパケットが生成または期待されます。

トランザクションの最後には、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の読み出しやさまざまな割り込み応答ルーチンが使用されます。マイクロコントローラは、以下の項目を判定できます。

- 処理すべきエンドポイント
- エラーが発生した場合（ビットスタッフィング、フォーマット、CRC、プロトコル、ACK の不在、オーバーラン／アンダーランなど）に行われたトランザクションのタイプ

アイソクロナス転送と高スループットのバルク転送に特別なサポートが提供され、ダブルバッファ方式を実装しています。このため、マイクロコントローラが一方のバッファを使用している間、USB peripheral は他方のバッファを使用できます。

このユニットは、必要に応じて制御レジスタに書き込むことによって、低電力（サスペンド）モードにできます。このとき、すべての静的な電力消費が抑えられ、USB クロックを低速にしたり停止したりすることができます。低電力モード時に USB 入力の動作が検出されると、デバイスは非同期でウェイクアップされます。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と、システムは通常のクロック生成をただちにリスタートしたり、クロックの直接的な開始/停止をサポートしたりできます。

24.3.1 USB ブロックの説明

USB peripheral は、USB インタフェースに関するすべての機能を実装し、以下のブロックを含みます。

- 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は、同期パターンの認識、ビットスタッフィング、CRC の生成と検査、PID の確認／生成、およびハンドシェイクの評価です。このブロックは、USB トランシーバとインタフェースする必要があり、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して、ローカルデータストレージを実現します。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 peripheral のイベントや、送信終了や正常なパケット受信などのエンドポイント関連のイベントに基づいて、信号も生成します。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、フレーム開始に同期したクロックパルスを生成し、トラフィックが 3 ms の間受信されないときは、ホストからのグローバルサスペンドを検出します。
- 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、送受信に使用される一連のバッファを柔軟に実装します。このブロックは、SIE からのリクエストに基づいて適切なバッファを選択して、エンドポイントレジスタが示すメモリアドレスに配置します。パケットの終了まで、1 ワード交換されるたびにアドレスをインクリメントし、交換されたバイト数を把握して、バッファが最大容量を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：個々の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むレジスタが関連付けられます。一方向／シングルバッファのエンドポイントの場合、1 個のレジスタを使用して、2 つの異なるエンドポイントを実装できます。レジスタは全部で 8 個あり、最大 16 個の一方向／シングルバッファエンドポイントか、最大 7 個のダブルバッファエンドポイント* を自由に組み合わせることができます。たとえば、USB peripheral は、4

個のダブルバッファエンドポイントと 8 個のシングルバッファ／一方向エンドポイントを持つようにプログラミングすることができます。

- 制御レジスタ：これらは、USB peripheral 全体のステータス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 イベントの強制実行に使用されます。
- 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。割り込み理由や割り込みステータスの問い合わせに使用したり、ペンディング割り込みのステータスのクリアに使用できます。

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制御転送に使用されます。

USB peripheral は APB1 インタフェースを介して APB1 バスに接続され、APB1 インタフェースには次のブロックが含まれます。

- パケットメモリ：パケットバッファを物理的に含むローカルメモリです。データ構造を作成し、アプリケーションソフトウェアによって直接アクセスできるパケットバッファインタフェースによって使用できます。パケットメモリのサイズは 512 bytes であり、16 ビット単位の 256 ワードとして構成されます。
- アービタ：このブロックは APB1 バスと USB インタフェースからのメモリリクエストを受け入れます。APB1 アクセスに優先権を与えることによって競合を解決すると同時に、メモリ帯域幅の 1/2 を常に確保することによってすべての USB 転送を完了させます。この時間二重化方式は、仮想デュアルポート SRAM を実装して、USB トランザクション中のメモリアクセスを可能にします。この方式では、任意の長さのマルチワード APB1 転送も可能です。
- レジスタマップ：このブロックは、USB peripheral のさまざまなバイト幅およびビット幅のレジスタを APB1 によってアドレス指定される 16 ビット幅の構造化されたワードセットにまとめます。
- APB1 ラッパ：メモリとレジスタに APB1 へのインタフェースを提供します。また、USB peripheral 全体を APB1 アドレス空間に割り付けます。
- 割り込みマップ：このブロックは、USB イベントによる割り込みの生成方法と、NVIC の 3 つの異なるラインへの割り付け方法の選択に使用します。
 - － USB 低優先割り込み（チャンネル 20）：すべての USB イベント（正しい転送や USB リセットなど）によってトリガされます。ファームウェアは、割り込みを処理する前に、割り込みソースをチェックしなければなりません。
 - － USB 高優先割り込み（チャンネル 19）：アイソクロナスおよびダブルバッファバルク転送が最高の転送速度に到達できるように正しい転送イベントによってのみトリガされます。
 - － USB ウェイクアップ割り込み（チャンネル 42）：USB サスペンドモードからのウェイクアップイベントによってトリガされます。

24.4 プログラミングの際の注意

以下のセクションでは、アプリケーションソフトウェアの開発を容易にするために、USB peripheral とアプリケーションプログラム間の予想される相互作用について説明します。

24.4.1 一般的な USB デバイスのプログラミング

ここでは、USB 準拠の動作を達成するために、アプリケーションソフトウェアに要求される主なタスクについて説明します。最も一般的な USB イベントに関連するアクションが考慮されており、ダブルバッファエンドポイントやアイソクロナス転送といった特殊なケースを取り上げている節もあります。システムリセットは別として、アクションは常に下記のいずれかの USB イベントによって駆動される USB peripheral によって開始されます。

24.4.2 システムリセットとパワーオンリセット

システムリセットやパワーオンリセット時に、アプリケーションソフトウェアが行うべき最初のアクションは、必要なすべてのクロック信号を USB peripheral に供給し、続いてその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することです。初期化シーケンス全体は、以下のとおりです。

内部プルアップ抵抗は Data+ (DP) ラインに接続され、SYSCFG モジュールの SYSCFG_PMC レジスタの USB_PU ビットを使用してソフトウェアで制御されています ([セクション 8 : システム設定コントローラ \(SYSCFG\) とルーティングインタフェース \(RI\)](#) 参照)。USB_PU ビットがリセットされ、DP ラインに接続されるプルアップ抵抗がなくなると、デバイスは USB バス上で検出できなくなります (外部プルアップが接続されていない場合)。USB_PU ビットがセットされると、内部プルアップ抵抗が接続され、デバイスが USB バス上で検出できるようになります。

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、デバイスの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、レジスタの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します。

次に、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、特別な操作が必要です。このビットは、ポートトランシーバに電源を供給する内部電圧基準をオンにします。この回路には定義済みの起動時間 (データシートでは $t_{STARTUP}$ で指定) があり、その間の 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は、この時間の経過を待ってから、(CNTR レジスタの FRES ビットをクリアして) USB 部のリセット条件を解除する必要があります。ISTR レジスタをクリアすると、スプリアスなペンディング割り込みが除去され、他のマクロセル操作が有効になります。

システムリセット時には、USB peripheral が割り込み生成とデータ転送を正しく行えるように、マイクログコントローラが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なければなりません。どのエンドポイントにも限定され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ニーズ (有効な割り込みの選択やパケットバッファの選択したアドレスなど) に従って初期化する必要があります。次に、このプロセスは USB リセット (次項) に引き継がれます。

USB リセット (RESET 割り込み)

このイベントが発生すると、USB peripheral は、前項で説明した初期化後にシステムリセットが行われた場合と同じ条件に置かれます。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 peripheral はパケットに 응답しません)。USB リセットイベントへの応答として、USB 機能を有効にする必要があります。すなわち、USB アドレスを 0 にし、デフォルトのコントロール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 です)。それには、USB_DADDR レジスタのイネーブルファンクション (EF) ビットをセットし、EP0R レジスタとそれに関連するパケットバッファを初期化します。USB エニュメレーションプロセスで、ホストは、このデバイスに一意なアドレスを割り当て (このアドレスは USB_DADDR レジスタの ADD[6:0] ビットに書き込む必要があります)、他の必要なエンドポイントを設定します。

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わりから 10 mS 以内に、デフォルトエンドポイントの USB 機能 0 を再び有効にする必要があります。

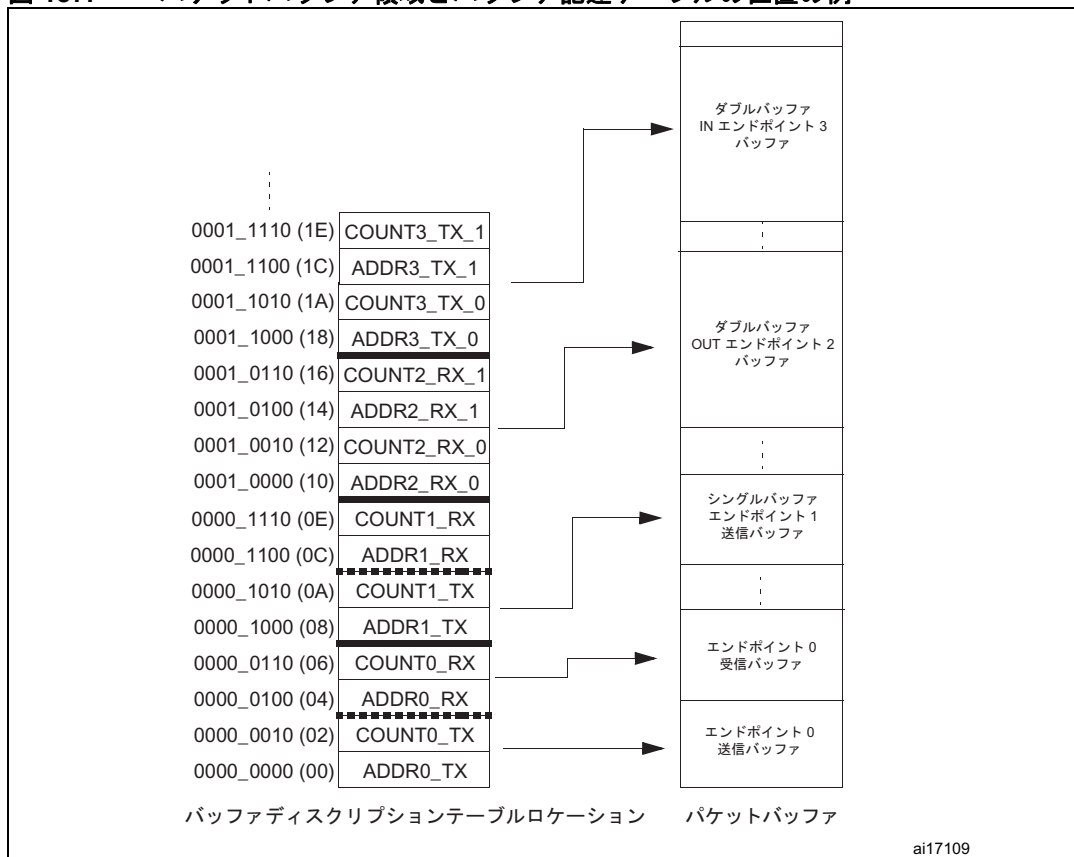
パケットバッファの構造と使用

個々の双方向エンドポイントは、ホストとの間でデータの送受信を行うことができます。受信されたデータはそのエンドポイント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、このエンドポイントから送信されるデータが格納されます。このメモリへはパケットバッファインタフェースブロックがアクセスします。このブロックは、メモリアクセスリクエストを配信し、その確認応答を待ちます。パケットバッファメモリもマイクロコントローラによってアクセスされる必要があるため、アービタロジックが APB1 サイクルの 1/2 をマイクロコントローラのアクセスに使用し、残りの 1/2 を USB peripheral のアクセスに使用してアクセスの競合を管理します。このようにして、マイクロコントローラが連続アクセスを行っている場合でも、双方とも競合を気にすることなく、パケットメモリがデュアルポート SRAM であるかのように動作することができます。USB peripheral ロジックは専用のクロックを使用します。この専用クロックの周波数は、USB 標準の要件によって 48 MHz に固定されており、APB1 バスへのインタフェース用のクロックとは異なる場合があります。APB1 クロック周波数を USB peripheral の周波数より高くも、低くもできる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の要件により、データのオーバーラン/アンダーラン問題を避けるには、APB1 クロック周波数が 10 MHz より大きくなければなりません。**

各エンドポイントは、2 個のパケットバッファ (通常、送信と受信に 1 個ずつ) に関連付けられています。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ため、バッファはパケットメモリ内のどこにでも置くことができます。バッファ記述テーブルも、USB_BTABLE レジスタが示すパケットメモリ内のアドレスにあります。各テーブルエントリは、エンドポイントレジスタに関連付けられ、テーブル開始アドレスが常に 8 バイト境界に配置されるように、4 つの 16 ビットワードで構成されます (USB_BTABLE レジスタの最下位 3 ビットは常に「000」です)。バッファ記述テーブルのエントリについては、[セクション 24.5.3: バッファディスクリプタテーブル](#) を参照してください。エンドポイントが一方であり、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関連するもの)。サポートされていない転送方向や未使用のエンドポイントに関連するその他のテーブル位置は、ユーザーが使用できます。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では、パケットバッファの特別な取り扱いが必要です (それぞれ、[セクション 24.4.4: アイソクロナス転送](#) と [セクション 24.4.3: ダブルバッファエンドポイント](#) を参照してください)。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 187](#) に示します。

図 187. 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、送受信時に、下端から使用されます。USB peripheral が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更することはありません。割り当てられたバッファ長を上回るパケットを受信した場合（バッファオーバーラン条件）、データは使用可能な最後の位置までしかメモリにコピーされません。

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 USB peripheral が既に用意されている送信すべきデータを見つけ出し、受信するデータをバッファに格納できるようにすることです。USB_EPnR レジスタの EP_TYPE ビットは、エンドポイントの種類に基づいてセットする必要があります。最終的には、EP_KIND ビットを使用して、必要な特殊機能を有効にします。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し、COUNTn_TX を初期化する必要があります。受信の場合は、STAT_RX ビットをセットして受信を有効にし、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で COUNTn_RX に書き込む必要があります。アイソクロナスとダブルバッファのバルクエンドポイントを除いて、一方向のエンドポイントでは、サポートされる方向に関連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。送信や受信が有効になると、USB_EPnR レジスタと ADDRn_TX/ADDRn_RX、COUNTn_TX/COUNTn_RX の各位置をアプリケーションソフトウェアで変更することはできません。これらの値は、ハードウェアがその場で変更できます。データ転送が完了すると、CTR 割り込みイベントによって通知され、これらの値は、新しい操作を再び有効にするように、再びアクセスできるようになります。

IN パケット (データ送信)

IN トークンパケットを受信したとき、受信したアドレスが設定済みの有効なエンドポイントアドレスと一致した場合、USB peripheral は、その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 ADDRn_TX と COUNTn_TX の位置にある内容にアクセスします。これらの位置の内容は、ソフトウェアではアクセスできない、内部の 16 ビットレジスタ ADDR と COUNT に格納されます。パケットメモリに再びアクセスして送信すべき最初のワードを読み出し ([パケットバッファの構造と使用 \(612 ページ\)](#) を参照)、USB_EPnR の DTOG_TX ビットに基づいて、DATA0 または DATA1 PID の送信を開始します。PID の送信が完了すると、バッファメモリから読み出されたワードの最初のバイトが、USB バスに送信される出力シフトレジスタにロードされます。最後のデータバイトが送信された後、算出された 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い場合、USB_EPnR レジスタの STAT_TX ビットに基づいて、データパケットの代わりに NAK または STALL ハンドシェイクパケットが送信されます。

内部レジスタ ADDR は現在のバッファメモリ位置へのポインタとして使用され、COUNT は送信される残りのバイト数のカウントに使用されます。パケットバッファメモリから読み出された各ワードは、USB バスを介して最下位バイトから送信されます。送信バッファメモリは、ADDRn_TX で示されるアドレスから COUNTn_TX/2 ワードが読み出されます。送信パケットが奇数バイトで構成される場合、アクセスされた最後のワードの下半分のみが使用されます。

ホストからの ACK 応答を受信すると、USB_EPnR レジスタでは、まず DTOG_TX ビットが反転され、STAT_TX=10 (NAK) をセットしてエンドポイントが無効になり、CTR_TX ビットがセットされます。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 ビットを調べて、マイクロコントローラのサービスを要求しているエンドポイントを特定する必要があります。CTR_TX イベントへの対応は、割り込みビットのクリアから始まります。アプリケーションソフトウェアは、送信すべきデータが格納された別のバッファを用意し、次の転送で送信すべきバイト数で COUNTn_TX テーブル位置を更新し、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) である間、このエンドポイントに対する IN リクエストは NAK となり、フロー制御条件が示されます。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に続いて、同じエンドポイントに対する 2 番目の IN トランザクションの通知を失わないためには、上記の順序で一連の操作を実行することが必要です。

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 peripheral によってほぼ同じように扱われます。SETUP パケットの扱いの相違点は、次の制御転送に関する項で詳しく説明します。OUT/SETUP PID を受信したとき、アドレスが有効なエンドポイントに一致した場合、USB peripheral は、その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 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、そのまま内部レジスタ ADDR に格納されます。COUNT はリセットされ、COUNTn_RX に読み込まれた BL_SIZE および NUM_BLOCK ビットフィールドの値は、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 BUF_COUNT の初期化に使用されます (これらの内部レジスタにはソフトウェアからアクセスできません)。その後 USB peripheral によって受信されたデータバイトは、最初に受信したバイトが最下位バイトとなるようにワード単位でパックされ、内部レジスタ ADDR に格納されたアドレスから始まるパケットバッファに転送されます。また、バイト転送のたびに 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 パケットの終了が検出されると、受信した CRC がチェックされ、受信中にエラーが発生していない場合にのみ、ACK ハンドシェイクパケットが送信側のホストに送り返されます。

CRC やその他の種類のエラー (ビットスタッフ違反、フレームエラーなど) が検出された場合、データバイトは、少なくともエラーが検出された位置まではパケットメモリバッファ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による対処は必要ありません。USB peripheral は受信エラーから回復し、次のトランザクションを受信することができます。指定されたエンドポイントが有効でない場合は、

USB_EPnR レジスタの STAT_RX ビットに基づいて、ACK の代わりに NAK または STALL ハンドシェイクパケットが送信され、受信メモリバッファにはデータが書き込まれません。

受信メモリバッファ位置には、ADDRn_RX レジスタに格納されたアドレスから始まって、CRC を含む受信データパケット長に対応するバイト数（すなわち、データペイロード長 + 2）か、または BL_SIZE と NUM_BLOCK のどちらか先に着信した方によって定義される最後に割り当てられたメモリ位置まで、データが書き込まれます。このように、USB peripheral は割り当てられた受信メモリバッファ領域を越えて書き込みを行うことはありません。データパケットのペイロードのサイズ（アプリケーションが使用した実際のバイト数）が割り当てられたバッファのサイズより大きな場合、USB peripheral はバッファオーバーランを検出し、通常の ACK の代わりに STALL ハンドシェイクを戻すことでホストに問題が生じたことを通知します。このときには割り込みは発生せず、トランザクションは失敗したものとみなされます。

トランザクションが正しく完了すると、ACK ハンドシェイクパケットが送信され、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 位置に書き戻され、BL_SIZE と NUM_BLOCK フィールドには影響を与えません（これらのフィールドは、通常、書き換え不要です）。USB_EPnR レジスタでは、まず DTOG_RX ビットがトグルされ、STAT_RX = 10 (NAK) をセットしてエンドポイントを無効にし、CTR_RX ビットがセットされます。エラーまたはバッファオーバーラン条件によってトランザクションが失敗した場合、上述の動作は一切行われません。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 ビットを調べて、マイクロコントローラのサービスを要求しているエンドポイントを特定する必要があります。CTR_RX イベントへの対応としては、まず、USB_EPnR レジスタの SETUP ビットでトランザクションのタイプを判定します。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、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 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、アプリケーションソフトウェアは、USB_EPnR レジスタの STAT_RX ビットを 11 (VALID) にセットして、以降のトランザクションを有効にします。STAT_RX ビットが 10 (NAK) である間、このエンドポイントに対する OUT リクエストは NAK となり、フロー制御条件が示されます。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に続いて、同じエンドポイントに対する 2 番目の OUT トランザクションの通知を失わないためには、上記の順序で一連の操作を実行することが必要です。

制御転送

制御転送は、SETUP トランザクション、0 個以上のデータステージ（すべて同一方向）、およびステータスステージ（逆方向の 0 バイト転送）で構成されます。SETUP トランザクションは、コントロールエンドポイントによってのみ処理され、OUT トランザクション（データ受信）に非常に似ています。ただし、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 ビットの値はそれぞれ 1 と 0 にセットされます。さらに、SETUP の内容に応じて後続のトランザクションが IN であるか OUT であるかをソフトウェアが決定できるように、STAT_TX と STAT_RX はともに 10 (NAK) にセットされます。コントロールエンドポイントは、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、SETUP トランザクションから通常の OUT トランザクションを見分ける必要があります。USB デバイスは、SETUP ステージで転送されたデータを解釈してデータステージの数と方向を判断することができ、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、最後を除くすべてのデータステージで、未使用の方向を STALL に設定します。これによって、ホストによる転送方向の切り替えが早すぎた場合は、ステータスステージとして STALL が戻されることになります。

最後のデータステージを有効にしている間、反対方向は NAK に設定します。これによって、ホストがステータスステージを実行するためにただちに転送方向を切り替えた場合は、制御操作の完了を待つことになります。制御操作が正しく完了した場合、ソフトウェアは NAK を VALID に変更します。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 になる場合は、STATUS_OUT (USB_EPnR レジスタの EP_KIND) ビットをセットします。これによって、ステータストランザクションが 0 以外のデータで実行された場合にエラーが生成されます。ステータストランザクションが処理されるとき、アプリケーションは、STATUS_OUT ビットをクリアし、STAT_RX

を VALID にセットして新しいコマンドを受け付け、STAT_TX を NAK にセットして次の SETUP に続くステータスステージを遅らせます。

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ないと定められているため、最終的には以前に発行したコマンドをアボートして新しいコマンドを開始することになります。USB ロジックでは、コントロール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 で応答できません。

STAT_RX ビットが 01 (STALL) または 10 (NAK) にセットされ、SETUP トークンを受信すると、USB はデータを受け入れ、必要なデータ転送を実行し、ACK ハンドシェイクを返します。そのエンドポイントに、まだアプリケーションの確認応答を受けていない発行済みの CTR_RX リクエストがある（すなわち、CTR_RX ビットが以前の受信完了によってまだセットされている）場合、USB は SETUP トランザクションを破棄し、どのようなステータスであっても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、ホストに SETUP トークンの再送信を強制します。これは、CTR_RX 割り込みをトリガしたトランザクションに続いて、同じエンドポイントに対する SETUP トランザクションの通知が失われるのを避けるためです。

24.4.3 ダブルバッファエンドポイント

USB 標準で定義されるすべてのエンドポイントタイプは、それぞれ異なるトラフィックモデルを表し、さまざまな種類のデータ転送操作の典型的な条件を記述します。データの大部分がホスト PC と USB 機能の間で転送される場合は、バルクエンドポイントタイプが最適なモデルです。これは、ホストがフレーム内の使用可能な帯域幅すべてが満たされるようにバルクトランザクションをスケジュールするため、USB 機能が指定されたバルクトランザクションに対応できる限り、実際の転送速度を最大化できるからです。次のトランザクションの到着時に、USB 機能が前のトランザクションの処理でまだビジー状態であった場合は、NAK ハンドシェイクで応答します。このとき、ホスト PC は、USB 機能が対応できるようになるまで同じトランザクションを再発行するため、再送信によって帯域幅が占有され、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”と呼ばれる専用の機能を使用できます。

“ダブルバッファリング”が有効になると、データトグルシーケンスを使用して、必要なデータ転送を行うために USB peripheral が使用するバッファを選択します。この際、“送信”および“受信”パケットメモリ領域を使用してトランザクションが正常に終了するたびにバッファ交換を管理すると、アプリケーションが使用するバッファは常に完全となり、USB peripheral は他方のバッファを満たします。たとえば、“受信”ダブルバッファバルクエンドポイントを対象とする OUT トランザクションでは、一方のバッファが USB ホストから送信される新しいデータで満たされている間、もう一方のバッファをマイクロコントローラソフトウェアで使用できます（“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）。

バッファ交換の管理では割り当てられたメモリバッファの長さとおアドレスポイントを格納する 4 つ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現に使用される USB_EPnR レジスタは、一方向エンドポイントとしての使用を強制されます。したがって、ダブルバッファバルクエンドポイントが受信用に有効にされている場合は STAT_RX、ダブルバッファバルクエンドポイントが送信用に有効にされている場合は STAT_TX というように、ただ 1 つの STAT ビットペアのみを 00 (無効) とは異なる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を送信と受信の両方に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。

ダブルバッファ機能を活用して最高の転送速度を達成するには、これまでの章で説明したエンドポイントのフロー制御構造を変更して、トランザクションが正常終了するたびにではなく、USB peripheral とアプリケーションソフトウェアの間でバッファ競合が発生した場合にのみ、エンドポイントステータスを NAK に切り替えます。USB peripheral 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 ビットによって指定されます。これは、“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)、“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。この新しいフロー制御方式を実現するには、どのパケットバッファが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のかを USB peripheral が知り、競合を認識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、USB peripheral がデータとバッファの順序付けに使用するのはそのうちの 1 つだけであり (ダブルバッファ方式による一方向という制約のため)、もう 1 つは現在どちらのバッファを使用中であるかをアプリケーションソフトウェアが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。次の表は、“送信”と“受信”の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 の定義の対応を示します。

表 98. ダブルバッファリングのバッファフラグ定義

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 peripheral 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識別されます。バッファフラグ値と使用されるパケットバッファの関係は、いずれの場合も同じで、次の表に示されています。

表 99. バルクダブルバッファリングメモリのバッファ使用

エンド ポイントの 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機能は、次のように有効にします。

- USB_EPnR レジスタの EP_TYPE ビットフィールドに 00 を書き込んで、エンドポイントをバルクとして定義し、
- 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、使用する最初のバッファに基づいて DTOG ビットと SW_BUF ビットを初期化します。その際には、これらのビットの持つ特別なトグルのみの特性に注意する必要があります。DBL_BUF をセットした後で最初のトランザクションが終了すると、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ます。この制御は、DBL_BUF がセットされている限り、このエンドポイントに指定された他のすべてのトランザクションに使用されます。各トランザクションの最後には、有効にされた方向に応じて、指定されたエンドポイントの 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、該当する USB_EPnR レジスタの DTOG ビットがハードウェアによってトグルされるため、USB peripheral のバッファ交換はソフトウェアとは完全に独立して行われます。一般的なトランザクションや、DBL_BUF をセットした後の最初のトランザクションとは異なり、STAT ビットペアはトランザクション終了による影響を受けず、その値は 11 (Valid) 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を受信すると、USB peripheral とアプリケーションソフトウェアの間のバッファ競合が検出されたときに（この条件は、表 99 (618 ページ) に示すように、DTOG と SW_BUF が同じ値を持つことによって識別されます）、実際のエンドポイントステータスは 10 (NAK) としてマスクされます。アプリケーションソフトウェアは、CTR イベント通知に対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に必要な処理を開始して応答します。アプリケーションのパケットバッファ使用が終わると、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であることを USB peripheral に通知します。このように、NAK となるトランザクションの数は、アプリケーションによるトランザクションデータのエラーレーション時間によってのみ制限されます。このエラーレーション時間が USB バス上でトランザクションの完了に必要な時間より短い場合、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、関連する USB_EPnR レジスタの STAT ビットペアに 11 (Valid) 以外の明示的なステータスを書き込み、ダブルバッファバブルエンドポイント用に実装された特殊なフロー制御をいつでも変更できます。この場合、USB peripheral は、バッファの使用状態とは無関係に、常にプログラミングされたエンドポイントステータスを使用します。

24.4.4 アイソクロナス転送

USB 標準は、一定で正確なデータ生成/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、この種のトラフィックを「アイソクロナス」と定義しています。このデータの典型例としては、オーディオサンプル、圧縮されたビデオストリーム、一般的に供給される周波数の精度に対して厳しい条件をもつあらゆるサンプルデータなどがあります。エニュメレーションフェーズにおいてエンドポイントが「アイソクロナス」と定義されると、ホストはフレームに必要な帯域幅を割り当て、エンドポイントの方向に応じて各フレームにちょうど 1 個の IN または OUT パケットを配信します。帯域幅要件を制限するため、アイソクロナストラフィックで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がなく、データパケットの後に ACK パケットが要求されることも送信されることもありません。同じ理由で、アイソクロナス転送ではデータトグルシーケンスはサポートされず、データパケットの開始には常に DATA0 PID が使用されます。

エンドポイントのアイソクロナス動作を選択するには、USB_EPnR レジスタの EP_TYPE ビットを 10 にセットします。ハンドシェイクフェーズは存在しないため、STAT_RX/STAT_TX ビットペアの適正な値は 00 (無効) と 11 (有効) のみであり、これ以外の値を指定すると、USB 標準に準拠しない結果となります。アイソクロナスエンドポイントでは、アプリケーションソフトウェアの開発を容易にするダブルバッファリングを実装し、「送信」と「受信」のパケットメモリ領域を使用してトランザクションが正常終了するたびにバッファ交換を管理します。これによって、アプリケーションは常に完全なバッファを使用すると同時に、USB peripheral は他方のバッファを使用できます。

USB peripheral によって現在使用されているメモリバッファは、表 100 に基づいてエンドポイント方向に関連する USB_EPnR レジスタの DTOG ビット ("受信"アイソクロナスエンドポイントでは DTOG_RX、"送信"アイソクロナスエンドポイントでは DTOG_TX) で定義されます。

表 100. アイソクロナスメモリバッファ

エンドポイントのタイプ	DTOG ビットの値	以下が使用するパケットバッファ : USB peripheral	以下が使用するパケットバッファ : アプリケーションソフトウェア
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。

ダブルバッファバルクエンドポイントの場合と同じく、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、一方向エンドポイントとしての使用を強制されます。アイソクロナスエンドポイントを送信と受信の両方に対して有効にする必要がある場合、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、使用する最初のバッファに基づいて DTOG ビットを初期化します。その際には、これらのビットの持つ特別なトグルのみの特性に注意する必要があります。各トランザクションの最後には、有効にされた方向に応じて、指定されたエンドポイントの 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、該当する USB_EPnR レジスタの DTOG ビットがハードウェアによってトグルされるため、バッファ交換はソフトウェアとは完全に独立して行われます。アイソクロナス転送にはハンドシェイクフェーズがなく、フロー制御が不可能なため、STAT ビットペアはトランザクション完了の影響を受けず、エンドポイントは常に 11 (有効) のままです。アイソクロナス OUT 転送中に CRC エラーやバッファオーバーラン条件が発生しても正しいトランザクションとみなされ、常に CTR_RX イベントがトリガされます。ただし、CRC エラーでは常に USB_ISTR レジスタの ERR ビットがセットされ、ソフトウェアにはデータ破壊の可能性が通知されます。

24.4.5 サスペンド／レジュームイベント

USB 標準では、サスペンド (SUSPEND) と呼ばれる特殊なペリフェラル状態が定義されます。この状態では、USB バスからの平均電流が 2.5 mA を超えてはいけません。この要件は、バスパワー型のデバイスにとっては非常に重要ですが、セルフパワー型のデバイスはこの厳しい消費電力制約に従う必要はありません。サスペンドモードでは、ホスト PC は USB バスに 3 mS 以上にわたって一切トラフィックを送信しないという通知を送信します。通常の動作では 1 mS ごとに SOF パケットを送信する必要があるため、USB peripheral は SOF パケットが 3 回続けて存在しないときにホスト PC からサスペンドリクエストがあったと認識し、USB_ISTR レジスタの SUSP ビットを 1 にセットして、割り込みを発生させます (有効な場合)。サスペンド状態になったデバイスは、いわゆるレジュームシーケンスによって通常の動作に復帰できます。このシーケンスはホスト PC から開始したり、ペリフェラル自身から直接開始できますが、必ずホスト PC が終了します。サスペンドされた USB peripheral は、リセットシーケンスを検出してこのイベントに通常の USB リセットイベントとして対処できる必要があります。

USB ペリフェラルのサスペンドに使用される実際の手順は、全体の消費電力を低減する操作がデバイス構成によって異なるため、デバイスに依存するものとなります。

次に、代表的なサスペンド手順について簡単に説明します。この説明では、USB peripheral の SUSP 通知に応答するアプリケーションソフトウェアルーチンの USB 関連面に焦点を合わせています。

1. USB_CNTR レジスタの FSUSP ビットを 1 にセットします。これによって USB peripheral 内で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 のサスペンド中に別の SUSP 割り込みが発行されないように、SOF 受信のチェックが無効になります。
2. USB peripheral 以外のブロックの静的消費電力をカットまたは削減します。
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カットしますが、レジューム動作は検出できるように維持します。
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、デバイス内部の動作を停止させることもできます。

デバイスがサスペンドモードにあるときに USB イベントが発生した場合には、レジューム手順を呼び出して公称クロックを復帰させ、USB 動作を通常の状態に戻す必要があります。発生したイベントが USB リセットシーケンスの場合、このプロセスが 10 mS 以内に終わるように特に注意を払う必要があります（詳細については、「Universal Serial Bus Specification」を参照してください）。USB peripheral 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 ビットが非同期的にクリアされます。このイベントが WKUP 割り込み（有効な場合）をトリガできる場合でも、システムクロックのリスタートには長い遅延が伴うため、割り込み応答ルーチンの使用には十分な注意が必要です。公称クロックが回復するまでの遅延を短縮するには、サスペンド手順の直後にレジューム手順を配置し、システムクロックがリスタートしたらすぐにそのコードが実行されるようにすることを推奨します。システムのウェイクアップによる ESD 放電やその他のノイズを避けるには（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で適切なアナログフィルタを起動します。フィルタ幅は約 70 ns です。

レジューム手順で対処すべき操作は、次のとおりです。

1. オプションで、外部オシレータやデバイス 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. レジュームトリガイイベントを識別する必要がある場合には、表 101 に従って USB_FNR レジスタの RXDP ビットと RXDM ビットを使用できます。この表には、あらゆる場合に予定されるソフトウェア動作も示されています。必要な場合には、上記のビットのステータスを監視して（バスのアイドル状態を示す「10」設定になったかどうかをチェックして）、レジュームまたはリセットシーケンスの終わりを検出できます。さらに、リセットシーケンスの終わりでは、USB_ISTR レジスタの RESET ビットが 1 にセットされ、割り込みが発行され（有効な場合）、これは通常の手順で処理されます。

表 101. 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動作
「00」	ルートリセット	なし
「10」	なし（バス上のノイズ）	サスペンドモードに戻る
「01」	ルートレジューム	なし
「11」	禁止（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への応答として、サスペンドモードを終了する必要があることがあります（マウスを動かすことによるシステム全体のウェイクアップなど）。この場合、レジュームシーケンスを開始するには、USB_CNTR レジスタの RESUME ビットを 1 にセットし、1 から 15 mS のインターバル後に 0 にリセットします（このインターバルを計るには、システムクロックが公称周波数で動作しているときに 1 mS の周期で発生する ESOF 割り込みを使用できます）。RESUME ビットがクリアされると、ホスト PC によってレジュームシーケンスが完了し、その完了も USB_FNR レジスタの RXDP ビットと RXDM ビットで監視できます。

注： RESUME ビットは、USB peripheral をサスペンドモードにして USB_CNTR レジスタの FSUSP ビットを 1 にセットしなければ使用できません。

24.5 USB レジスタ

USB peripheral レジスタは、次のグループに分類できます。

- 一般レジスタ：割り込みおよび制御レジスタ
- エンドポイントレジスタ：エンドポイントの設定とステータス
- バッファディスクリプタテーブル：データバッファの検索に使用されるパケットメモリの位置

すべてのレジスタアドレスは、USB peripheral レジスタのベースアドレス 0x4000 5C00 を基準にしたオフセットとして表現されます。ただし、バッファディスクリプタテーブルの位置は、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。ワード単位のアドレス指定には APB1 ブリッジという共通の制約があるため、すべてのレジスタアドレスは、16ビット幅であっても、32ビットのワード境界に配置されます。0x4000 6000 から始まるパケットバッファのメモリ位置にアクセスする場合にも、同じアドレス配置が使用されます。

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）単位でアクセスできます。

24.5.1 一般レジスタ

これらのレジスタは、USB peripheral の一般的な動作に影響し、動作モード、割り込み処理、デバイスアドレスを定義し、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を提供します。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTRM	PMAOVRM	ERRM	WKUPM	SUSPM	RESETM	SOFM	ESOFM	予約済み			RESUME	FSUSP	LP_MODE	PDWN	FRES
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 **CTRM**：転送の正常終了割り込みマスク

0：転送の正常終了（CTR）割り込みは無効です。

1：CTR 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、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 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、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：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。USB_ISTR レジスタの対応するビットがセットされると、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 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、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 : サスペンド割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、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 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、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 (SOF) 割り込みマスク

0 : SOF 割り込みは無効です。

1 : SOF 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、割り込みリクエストが生成されます。

ビット 8 ESOFM : フレーム開始欠落割り込みマスク

0 : フレーム開始欠落 (ESOF) 割り込みは無効です。

1 : フレーム開始欠落 (ESOF) 割り込みは有効です。USB_ISTR レジスタの対応するビットがセットされると、割り込みリクエストが生成されます。

ビット 7:5 予約済み。**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、1 から 15 ms の間有効にしてから、ホスト PC はレジュームシーケンスを最後まで駆動できるようになります。

ビット 3 FSUSP : 強制サスペンド

ソフトウェアは、USB peripheral が 3 ms の間トラフィックを受信しなかったときに発行される SUSP 割り込みを受信したときに、このビットをセットする必要があります。

0 : 影響ありません。

1 : サスペンドモードに入ります。クロックとアナログトランシーバの静的消費電力は影響を受けません。バスパワーデバイスで消費電力をサスペンド状態にする必要がある場合は、後述のようにアプリケーションソフトウェアは FSUSP 後に 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、サスペンドモードでの電力制約によって、外部プルアップ抵抗への供給を除くすべての静的消費電力を避ける必要があるときに使用されます。このモードに入るのは、USB サスペンド状態の消費電力条件を満たすために、アプリケーションがすべてのシステムクロックを停止させ、その周波数を減らしたりする用意が整ったときです。サスペンドモード中の USB 動作 (WKUP イベント) によってこのビットが非同期にリセットされます (ソフトウェアによるリセットも可能です)。

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 peripheral を完全に無効にする必要がある場合に、USB 関連のすべてのアナログ部品を完全にスイッチオフするのに使用します。このビットがセットされると、USB peripheral はトランシーバから切断され、使用できなくなります。

0 : パワーダウンモードから復帰します。

1 : パワーダウンモードに入ります。

ビット 0 FRES : USB リセットを強制します。

0 : USB リセットをクリアします。

1 : USB peripheral を強制的にリセットします。USB 上の RESET 信号と同じです。USB peripheral は、ソフトウェアがこのビットをクリアするまで、リセット状態に保持されます。「USB-RESET」割り込みが生成されます (有効な場合)。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTR	PMA OVR	ERR	WKUP	SUSP	RESET	SOF	ESOF	予約済み				DIR	EP_ID[3:0]		
r	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0					r	r	r	r

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み、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と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には、特定のイベントを表すシングルビット群が格納されています。これらのビットは、関連するイベントの発生時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ると、汎用割り込みリクエストが生成されます。割り込みルーチンは、各ビットをチェックし、必要な動作を実行した上で、処理済みのビットをクリアします。クリアされていないビットがあれば、割り込みはまだ保留中とみなされ、割り込みラインは再びハイレベルに保持されます。いくつかのビットが同時にセットされた場合、生成される割り込みは 1 つだけです。

エンドポイントのトランザクション完了は、別の方法で対処して割り込み応答の遅延を軽減できます。エンドポイントでトランザクションが正常に完了すると、ハードウェアによって CTR ビットがセットされ、USB_CNTR の対応するビットがセットされている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の割り込み条件は、USB_CNTR レジスタの CTRM ビットとは無関係に有効にされます。どちらの割り込み条件も、ソフトウェアが対応する USB_EPnR レジスタのペンディングビットをクリアするまで有効なままです（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、DIR（Direction of Transaction）ビットと 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行ったエンドポイントと、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識別できます。

ユーザーは、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して、同時にペンディングしている USB_ISTR イベントの相対的な優先順位を選択できます。処理済みのイベントに関連するビットのみがクリアされます。サービスルーチンの最後に、残りの条件を処理する別の割り込みがリクエストされます。

ビットのスプリアスなクリアを避けるには、変更してはならないビットには 1 を書き込み、クリアすべきビットには 0 を書き込むロード命令によるビットのクリアを推奨します（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／書き込みサイクルの使用は避けます。読み出しと書き込みの間にハードウェアによって別のビットがセットされ、マイクロプロセッサがイベントを処理する前に、次の書き込みでそのビットがクリアされる可能性があります。

以下に、個々のビットの詳細を示します。

ビット 15 CTR : 転送の正常終了

このビットはハードウェアによってセットされ、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します。ソフトウェアは、DIR と EP_ID ビットを使用して、割り込みを要求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が USB のメモリリクエストに時間内に応答できなかった場合にセットされます。USB peripheral は、このイベントを次のように処理します。受信時には、ACK ハンドシェイクパケットは送信されず、送信時には、送信されたストリームに強制的にビットスタッフエラーが設定されます。どちらの場合も、ホストはトランザクションを再試行します。PMAOVR 割り込みは、通常の動作時には発生しません。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ため、アプリケーションソフトウェアはこの割り込み処理時にデバイス操作のスピードアップを図り、次のトランザクションの再試行の準備をすることができます。ただし、アイソクロナス転送では再試行は行われず、この場合データは失われます。これは読み書きビットですが、書き込むことができるのは 0 のみで、1 を書き込んでも効果はありません。

ビット 13 ERR : エラー

このフラグは、次のいずれかのエラーが発生したときにセットされます。

NANS : 応答なし (No Answer)。ホストからの応答がタイムアウトしました。

CRC : 巡回冗長検査エラー。受信した CRC のトークンまたはデータにエラーがありました。

BST : ビットスタッフイングエラー。PID、データ、または CRC でビットスタッフイングエラーが検出されました。

FVIO : フレームフォーマット違反。非標準フレームを受信しました (EOP が正しい位置にない、トークンシーケンスの誤り、など)。

USB ソフトウェアは、通常、エラーを無視できます。エラーが発生した場合には、USB peripheral と PC ホストが完全に透過的な方法で再送信を管理します。この割り込みは、ソフトウェア開発段階において、または USB バスでの転送品質の監視にて考えられる問題 (コネクタの緩み、ノイズが多い環境、USB ケーブルの導体破損など) を警告するのに役立つことがあります。これは読み書きビットですが、書き込むことができるのは 0 のみで、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、サスペンドモードで USB peripheral をウェイクアップする動作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 ラインを有効にします。これを使用して、レジュームプロセスが開始されたことを残りのデバイス (ウェイクアップユニットなど) に知らせることができます。これは読み書きビットですが、書き込むことができるのは 0 のみで、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、USB バス上で 3 msec の間トラフィックが存在しないサスペンドモードリクエス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。サスペンド条件のチェックは、USB リセット後ただちに有効になり、サスペンドモードが有効になると (FSUSP=1)、レジュームシーケンスの終わりまでハードウェアによって無効になります。これは読み書きビットですが、書き込むことができるのは 0 のみで、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 peripheral が入力で有効な USB RESET 信号を検出したときにセットされます。USB peripheral は、RESET に応えて、内部プロトコルステート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は割り込みを生成します。送受信は、RESET ビットがクリアされるまで無効です。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 割り込みが確実に送信され、RESET の直前のトランザクションが完了されるようにするためです)。この機能のアドレスと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。これは読み書きビットですが、書き込むことができるのは 0 のみで、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 フレームの開始を通知し、SOF パケットが到着すると USB バスを介してセットされます。割り込みサービ斯拉ーチンは、SOF イベントを監視して USB ホストに対して 1 mS の同期をとり、SOF パケットの受信時に更新される USB_FNR レジスタを確実に読み出すこともできます（これはアイソクロナスアプリケーションで有用なことがあります）。これは読み書きビットですが、書き込むことができるのは 0 のみで、1 を書き込んでも効果はありません。

ビット 8 ESOFF : フレーム開始欠落

このビットは、SOF パケットを待っているにもかかわらず受信されない場合に、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、ハブがこれを正しく受信しない場合、サスペンドタイマがこの割り込みを発行します。トラフィックがまったく発生せず、ESOF 割り込みが 3 つ連続して生成される（つまり、3 つの SOF パケットが失われた）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 パケットが失われる場合にセットされます。これは読み書きビットですが、書き込むことができるのは 0 のみで、1 を書き込んでも効果はありません。

ビット 7:5 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、正しく完了し、割り込みリクエストを生成したトランザクションの方向に基づいて、ハードウェアによって書き込まれます。

DIR ビットが 0 の場合、割り込み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。割り込みトランザクションは IN 型です（データは USB peripheral によってホスト PC に送信されます）。

DIR ビットが 1 の場合、割り込み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/CTR_RX の両方がセットされます。割り込みトランザクションは OUT 型です（ホスト PC からのデータを USB peripheral が受信）。あるいは、2 つのペンディングトランザクションが処理を待っています。

これは保留中の割り込みの方向を表すため、アプリケーションソフトウェアはこの情報を使用して、トランザクションをトリガした 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優先順位が最も高い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。2 つのエンドポイントセットは、優先度の高い順に定義されます。アイソクロナスエンドポイントとダブルバッファバブルクエンドポイントが最初に検討され、続いて他のエンドポイントが調査されます。同じセットの複数のエンドポイントが割り込みを要求している場合、USB_ISTR レジスタの EP_ID ビットは、リクエストしている最下位のエンドポイントレジスタ、優先順位が一番高い EP0R、続いて EP1R という具合に割り当てられます。アプリケーションソフトウェアは、この優先順位方式に基づいて各エンドポイントにレジスタを割り当て、同時に発生するエンドポイントリクエストを適切に順序付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 + アップストリームポートデータラインのステータスを監視できます。このビットは、「end-of-suspend」ルーチンでウェイクアップイベントの判定に使用できます。

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 - アップストリームポートデータラインのステータスを監視できます。このビットは、「end-of-suspend」ルーチンでウェイクアップイベントの判定に使用できます。

ビット 13 LCK：ロック状態

このビットは、USB リセット条件または USB レジュームシーケンスが終わった後、2 個以上の連続した SOF パケットを受信した場合にハードウェアによってセットされます。ロック状態になると、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を維持します。

ビット 12:11 LSOF[1:0]：SOF 欠落

これらのビットは、ESOF 割り込みが生成され、連続した SOF パケットの欠落をカウントした場合に、ハードウェアによって書き込まれます。SOF パケットを受信すると、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、最後に受信した SOF パケットに格納されていた 11 ビットのフレーム番号を含みます。フレーム番号は、ホストからフレームが送信されるたびにインクリメントされ、アイソクロナス転送に役立ちます。このビットフィールドは、SOF 割り込みの生成によって更新されます。

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
予約済み								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**：機能有効化

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] ビットに格納されます。このビットが 0 の場合、USB_EPnR レジスタの設定に関わらず、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でホスト PC によって割り当てられる USB 機能アドレスを含みます。要求されたエンドポイントへのトランザクションが処理されるためには、このフィールドと、関連する USB_EPnR レジスタの EA（エンドポイントアドレス）フィールドの両方が、USB トークンに含まれている情報と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]													予約済み		
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示し、8バイト境界に配置される必要があります（最下位 3 ビットは常に 0 です）。このデバイスに指定された個々のトランザクションの最初で、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テーブルの要素を読み出し、その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使用 \(612 ページ\)](#) 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。

24.5.2 エンドポイント固有のレジスタ

これらのレジスタの数は、USB peripheral の設計で扱うことになっているエンドポイントの数によって異なります。USB peripheral は、最大 8 個の双方向エンドポイントをサポートします。各 USB デバイスはコントロールエンドポイントをサポートする必要があり、そのアドレス（EA ビット）は 0 にセットする必要があります。同じエンドポイント番号をもつ複数のエンドポイントが有効にされた場合、USB peripheral の動作は不定です。エンドポイントごとに、エンドポイント固有の情報を格納する USB_EPnR レジスタが用意されています。

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 バスから USB リセットを受信した場合や、CTRL レジスタの FRES ビットによって USB リセットが強制された場合は、これらのレジスタもリセットされます。ただし、USB リセットイベントの直前にある正規の packets 通知を見逃さないように、CTR_RX ビットと CTR_TX ビットは変更されません。各エンドポイントには USB_EPnR レジスタがあります（n はエンドポイント識別子です）。

読み出し／変更／書き込みサイクルの使用は避けます。読み出しと書き込みの間にハードウェアによって別のビットがセットされ、CPU が変更を検出する前に、次の書き込みでそのビットが変更される可能性があります。このため、この問題に関連するビットでは、ビット値の変更の必要がない場合に必ず使用する「不変」値が用意されています。これらのレジスタを変更する場合、ハードウェアによってのみ変更できるすべてのビットに対して「不変」値を書き込むロード命令を使用することが推奨されます。

ビット 15 **CTR_RX** : 受信転送の正常終了

このビットは、このエンドポイントに対する OUT/SETUP トランザクションが正常に完了したとき、ハードウェアによってセットされます。ソフトウェアは、このビットのクリアのみ可能です。USB_CNTR レジスタの CTRM ビットが正しくセットされていれば、常に有効なエンドポイント関連の割り込み条件で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、次に示す SETUP ビットから判断できます。

NAK または STALL ハンドシェイクで終わるトランザクションは、このビットをセットしません。これは、プロトコルエラーやデータトグルの不一致と同様に、実際にはデータが転送されないからです。

これは読み書きビットですが、0 のみを書き込むことができます。1 を書き込んでも効果はありません。

ビット 14 **DTOG_RX** : 受信転送のデータトグル

エンドポイントがアイソクロナスでない場合、このビットには、次に受信するデータパケットのデータトグルビットの期待値 (0=DATA0、1=DATA1) が含まれています。一致するデータ PID 値を持つデータパケットの受信に続いて ACK ハンドシェイクが USB ホストに送信されると、ハードウェアはこのビットをトグルします。エンドポイントのタイプがコントロールの場合、ハードウェアはこのエンドポイントに指定された 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の交換のサポートにも使用されます ([セクション 24.4.3 : 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である場合、このビットはパケットバッファの交換のサポートにのみ使用されます。これはこの種のエンドポイントに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からです ([セクション 24.4.4 : 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が使用されないため、ハードウェアは、データパケット受信の直後にこのビットをトグルします。

ソフトウェアがこのビットをトグルすると、値を初期化したり (コントロールエンドポイント以外では必須)、特定のデータトグル/パケットバッファ使用を強制したりできます。アプリケーションソフトウェアが 0 を書き込んでも DTOG_RX の値は変化しませんが、1 を書き込むとビット値はトグルします。これは読み書きビットですが、1 を書き込むことによってトグルのみできます。

ビット 13:12 **STAT_RX [1:0]** : 受信転送のステータスビット

これらのビットには、[表 102 : 受信ステータスエンコーディング \(632 ページ\)](#) 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が含まれており、ソフトウェアで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 を書き込んでも値は変化しませんが、1 を書き込むとビット値はトグルします。このエンドポイントに指定された OUT または SETUP (制御のみ) トランザクションに対応して転送が正常に終了したとき (CTR_RX=1)、ハードウェアは STAT_RX ビットに NAK を設定します。したがって、ソフトウェアは、新しいトランザクションをアクノレッジする前に、受信したデータを調べる時間を確保できます。

ダブルバッファバルクエンドポイントは、特殊なトランザクションフロー制御を実装し、バッファの利用可能状況に基づいてステータスが制御されます ([セクション 24.4.3 : 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」であるため、正常なトランザクション後にハードウェアが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。アイソクロナスエンドポイントに対して、ソフトウェアが STAT_RX ビットを「STALL」または「NAK」にセットした場合、USB peripheral の動作は不定です。これらは読み書きビットですが、1 を書き込むことによってトグルのみできます。

ビット 11 **SETUP** : SETUP 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 であった場合に、ハードウェアによってセットされます。このビットは、コントロールエンドポイントの場合にのみ値が変わります。正常な受信トランザクションの場合 (CTR_RX イベント)、このビットを調べて、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。次の着信トークンによる SETUP ビットの変化から割り込みサービスルーチンを保護するため、CTR_RX ビットが 1 の間はこのビットは固定され、CTR_RX ビットが 0 になると状態が変化します。このビットは読み出し専用です。

ビット 10:9 EP_TYPE[1:0] : エンドポイントのタイプ

これらのビットは、このエンドポイントの動作を [表 103 : エンドポイントタイプエンコーディング \(633 ページ\)](#) に示されているように設定します。エンドポイント 0 は常にコントロールエンドポイントである必要があり、各 USB 機能には、アドレス 0 のコントロールエンドポイントが 1 つ以上必要ですが、必要な場合には、他にもコントロールエンドポイントがあってもかまいません。コントロールエンドポイントだけが SETUP トランザクションを扱うことができます。他の種類のエンドポイントでは SETUP トランザクションは無視されます。SETUP トランザクションに対して NAK や STALL で応答することはできません。コントロールエンドポイントが NAK として定義された場合、USB peripheral は応答せず、SETUP トランザクションの受信時に受信方向で受信エラーをシミュレートします。コントロールエンドポイントが受信方向で STALL として定義されても、SETUP パケットは受け付けられるため、データは転送され、CTR 割り込みが発行されます。OUT トランザクションの受信は、エンドポイントがコントロールエンドポイントの場合でも、通常どおりに処理されます。

バルクおよび割り込みエンドポイントはほとんど同じように動作しますが、EP_KIND 設定ビットを使用する特殊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い方は、[セクション 24.4.4 : アイソクロナス転送](#)に記載されています。

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タイプによって異なります。[表 104](#)意味の違いを示します。

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い方については、[セクション 24.4.3 : ダブルバッファエンドポイント](#)を参照してください。

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期待されてい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 バイト以上のデータを含むすべての OUT トランザクションは、すべて"ACK"ではなく"STALL"になります。このビットを使用して、コントロール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高めることができます。この使用は、コントロール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バイト数を扱うことができます。

ビット 7 CTR_TX : 送信転送の正常終了

このビットは、このエンドポイントに対する 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。ソフトウェアは、このビットのクリアのみ可能です。USB_CNTR レジスタの CTRM ビットが正しくセットされていれば、常に有効なエンドポイント関連の割り込み条件で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わるトランザクションは、このビットをセットしません。これは、プロトコルエラーやデータトグルの不一致と同様に、実際にはデータが転送されないからです。

これは読み書きビットですが、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、このビットには次に送信されるデータパケットのデータトグルビットの必要な値（0=DATA0、1=DATA1）が含まれます。データパケット送信に続いて、USB ホストから ACK ハンドシェイクを受信すると、ハードウェアがこのビットをトグルします。エンドポイントのタイプがコントロールである場合、このエンドポイントに指定された SETUP PID の受信時に、ハードウェアがこのビットを 1 に設定します。

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の交換のサポートにも使用されます（[セクション 24.4.3 : 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である場合、このビットはパケットバッファの交換のサポートに使用されます。これはこの種のエンドポイントに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からです（[セクション 24.4.4 : 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が使用されないため、ハードウェアは、データパケット送信の直後にこのビットをトグルします。

ソフトウェアがこのビットをトグルすると、値を初期化したり（コントロールエンドポイント以外では必須）、特定のデータトグル/パケットバッファ使用を強制したりできます。アプリケーションソフトウェアが 0 を書き込んでも DTOG_TX の値は変化しませんが、1 を書き込むとビット値はトグルします。これは読み書きビットですが、1 を書き込むことによってトグルのみできます。

ビット 5:4 **STAT_TX [1:0]** : 送信転送のステータスビット

これらのビットには、[表 105](#) にリストされているエンドポイントステータスの情報が含まれています。これらのビットは、値を初期化するために、ソフトウェアによってトグルできます。アプリケーションソフトウェアが 0 を書き込んでも値は変化しませんが、1 を書き込むとビット値はトグルします。このエンドポイントに指定された IN または SETUP（コントロールのみ）トランザクションに対応して転送が正常に終了したとき（CTR_TX=1）、ハードウェアは STAT_TX ビットに NAK を設定します。この後、次に送信されるデータセットがソフトウェアによって準備されるのを待ちます。

ダブルバッファバルクエンドポイントは、特殊なトランザクションフロー制御を実装し、バッファの利用可能状況に基づいてステータスが制御されます（[セクション 24.4.3 : 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である場合、そのステータスとしては「VALID」または「DISABLED」のみ可能です。したがって、トランザクションが正常に処理された後、ハードウェアはエンドポイントのステータスを変更できません。アイソクロナスエンドポイントに対して、ソフトウェアが STAT_TX ビットを「STALL」または「NAK」にセットした場合、USB peripheral の動作は不定です。これらは読み書きビットですが、1 を書き込むことによってトグルのみできます。

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、このフィールドに、このエンドポイントに送信されたトランザクションを識別するための 4 ビットアドレスを書き込む必要があります。対応するエンドポイントを有効にする前に、値を書き込む必要があります。

表 102. 受信ステータスエンコーディング

STAT_RX[1:0]	意味
00	DISABLED : このエンドポイントへのすべての受信リクエストは無視されます。
01	STALL : エンドポイントはストール状態であり、すべての受信リクエストに対して STALL ハンドシェイクが返されます。
10	NAK : エンドポイントは NAK となり、すべての受信リクエストに対して NAK ハンドシェイクが返されます。
11	VALID : このエンドポイントは受信を行うことができます。



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。

表 103. エンドポイントタイプエンコーディング

EP_TYPE[1:0]	意味
00	バルク
01	制御
10	アイソクロナス
11	割り込み

表 104. エンドポイントの種類の意味

EP_TYPE[1:0]	EP_KIND の意味
00	バルク DBL_BUF
01	制御 STATUS_OUT
10	アイソクロナス 未使用
11	割り込み 未使用

表 105. 送信ステータスのエンコーディング

STAT_TX[1:0]	意味
00	DISABLED : このエンドポイントへのすべての送信リクエストは無視されます。
01	STALL : エンドポイントはストール状態であり、すべての送信リクエストに対して STALL ハンドシェイクが返されます。
10	NAK : エンドポイントは NAK となり、すべての送信リクエストに対して NAK ハンドシェイクが返されます。
11	VALID : このエンドポイントは送信を行うことができます。

24.5.3 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 STM32L1xxxx との間でデータ交換を行う際に使用するパケットバッファの位置とサイズを設定する追加のレジスタとみなすことができます。ワードアドレス指定に関する一般的な APB ブリッジ制約によって、APB は 32ビットに配置されたアドレスを使用してすべてのパケットメモリ位置にアクセスします。したがって、USB_BTABLE レジスタやバッファ記述テーブルの位置に対して USB ペリフェラルが使用する実際のメモリ位置アドレスは使用されません。

以下のページには 2 つの位置アドレスが記載されています。1 つは、パケットメモリにアクセスするときに、アプリケーションソフトウェアによって使用されます。もう 1 つは、USB ペリフェラルアクセスに関連するローカルアドレスです。アプリケーションソフトウェアがパケットメモリにアクセスする際に使用する正しい STM32L1xxxx メモリアドレス値を得るには、実際のメモリ位置アドレスを 2 倍にする必要があります。最初のパケットメモリ位置は 0x4000 6000 です。以下に、USB_EPnR レジスタに関連付けられたバッファディスクリプタテーブルのエントリを示します。

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い方については、[パケットバッファの構造と使用 \(612 ページ\)](#) 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*16

USB ローカルアドレス : [USB_BTABLE] + n*8

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、アドレス指定された次の IN トークンで USB_EPnR レジスタに関連付けられたエンドポイントによって送信されるデータを含むパケットバッファの開始アドレスを示します。

ビット 0 パケットメモリはワード幅であり、すべてのパケットバッファはワード配置されている必要があるため、常に 0 として書き込まれます。



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*16 + 4

USB ローカルアドレス : [USB_BTABLE] + n*8 + 2

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み						COUNTn_TX[9:0]									
						rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 15:10 USB 仕様によるパケットサイズの上限は 1023 バイトであるため、これらのビットは使用されません。
USB peripheral はこれらのビットの値を無視します。

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、アドレス指定された次の IN トークンで USB_EPnR レジスタに関連付けられたエンドポイントによって送信されるバイト数を含みます。

注 : **ダブルバッファおよびアイソクロナスの IN エンドポイントには、以下の内容を持つ 2 個の USB_COUNTn_TX レジスタ (USB_COUNTn_TX_1 と USB_COUNTn_TX_0) があります。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
予約済み						COUNTn_TX_1[9:0]									
						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
予約済み						COUNTn_TX_0[9:0]									
						rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

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*16 + 8

USB ローカルアドレス : [USB_BTABLE] + n*8 + 4

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRn_RX[15:1]															-
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	-

ビット 15:1 **ADDRn_RX[15:1]** : 受信バッファアドレス

これらのビットは、アドレス指定された次の OUT/SETUP トークンで USB_EPnR レジスタに関連付けられたエンドポイントによって受信されるデータを含むパケットバッファの開始アドレスを示します。

ビット 0 パケットメモリはワード幅であり、すべてのパケットバッファはワード配置されている必要があるため、このビットは常に 0 として書き込まれます。

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*16 + 12

USB ローカルアドレス : [USB_BTABLE] + n*8 + 6

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、パケット受信時に必要な 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、割り当てられたバッファサイズの定義を含み、バッファオーバーフローの検出を可能にします。一方、この位置の最下位部分には、USB peripheral によって受信の最後に実際の受信バイト数が書き戻されます。使用可能なビット数に関する制約により、バッファサイズは割り当てられたメモリブロックの数を使用して表現されます。ここでは、ブロックサイズは粒度の細かい小さなバッファと粒度の粗い大きなバッファとの間のトレードオフで選択できます。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、通常はエニュメレーションプロセス時に maxPacketSize パラメータ値に基づいて定義されます（「Universal Serial Bus Specification」を参照）。

ビット 15 BL_SIZE : ブロックサイズ

このビットは、割り当てられるバッファ領域の定義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、メモリブロックの大きさは 2 バイトであり、これはワード幅メモリで許容される最小ブロックです。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 から 62 バイトです。
- BL_SIZE=1 の場合、メモリブロックの大きさは 32 バイトであり、USB 仕様で規定される最大のパケット長が指定できます。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 32 から 1024 バイトであり、これは USB 標準仕様で許容される最長のパケットサイズです。

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の数を定義します。割り当てられるメモリの実際の量は、表 106 に示すように、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、アドレス指定された最後の OUT/SETUP トランザクションにおいて USB_EPnR レジスタに関連付けられたエンドポイントによって受信されたバイト数を含みます。

注 : **ダブルバッファおよびアイソクロナスの IN エンドポイントには、以下の内容を持つ 2 個の USB_COUNTn_TX レジスタ (USB_COUNTn_TX_1 と USB_COUNTn_TX_0) があります。**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLSIZE_1	NUM_BLOCK_1[4:0]					COUNTn_RX_1[9:0]									
rw	rw	rw	rw	rw	rw	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
BLSIZE_0	NUM_BLOCK_0[4:0]					COUNTn_RX_0[9:0]									
rw	rw	rw	rw	rw	rw	r	r	r	r	r	r	r	r	r	r



表 106. 割り当てられるバッファメモリの定義

値 NUM_BLOCK[4:0]	BL_SIZE=0 の場合に 割り当てられるメモリ	BL_SIZE=0 の場合に 割り当てられるメモリ
0 (00000)	設定禁止	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
...	...	...
15 (01111)	30バイト	512バイト
16 (10000)	32バイト	N/A
17 (10001)	34バイト	N/A
18 (10010)	36バイト	N/A
...	...	...
30 (11110)	60バイト	N/A
31 (11111)	62バイト	N/A

24.5.4 USB レジスタマップ

次の表に、USB のレジスタマップとリセット値を示します。

表 107. 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	予約済み																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	予約済み																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	予約済み																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	予約済み																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	予約済み																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	予約済み																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 107. 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														
0x18	USB_EP6R	予約済み																CTR_RX	DT0G_RX	STAT_RX [1:0]		SETUP	EP_TYPE [1:0]		EP_KIND	CTR_TX	DT0G_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	0	0	0											
0x1C	USB_EP7R	予約済み																CTR_RX	DT0G_RX	STAT_RX [1:0]		SETUP	EP_TYPE [1:0]		EP_KIND	CTR_TX	DT0G_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	0	0	0	0	0									
0x20-0x3F	予約済み																																														
0x40	USB_CNTR	予約済み																CTRM	PMAOVRM	ERRM	WKUPM	SUSPM	RESETM	SOFM	ESOFM	予約済み				RESUME	FSUSP	LPMODE	PDWN	FRES													
	リセット値																	0	0	0	0	0	0	0	0					0	0	0	1	1													
0x44	USB_ISTR	予約済み																CTR	PMAOVR	ERR	WKUP	SUSP	リセット	SOF	ESOF	予約済み				DIR	EP_ID[3:0]																
	リセット値																	0	0	0	0	0	0	0	0					0	0	0	0	0													
0x48	USB_FNR	予約済み																RXDP	RXDM	LCK	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	x	x	x	x										
0x4C	USB_DADDR	予約済み																										EF		ADD[6:0]																	
	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
0x50	USB_BTABLERE	予約済み																BTABLE[15:3]														予約済み															
	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									

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。



25 フレキシブルスタティックメモリコントローラ (FSMC)

このセクションは、Cat4デバイスにのみ適用されます。FSMC の利用可能性については、デバイスデータシートを参照してください。

25.1 FSMC の主な機能

FSMC ブロックは、同期および非同期メモリとインタフェースできます。主な目的は、次のとおりです。

- AHB トランザクションを適切な外部プロトコルへ変換
- 外部デバイスのアクセスタイミング要件を満たす

すべての外部メモリは、アドレス、データ、および制御信号をコントローラと共有します。各外部デバイスは、一意なチップセレクトでアクセスします。FSMC は、外部デバイスに対して、一度に 1 つのアクセスを行います。

FSMC は、以下のような主な機能を備えています。

- スタティックメモリマップデバイスとのインタフェース
 - － スタティックランダムアクセスメモリ (SRAM)
 - － NOR 型フラッシュメモリ / 1 つの NAND 型フラッシュメモリ
 - － PSRAM (4 個のメモリバンク)
- 同期デバイスへのバーストモードアクセスをサポート (NOR 型フラッシュおよび PSRAM)
- 8 または 16 ビット幅のデータバス
- メモリバンクごとに独立したチップセレクト制御
- メモリバンクごとに独立した構成
- 広範囲のデバイスをサポートするプログラマブルタイミング
 - － プログラマブルウェイトステート (最大 15 個)
 - － プログラマブルバスターンアラウンドサイクル (最大 15 個)
 - － プログラマブルな出カインエーブルおよび書き込みインエーブル遅延 (15 まで)
 - － 独立した読み出しおよび書き込みタイミングとプロトコルにより、広範囲のメモリおよびタイミングをサポート
- PSRAM および SRAM デバイス用書き込みインエーブルとバイトレーン選択出力
- 32 ビット幅 AHB トランザクションを外部 16 ビットまたは 8 ビットデバイスへの連続した 16 ビットまたは 8 ビットアクセスに変換
- アドレスではなくデータのみを格納する 2 ワード長の書き込み FIFO。各ワードは 32 ビット幅であり、したがって、この FIFO は AHB 書き込みバーストトランザクションのみをバッファします。これによって、低速メモリに書き込み、AHB を迅速に解放して他の操作に使用することができます。一度にバッファされるバーストは 1 つだけです。操作中に新しい AHB バーストまたは単一のトランザクションが発生した場合、最初に FIFO の内容が読み出されます。FSMC は、現在のメモリアクセスが完了するまで、ウェイトステートを挿入します。
- 外部非同期ウェイト制御

外部デバイスタイプおよびその関連特性を定義する FSMC レジスタは、通常、起動時に設定され、次のリセットまたは電源投入まで変更されません。ただし、設定の変更はいつでも可能です。

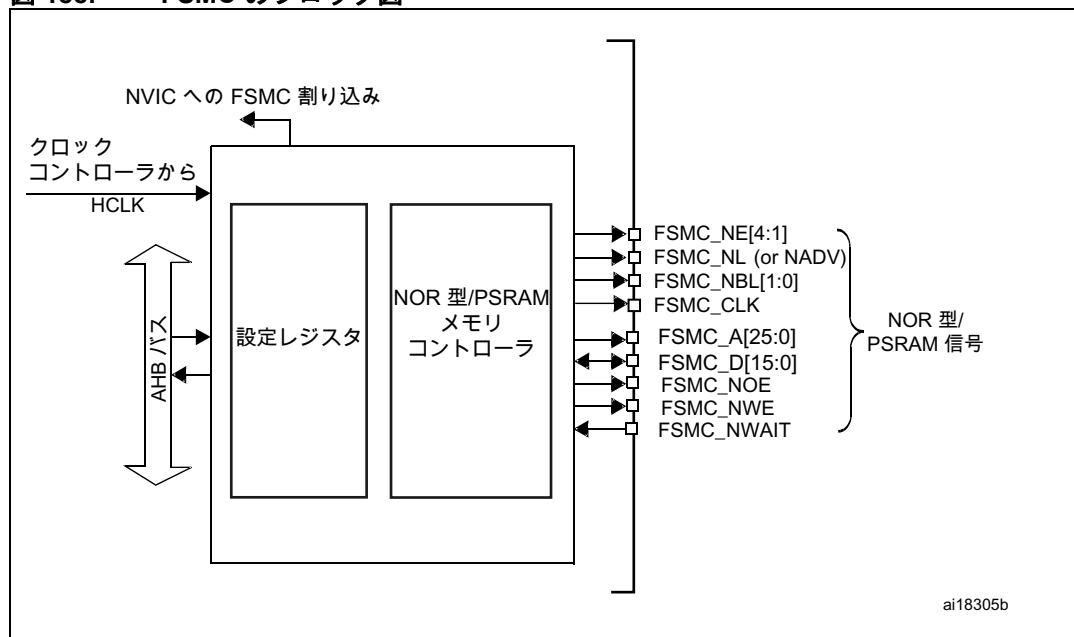
25.2 ブロック図

FSMC は 4 つのメインブロックで構成されます。

- AHB インタフェース (FSMC 設定レジスタを含む)
- NOR 型フラッシュ/PSRAM コントローラ
- 外部デバイスインタフェース

図 188 にブロック図を示します。

図 188. FSMC のブロック図



25.3 AHB インタフェース

AHB スレーブインタフェースは、内部 CPU および他のバスマスタペリフェラルから外部スタティックメモリへのアクセスを可能にします。

AHB トランザクションは、外部デバイスプロトコルに変換されます。特に、選択された外部メモリが 16 または 8 ビット幅の場合、AHB 上の 32 ビット幅のトランザクションは、16 または 8 ビットの連続したアクセスに分割されます。チップセレクトは、アクセス毎に反転します。

FSMC は、以下の条件下で AHB エラーを生成します。

- 有効でない FSMC バンクの読み出しまたは書き込みを行ったとき。
- FSMC_BCRx レジスタの FACCEN ビットがリセットされている状態で NOR 型フラッシュバンクの読み出しまたは書き込みを行ったとき。

この AHB エラーの影響は、R/W アクセスを試みた AHB マスタに依存します。

- Cortex®-M3 CPU の場合は、ハード障害割り込みが生成されます。
- DMA の場合は、DMA 転送エラーが生成され、対応する DMA チャンネルが自動的に無効になります。

AHB クロック (HCLK) は、FSMC の基準クロックです。

25.3.1 サポートされるメモリおよびトランザクション

一般的なトランザクション規則

リクエストされる AHB トランザクションのデータサイズは、アクセスされる外部デバイスのデータ幅が固定であっても、8、16、または 32ビット幅です。このため、転送に矛盾が生じることがあります。

したがって、次のようないくつかの単純なトランザクション規則を守る必要があります。

- AHB トランザクションサイズとメモリデータサイズが等しい場合。
この場合は問題ありません。
- AHB トランザクションサイズがメモリサイズより大きい場合。
この場合、FSMC は、AHB トランザクションを外部データ幅に合わせて、連続した小さなメモリアccessに分割します。
- AHB トランザクションサイズがメモリサイズより小さい場合。
外部デバイスのタイプによっては、非同期転送に一貫性がなくなる場合があります。
 - － バイト選択機能を持つデバイス (SRAM、ROM、PSRAM) への非同期アクセスの場合。
 - a) FSMC によって、バイトレーン NBL[1:0] を通じて正しいデータにアクセスする書き込みトランザクションが可能になります。
 - b) 読み出しトランザクションが可能になります。すべてのメモリバイトが読み出され、使用できないものは廃棄されます。読み出しトランザクション中、NBL[1:0] はローに維持されます。
 - － バイト選択機能を持たないデバイス (NOR 型) への非同期アクセスの場合。
この状況は、16ビット幅のフラッシュメモリへのバイトアクセスが要求されたときに発生します。明らかに、このデバイスにはバイトモードではアクセスできません (フラッシュメモリでは 16ビットワードでの読み出し/書き込みしかできません)。したがって、
 - a) 書き込みトランザクションは許可されません。
 - b) 読み出しトランザクションは許可されます。すべてのメモリバイトが読み出され、使用できないものは廃棄されます。読み出しトランザクション中、NBL[1:0] はローにセットされます。

設定レジスタ

FSMC は、レジスタセットを使用して設けます。NOR 型フラッシュ/PSRAM 制御レジスタの詳細については、[セクション 25.5.6](#) を参照してください。

25.4 外部デバイスアドレスマッピング

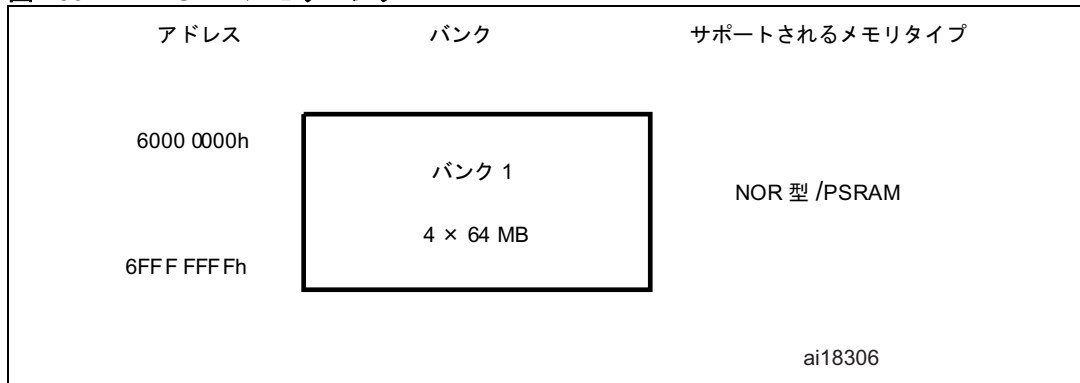
FSMC から見ると、外部メモリは 256 MB の 1 つの固定サイズのバンクで構成されています ([図 189](#) を参照)。

- バンク 1 は、4 つまでの NOR 型フラッシュまたは PSRAM メモリデバイスのアドレス指定に使用されます。このバンクは、次のように 4 つの専用チップセレクトを持つ 4 つの NOR/PSRAM サブバンクに分割されます。
 - － バンク 1 - NOR/PSRAM 1
 - － バンク 1 - NOR/PSRAM 2
 - － バンク 1 - NOR/PSRAM 3
 - － バンク 1 - NOR/PSRAM 4

各バンクで使用されるメモリのタイプは、ユーザーによって設定レジスタで定義されます。



図 189. FSMC メモリバンク



25.4.1 NOR/PSRAM アドレスマッピング

HADDR[27:26] ビットを使用して、表 108 に示される 4 つのメモリバンクの 1 つを選択します。

表 108. NOR/PSRAM バンク選択

HADDR[27:26] ⁽¹⁾	選択されるバンク
00	バンク 1 - NOR/PSRAM 1
01	バンク 1 - NOR/PSRAM 2
10	バンク 1 - NOR/PSRAM 3
11	バンク 1 - NOR/PSRAM 4

1. HADDR は、外部メモリに変換される内部 AHB アドレスラインです。

HADDR[25:0] は、外部メモリアドレスを含みます。HADDR はバイトアドレスですが、メモリはワード単位でアドレス指定されるため、メモリに対して実際に発行されるアドレスは、次の表に示すようにメモリのデータ幅に応じて異なります。

表 109. 外部メモリアドレス

メモリ幅 ⁽¹⁾	メモリに発行されるデータアドレス	最大メモリ容量 (ビット)
8ビット	HADDR[25:0]	64 MB x 8 = 512 M ビット
16ビット	HADDR[25:1] >> 1	64 MB/2 x 16 = 512 M ビット

1. 外部メモリが 16ビット幅の場合、FSMC は、内部で HADDR[25:1] を使用して、外部メモリ FSMC_A[24:0] に対するアドレスを生成します。
外部メモリの幅 (16ビットまたは 8ビット) にかかわらず、FSMC_A[0] は外部メモリアドレス A[0] に接続されます。

NOR 型フラッシュ/PSRAM のラップサポート

同期メモリのラップバーストモードはサポートされていません。メモリは、長さが未定義のリニアバーストモードに設定する必要があります。

25.5 NOR 型フラッシュ／PSRAM コントローラ

FSMCは、以下のタイプのメモリを駆動するのに適した信号タイミングを生成します。

- 非同期 SRAM および ROM
 - 8ビット
 - 16ビット
 - 32ビット
- PSRAM (セルラー RAM)
 - 非同期モード
 - 同期アクセス用バーストモード
- NOR 型フラッシュ
 - 非同期モード
 - 同期アクセス用バーストモード
 - マルチプレクスまたは非マルチプレクス

FSMC は、バンクごとに一意のチップセレクト信号 NE[4:1] を出力します。その他の信号（アドレス、データ、および制御）はすべて共有されます。

同期アクセスでは、FSMC は読み出し/書き込みトランザクションにおいてのみ、選択された外部デバイスにクロック (CLK) を送出します。このクロックは、HCLK クロックの約数です。各バンクのサイズは 64 MB に固定されます。

各バンクは、専用レジスタによって設定されます（[セクション 25.5.6](#) を参照）。

プログラマブルメモリパラメータにはアクセスタイミング（[表 110](#) を参照）が含まれており、ウェイト管理をサポートします（バーストモードでの PSRAM および NOR 型フラッシュへのアクセスの場合）。

表 110. プログラマブル NOR/PSRAM アクセスパラメータ

パラメータ	機能	アクセスモード	単位	最小値	最大値
アドレスセットアップ	アドレスセットアップフェーズ時間	非同期	AHB クロックサイクル (HCLK)	0	15
アドレスホールド	アドレスホールドフェーズ時間	非同期、マルチプレクス I/O	AHB クロックサイクル (HCLK)	1	15
データセットアップ	データセットアップフェーズ時間	非同期	AHB クロックサイクル (HCLK)	1	256
バスターン	バスターンアラウンドフェーズ時間	非同期および同期読み出し	AHB クロックサイクル (HCLK)	0	15
クロック分周比	1 つのメモリクロックサイクル (CLK) に必要な AHB クロックサイクル (HCLK) 数	同期	AHB クロックサイクル (HCLK)	2	16
データ遅延	バーストの最初のデータの前にメモリに発行されるクロックサイクル数	同期	メモリクロックサイクル (CLK)	2	17

25.5.1 外部メモリインタフェース信号

表 111、表 112、および 表 113 は、NOR 型フラッシュ、SRAM、および PSRAM へのインタフェースに通常使用される信号のリストです。

注：プレフィックス「N」は、関連する信号がアクティブローであることを示します。

NOR 型フラッシュ、非マルチプレクス I/O

表 111. 非マルチプレクス I/O NOR型フラッシュ

FSMC 信号名	入出力	機能
CLK	O	クロック（同期アクセス用）
A[25:0]	O	アドレスバス
D[15:0]	入出力	双方向データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力有効
NWE	O	書き込み有効
NL (=NADV)	O	ラッチ有効（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FSMC への NOR 型フラッシュウェイト入力信号

NOR 型フラッシュメモリは、16ビットワード単位でアドレス指定されます。最大容量は、512 M ビット（アドレスライン 26 個）です。

NOR 型フラッシュ、マルチプレクス I/O

表 112. マルチプレクス I/O NOR型フラッシュ

FSMC 信号名	入出力	機能
CLK	O	クロック（同期アクセス用）
A[25:16]	O	アドレスバス
AD[15:0]	入出力	16ビットマルチプレクス、双方向アドレス／データバス
NE[x]	O	チップセレクト、x = 1..4
NOE	O	出力有効
NWE	O	書き込み有効
NL (=NADV)	O	ラッチ有効（この信号は、一部の NOR 型フラッシュデバイスではアドレス有効（NADV）と呼ばれます）。
NWAIT	I	FSMC への NOR 型フラッシュウェイト入力信号

NOR 型フラッシュメモリは、16ビットワード単位でアドレス指定されます。最大容量は、512 M ビット（26 個のアドレスライン）です。

PSRAM/SRAM、非マルチプレクス I/O

表 113. 非マルチプレクス I/O、PSRAM/SRAM

FSMC 信号名	入出力	機能
CLK	O	クロック (PSRAM 同期アクセスの場合のみ)
A[25:0]	O	アドレスバス
D[15:0]	入出力	双方向データバス
NE[x]	O	チップセレクト、x = 1..4 (PSRAM (セルラー RAM、または CRAM) では NCE と呼ばれます。)
NOE	O	出力有効
NWE	O	書き込み有効
NL (= NADV)	O	PSRAM 入力の場合のみアドレス有効 (メモリ信号名: NADV)
NWAIT	I	FSMC への PSRAM ウェイト入力信号
NBL[1]	O	上位バイト有効 (メモリ信号名: NUB)
NBL[0]	O	下位バイト有効 (メモリ信号名: NLB)

PSRAM メモリは、16ビットワード単位でアドレス指定されます。最大容量は、512 M ビット (アドレスライン 26 個) です。

PSRAM、マルチプレクス I/O

表 114. マルチプレクス I/O PSRAM

FSMC 信号名	入出力	機能
CLK	O	クロック (同期アクセス用)
A[25:16]	O	アドレスバス
AD[15:0]	入出力	16ビットマルチプレクス、双方向アドレス/データバス
NE[x]	O	チップセレクト、x = 1..4 (PSRAM (セルラー RAM、または CRAM) では NCE と呼ばれます。)
NOE	O	出力有効
NWE	O	書き込み有効
NL (= NADV)	O	アドレス有効 PSRAM 入力 (メモリ信号名: NADV)
NWAIT	I	FSMC への PSRAM ウェイト入力信号
NBL[1]	O	上位バイト有効 (メモリ信号名: NUB)
NBL[0]	O	下位バイト有効 (メモリ信号名: NLB)

PSRAM メモリは、16ビットワード単位でアドレス指定されます。最大容量は、512 MB (アドレスライン 26 個) です。

25.5.2 サポートされるメモリおよびトランザクション

次の表 115 に、NOR、PSRAM、SRAM のメモリデータバスが 16 ビットの場合にサポートされるデバイス、アクセスモード、およびトランザクション例を示します。この例では、FSMC で許可されない（サポートされない）トランザクションをグレーで示しています。

表 115. NOR 型フラッシュ/PSRAM コントローラ：サポートされるメモリとトランザクションの例

デバイス	モード	R/W	AHB データサイズ	メモリデータサイズ	許可/禁止	コメント
NOR 型フラッシュ (マルチプレクス I/O と非マルチプレクス I/O)	非同期	R	8	16	Y	
	非同期	W	8	16	N	
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FSMC アクセスに分割
	非同期	W	32	16	Y	2 つの FSMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32	16	Y	
PSRAM (マルチプレクス I/O と非マルチプレクス I/O)	非同期	R	8	16	Y	
	非同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	16	16	Y	
	非同期	W	16	16	Y	
	非同期	R	32	16	Y	2 つの FSMC アクセスに分割
	非同期	W	32	16	Y	2 つの FSMC アクセスに分割
	非同期ページ	R	-	16	N	モードはサポートされていません。
	同期	R	8	16	N	
	同期	R	16	16	Y	
	同期	R	32	16	Y	
	同期	W	8	16	Y	バイトレーン NBL[1:0] の使用
	同期	W	16/32	16	Y	
SRAM と ROM	非同期	R	8/16	16	Y	
	非同期	W	8/16	16	Y	バイトレーン NBL[1:0] の使用
	非同期	R	32	16	Y	2 つの FSMC アクセスに分割
	非同期	W	32	16	Y	2 つの FSMC アクセスに分割 バイトレーン NBL[1:0] の使用

25.5.3 一般的なタイミング規則

信号の同期

- すべてのコントローラ出力信号は、内部クロック (HCLK) の立ち上がりエッジで変化します。
- 同期モード (読み出しまたは書き込み) では、すべての出力信号が HCLK の立ち上がりエッジで変化します。CLKDIV の値にかかわらず、すべての出力は次のように変化します。
 - NOEL/NWEL/ NEL/NADV/ NADVH /NBLL/ アドレス有効出力は、FSMC_CLK クロックの立ち下がりエッジで変化します。
 - NOEH/ NWEH / NEH/ NOEH/NBLH/ アドレス無効出力は、FSMC_CLK クロックの立ち上がりエッジで変化します。

25.5.4 NOR 型フラッシュ/PSRAM コントローラ非同期トランザクション

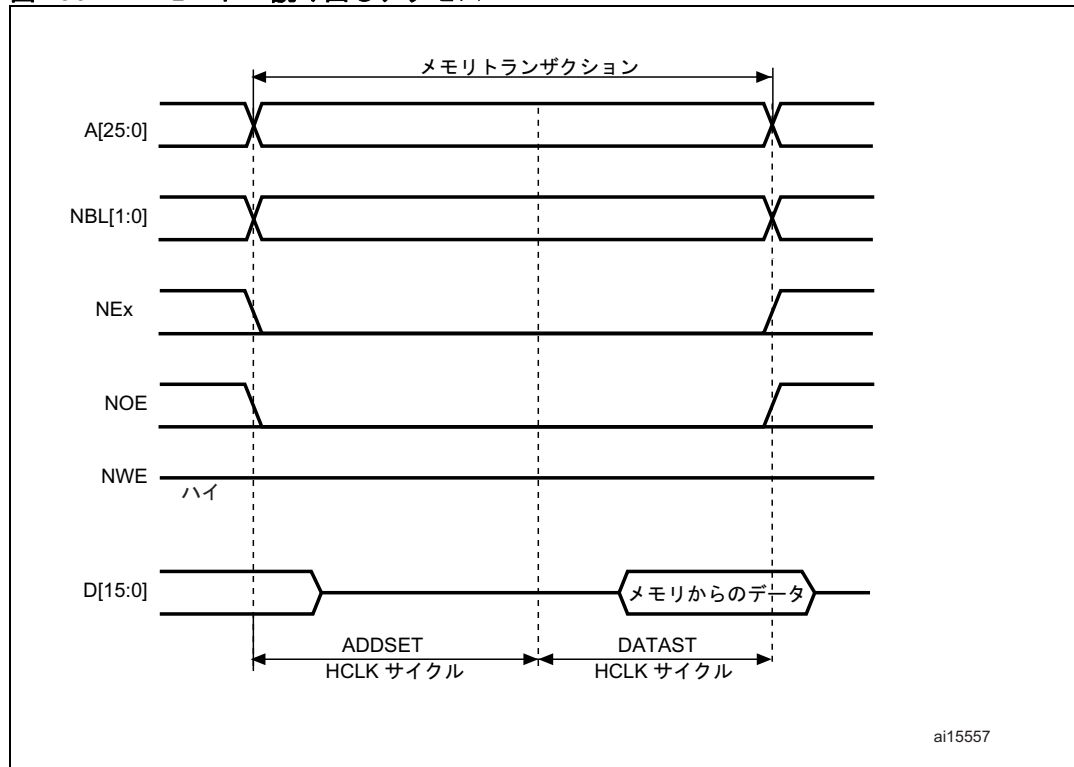
非同期スタティックメモリ (NOR 型フラッシュ、PSRAM、SRAM)

- 信号は、内部クロック HCLK によって同期されます。このクロックはメモリには発行されません。
- FSMC は、チップセレクト信号 NE をディアサートする前に、常にデータをサンプリングします。これにより、メモリデータホールドタイミングに関する制約が満たされます (チップ有効ハイからのデータ遷移は、通常、最小 0 ns です)。
- 拡張モードが有効になっている (FSMC_BCRx レジスタの EXTMOD ビットがセットされている) 場合、最大で 4 つの拡張モード (A、B、C、D) が利用できます。読み出し操作や書き込み操作に A、B、C、D のモードを混合することもできます。例えば、読み出し操作をモード A で実施し、書き込み操作をモード B で実施することができます。
- 拡張モードが無効になっている (FSMC_BCRx レジスタの EXTMOD ビットがリセットされている) 場合、FSMC は次のようにモード 1 またはモード 2 で動作できます。
 - SRAM/PSRAM メモリタイプが選択されている場合 (FSMC_BCRx レジスタで MTYP = 0x0 または 0x01)、モード 1 がデフォルトモードとなります。
 - NOR 型メモリが選択されている場合 (FSMC_BCRx レジスタで MTYP = 0x10)、モード 2 がデフォルトモードとなります。

モード 1 - SRAM/PSRAM (CRAM)

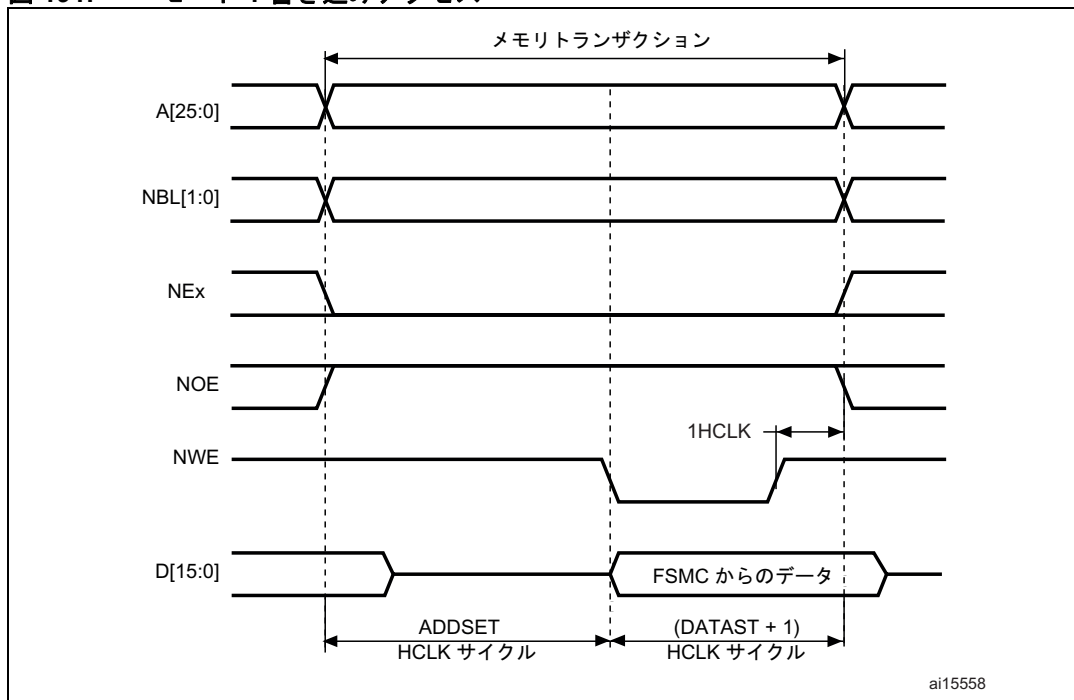
次の図は、サポートされているモードの読み出しおよび書き込みトランザクションと、FSMC_BCRx および FSMC_BTRx/FSMC_BWTRx レジスタの必要設定を示します。

図 190. モード 1 読み出しアクセス



1. 読み出しアクセス中、NBL[1:0] はローになります。

図 191. モード 1 書き込みアクセス



書き込みトランザクションの最後の 1 HCLK サイクルにより、NWE の立ち上がりエッジ後のアドレスおよびデータホールド時間の保証が支援されます。この 1 HCLK サイクルがあるため、DATAST の値はゼロより大きくなければなりません (DATAST>0)。

表 116. FSMC BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x0（非同期モードには影響なし）
18:16	予約済み	0x0
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0（非同期モードには影響なし）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	無視
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて設定しますが、0x2（NOR 型フラッシュ）を除きます。

表 116. FSMC_BCRx ビットフィールド (続き)

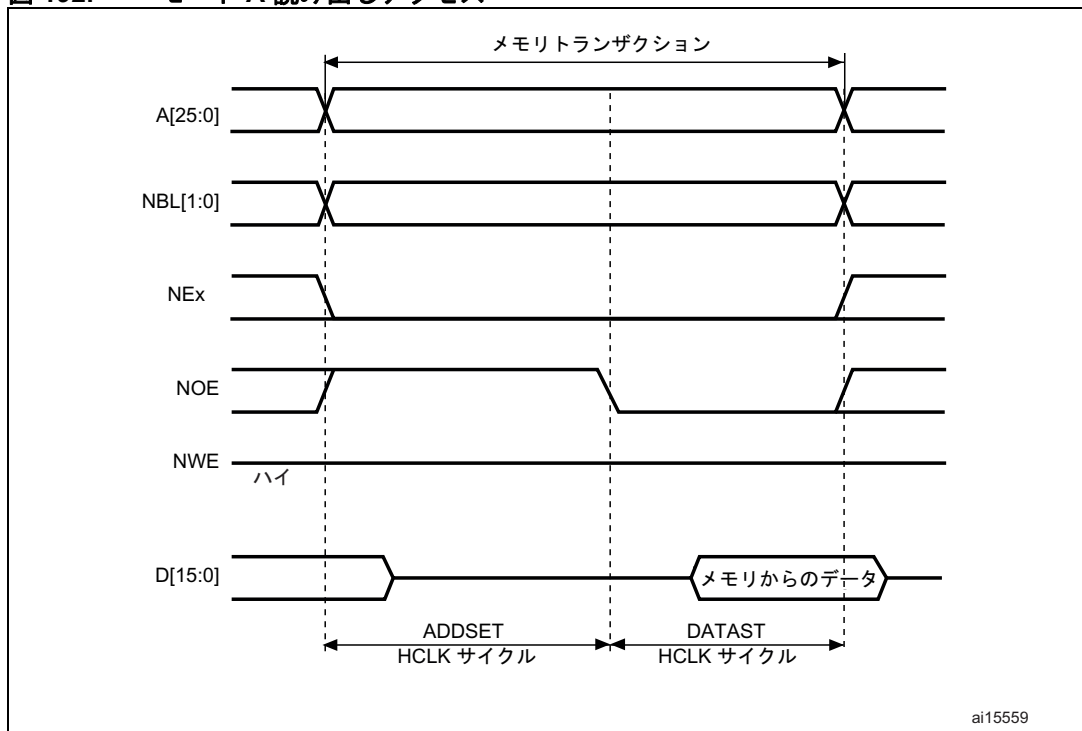
ビット番号	ビット名	設定値
1	MUXE	0x0
0	MBKEN	0x1

表 117. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	無視
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	2 番目のアクセスフェーズの時間 (書き込みアクセスの場合は DATAST+1 HCLK サイクル、読み出しアクセスの場合は DATAST HCLK サイクル)。
7-4	ADDHLD	無視
3-0	ADDSET	最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。ADDSET の最小値は 0 です。

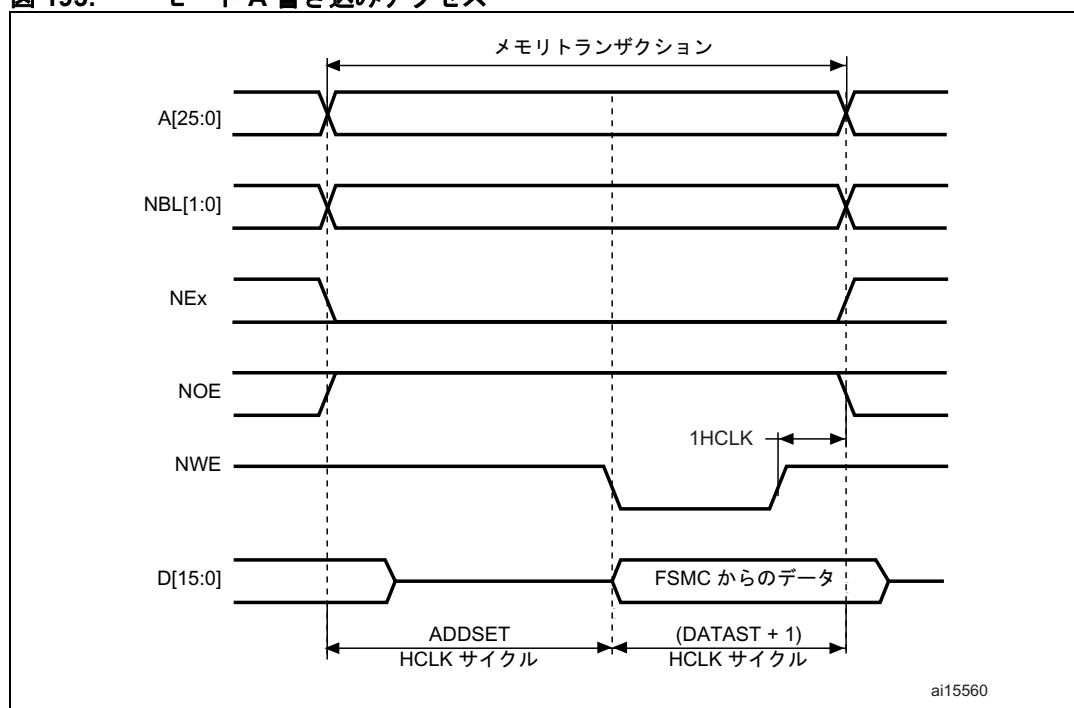
モード A - SRAM/PSRAM (CRAM) OE トグリング

図 192. モード A 読み出しアクセス



1. 読み出しアクセス中、NBL[1:0] はローになります。

図 193. モード A 書き込みアクセス



モード 1 と比較した場合の相違点は、NOE のトグルリングと、独立した読み出しおよび書き込みのタイミングです。

表 118. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x0 (非同期モードには影響なし)
18:16	予約済み	0x0
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0 (非同期モードには影響なし)
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	無視
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて設定しますが、0x2 (NOR 型フラッシュ) を除きます。

表 118. FSMC_BCRx ビットフィールド (続き)

ビット番号	ビット名	設定値
1	MUXEN	0x0
0	MBKEN	0x1

表 119. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 (DATAST HCLK サイクル)。
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。 ADDSET の最小値は 0 です。

表 120. FSMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 (DATAST+1 HCLK サイクル)。
7-4	ADDHLD	無視
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。 ADDSET の最小値は 0 です。

モード 2/B - NOR 型フラッシュ

図 194. モード 2 およびモード B 読み出しアクセス

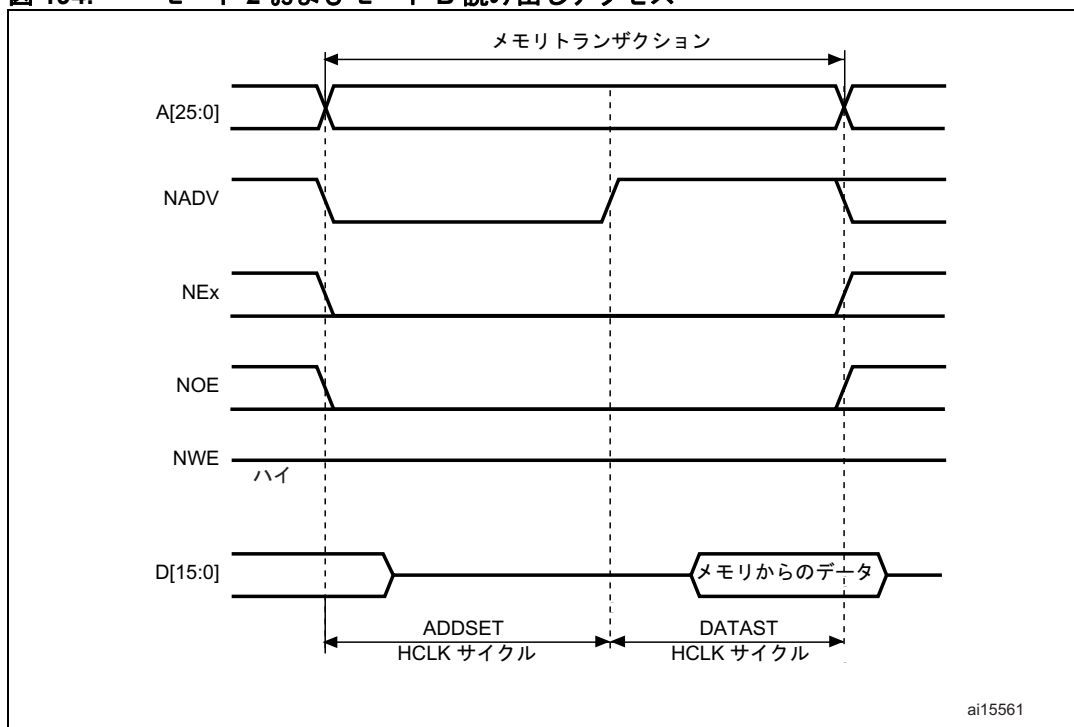


図 195. モード 2 書き込みアクセス

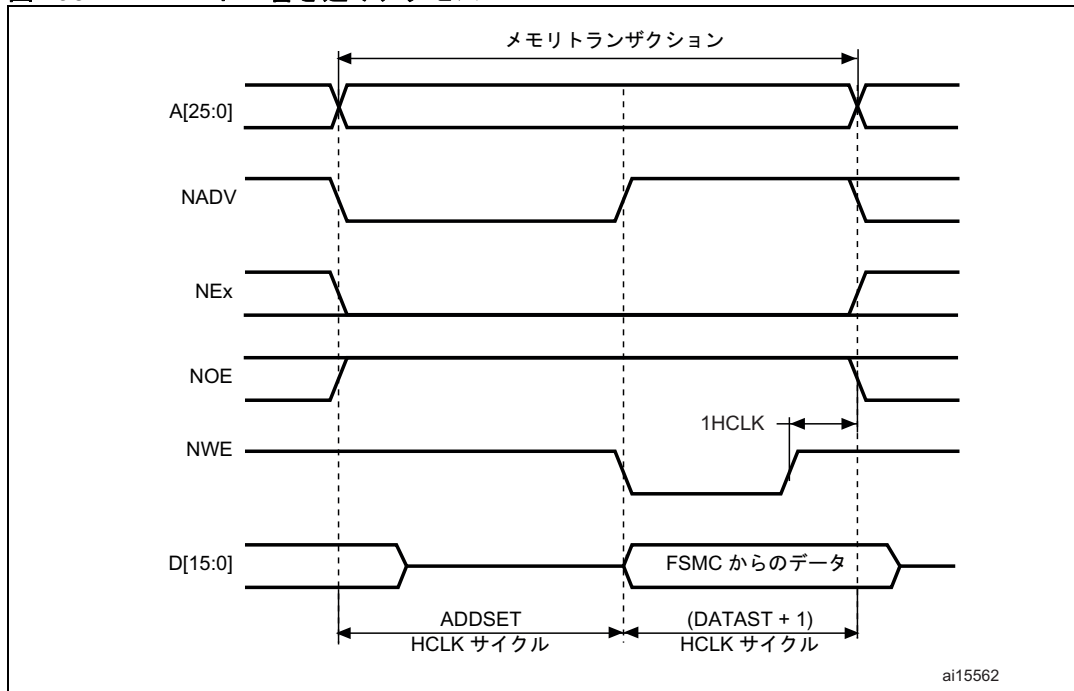
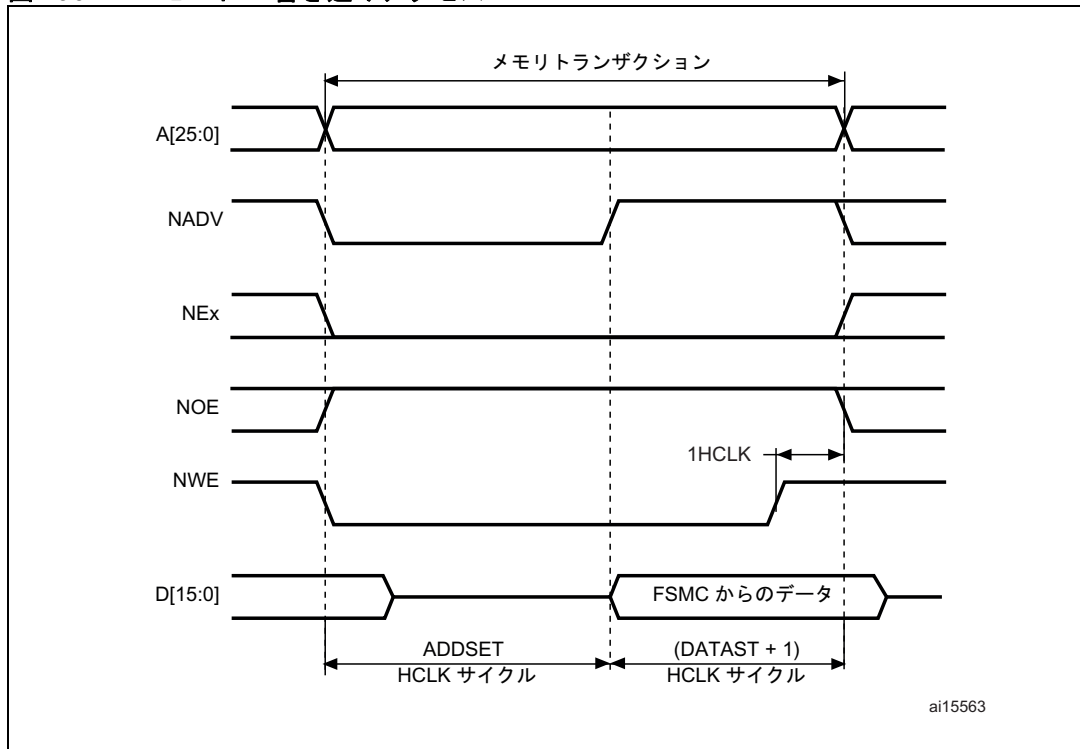


図 196. モード B 書き込みアクセス



モード 1 と比較した場合の相違点は、NWE のトグルリングと、拡張モードがセットされたとき（モード B）の独立した読み出しおよび書き込みのタイミングです。

表 121. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x0（非同期モードには影響なし）
18:16	予約済み	0x0
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	モード B では 0x1、モード 2 では 0x0
13	WAITEN	0x0（非同期モードには影響なし）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x2（NOR 型フラッシュメモリ）

表 121. FSMC_BCRx ビットフィールド (続き)

ビット番号	ビット名	設定値
1	MUXEN	0x0
0	MBKEN	0x1

表 122. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x1
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 (DATAST HCLK サイクル)。
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。 ADDSET の最小値は 0 です。

表 123. FSMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x1
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 (DATAST+1 HCLK サイクル)。
7-4	ADDHLD	無視
3-0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。 ADDSET の最小値は 0 です。

注: FSMC_BWTRx レジスタは、拡張モードがセットされている場合 (モード B) のみ有効です。それ以外の場合には、その内容は意味を持ちません。

モード C - NOR 型フラッシュ - OE トグリング

図 197. モード C 読み出しアクセス

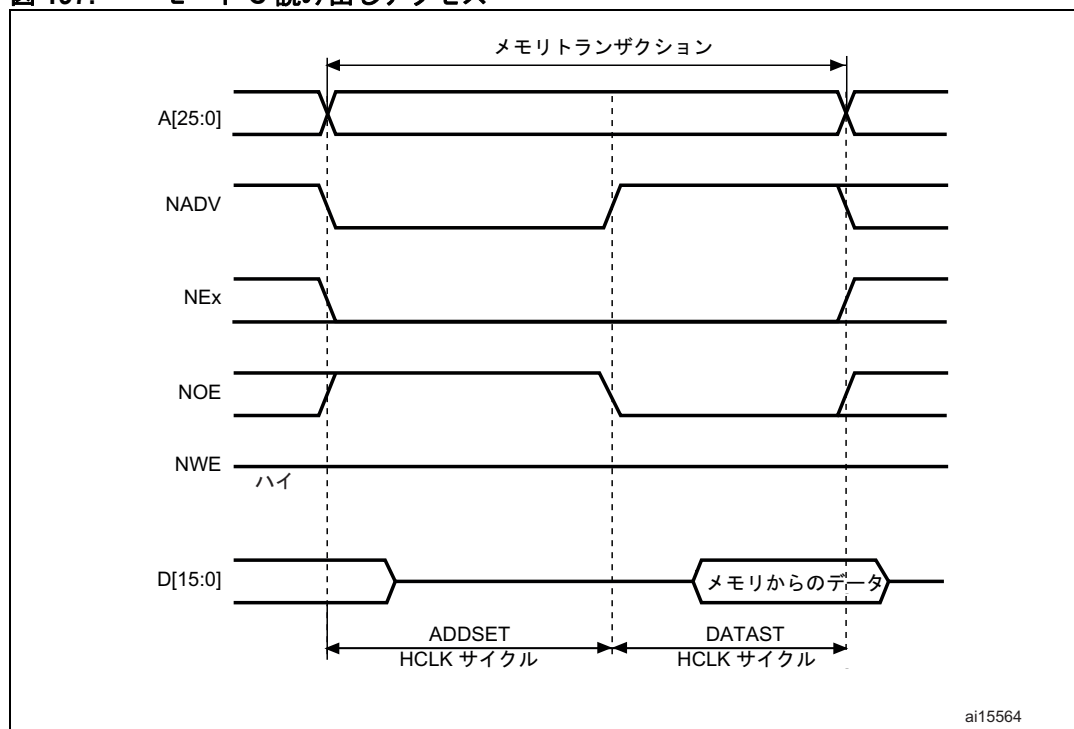
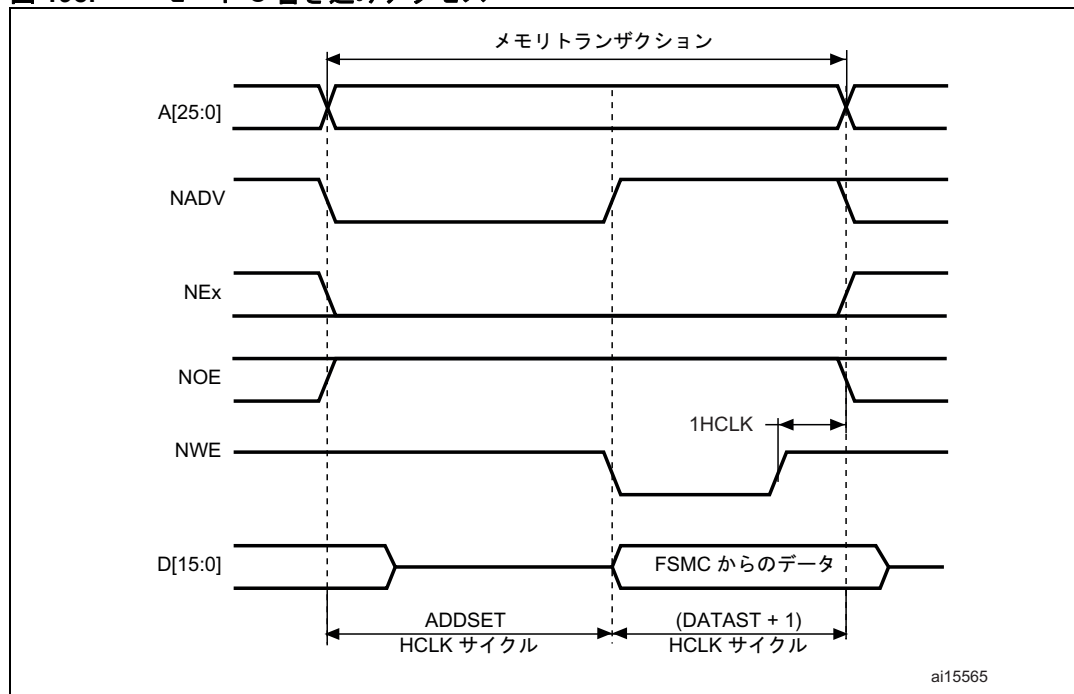


図 198. モード C 書き込みアクセス



モード 1 と比較した場合の相違点は、NOE のトグリングと、独立した読み出しおよび書き込みのタイミングです。

表 124. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x0（非同期モードには影響なし）
18:16	予約済み	0x0
15	ASYNCWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0（非同期モードには影響なし）
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x2（NOR 型フラッシュメモリ）
1	MUXEN	0x0
0	MBKEN	0x1

表 125. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x2
27-24	DATLAT	0x0
23-20	CLKDIV	0x0
19-16	BUSTURN	NEx ハイから NEx ローまでの時間（BUSTURN HCLK）
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間（DATAST HCLK サイクル）。
7-4	ADDHLD	無視
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間（ADDSET HCLK サイクル）。 ADDSET の最小値は 0 です。

表 126. FSMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x2
27:24	DATLAT	無視
23:20	CLKDIV	無視
19:16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15:8	DATAST	書き込みアクセスの 2 番目のアクセスフェーズの時間 (DATAST+1 HCLK サイクル)、。
7:4	ADDHLD	無視
3:0	ADDSET	書き込みアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。 ADDSET の最小値は 0 です。

モード D - 拡張アドレスによる非同期アクセス

図 199. モード D 読み出しアクセス

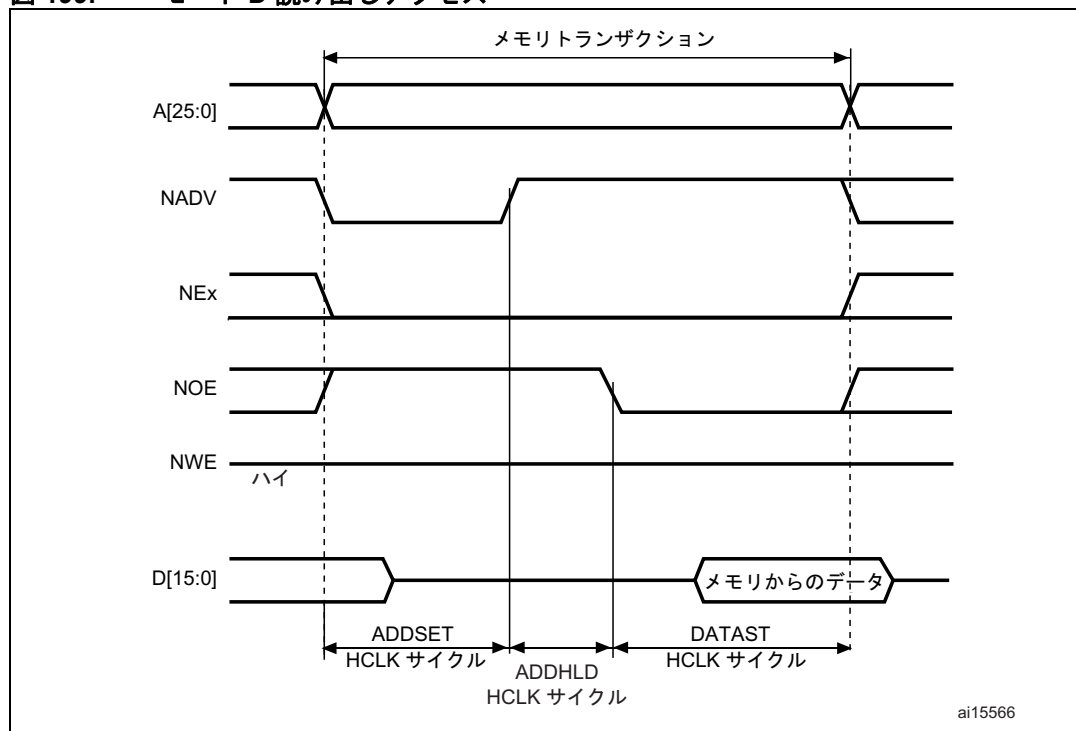
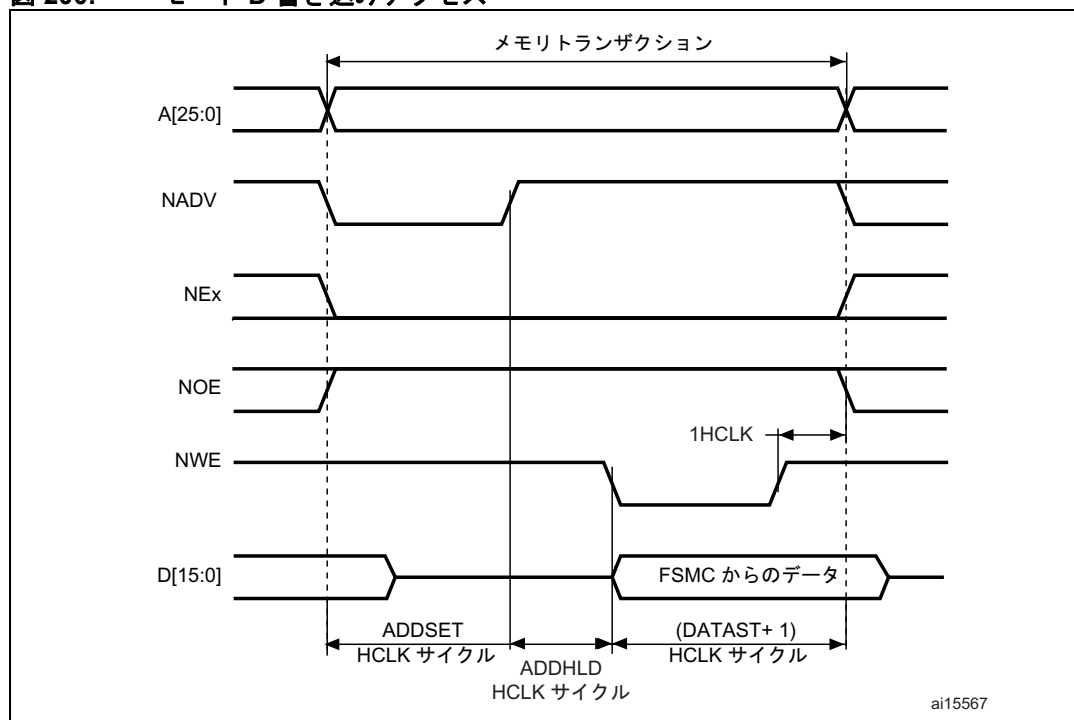


図 200. モード D 書き込みアクセス



モード 1 と比較した場合の相違点は、NADV が変化した後にはトグルする NOE のトグルおよび独立した読み込み書き込みタイミングです。

表 127. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x0 (非同期モードには影響なし)
18:16	予約済み	0x0
15	ASYNCAWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x1
13	WAITEN	0x0 (非同期モードには影響なし)
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	必要に応じて設定します。

表 127. FSMC_BCRx ビットフィールド (続き)

ビット番号	ビット名	設定値
1	MUXEN	0x0
0	MBKEN	0x1

表 128. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x3
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	読み出しアクセスの 2 番目のアクセスフェーズの時間 (DATAST HCLK サイクル)。
7-4	ADDHLD	読み出しアクセスの中間フェーズの時間 (ADDHLD HCLK サイクル)。
3-0	ADDSET	読み出しアクセスの最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。ADDSET の最小値は 1 です。

表 129. FSMC_BWTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x3
27-24	DATLAT	0x0
23-20	CLKDIV	0x0
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	2 番目のアクセスフェーズの時間
7-4	ADDHLD	書き込みアクセスの中間フェーズの時間 (ADDHLD HCLK サイクル)
3-0	ADDSET	最初のアクセスフェーズの時間ADDSET の最小値は 1 です。

マルチプレクスモード - NOR 型フラッシュメモリへのマルチプレクス非同期アクセス

図 201. マルチプレクス読み出しアクセス

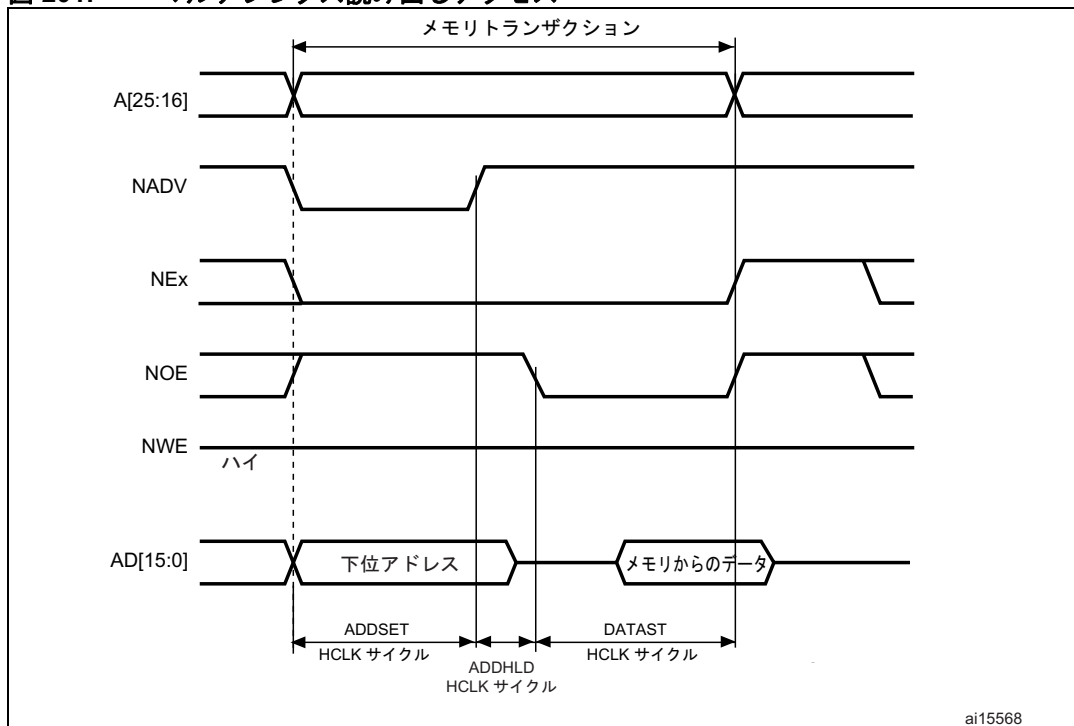
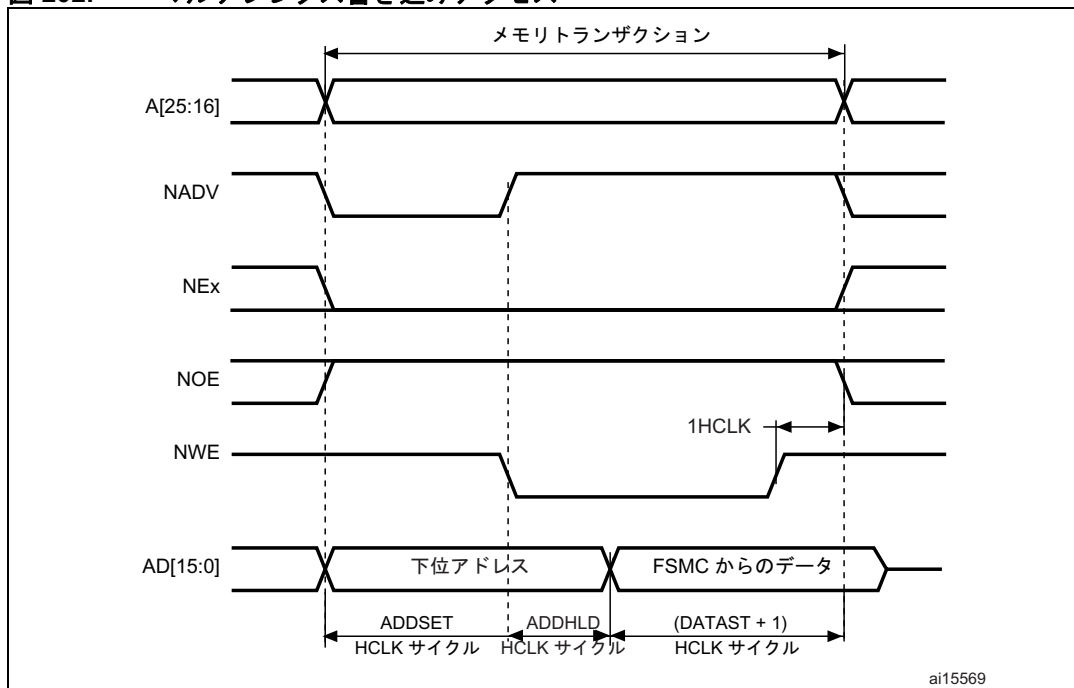


図 202. マルチプレクス書き込みアクセス



モード D との相違点は、データバス上の下位アドレスバイトのドライブです。

表 130. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-21	予約済み	0x000
19	CBURSTRW	0x0 (非同期モードには影響なし)
18:16	予約済み	0x0
15	ASYNCAWAIT	メモリがこの機能をサポートする場合は 1 にセットします。そうでない場合は、0 のままにしておきます。
14	EXTMOD	0x0
13	WAITEN	0x0 (非同期モードには影響なし)
12	WREN	必要に応じて設定します。
11	WAITCFG	無視
10	WRAPMOD	0x0
9	WAITPOL	ビット 15 が 1 の場合のみ、意味を持ちます。
8	BURSTEN	0x0
7	予約済み	0x1
6	FACCEN	0x1
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x2 (NOR 型フラッシュメモリ)
1	MUXEN	0x1
0	MBKEN	0x1

表 131. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29-28	ACCMOD	0x0
27-24	DATLAT	無視
23-20	CLKDIV	無視
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	2 番目のアクセスフェーズの時間 (読み出しアクセスの場合は DATAST HCLK サイクル、書き込みアクセスの場合は DATAST+1 HCLK サイクル)。
7-4	ADDHLD	アクセスの中間フェーズの時間 (ADDHLD HCLK サイクル)。
3-0	ADDSET	最初のアクセスフェーズの時間 (ADDSET HCLK サイクル)。ADDSET の最小値は 1 です。

非同期アクセスにおけるウェイト管理

非同期メモリが、データの受け取りまたは提供の準備がまだできていないことを示すために WAIT 信号をアサートする場合、FSMC_BCRx レジスタの ASYNCWAIT ビットをセットする必要があります。

WAIT 信号がアクティブな場合 (WAITPOL ビットに依存してハイまたはロー)、DATAST ビットによってプログラムされる 2 番目のアクセスフェーズ (データセットアップフェーズ) は、WAIT ビットがインアクティブになるまで延長されます。データセットアップフェーズとは異なり、ADDSET および ADDHLD ビットによってプログラムされる最初のアクセスフェーズ (アドレスセットアップおよびアドレスホールドフェーズ) は、WAIT に対応しないため延長されません。

データセットアップフェーズ (FSMC_BTRx レジスタの DATAST) は、メモリトランザクションが終了する 4 HCLK サイクル前に WAIT が検出されるようにプログラムする必要があります。次のようなケースを考慮する必要があります。

1. メモリが、次のようにトグルする NOE/NWE に合わせた WAIT 信号をアサートする場合。

$$\text{DATAST} \geq (4 \times \text{HCLK}) + \text{max\_wait\_assertion\_time}$$

2. メモリが、NEx (またはトグルしない NOE/NWE) に合わせた WAIT 信号をアサートする場合。
次のような場合、

$$\text{max\_wait\_assertion\_time} > \text{address\_phase} + \text{hold\_phase}$$

このようになります。

$$\text{DATAST} \geq (4 \times \text{HCLK}) + (\text{max\_wait\_assertion\_time} - \text{address\_phase} - \text{hold\_phase})$$

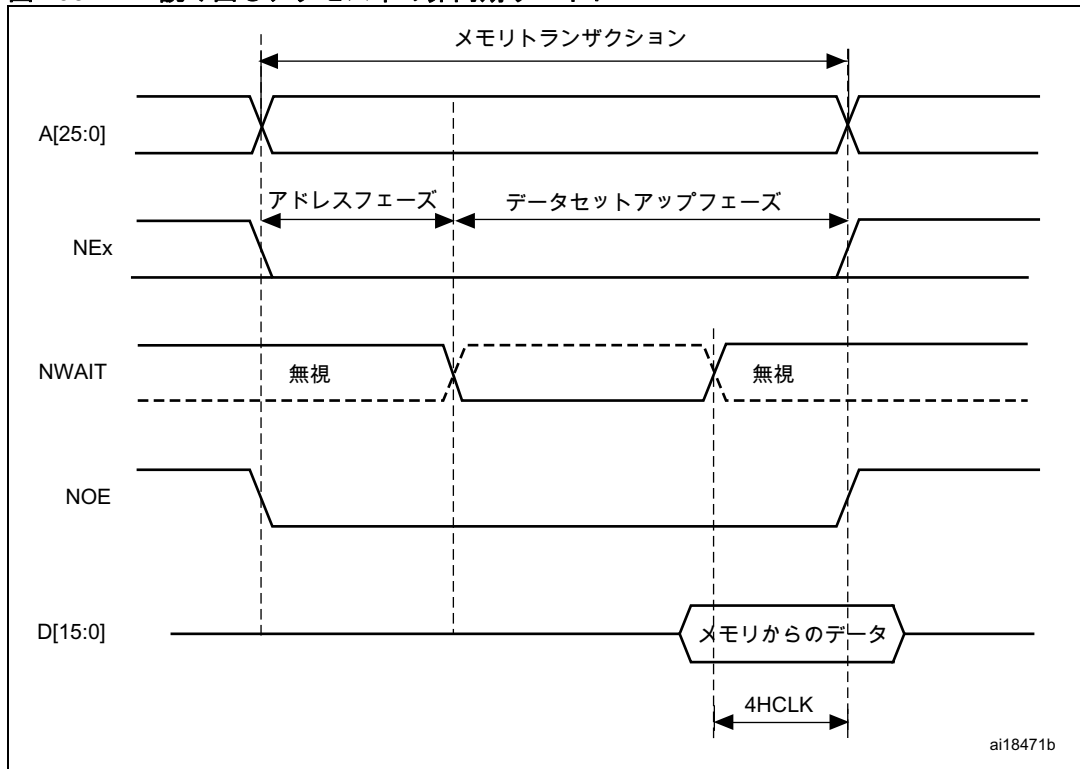
それ以外は次のようになります。

$$\text{DATAST} \geq 4 \times \text{HCLK}$$

ここで、max_wait_assertion_time は、NEx/NOE/NWE がローになったときにメモリが WAIT 信号のアサートに要する最大時間です。

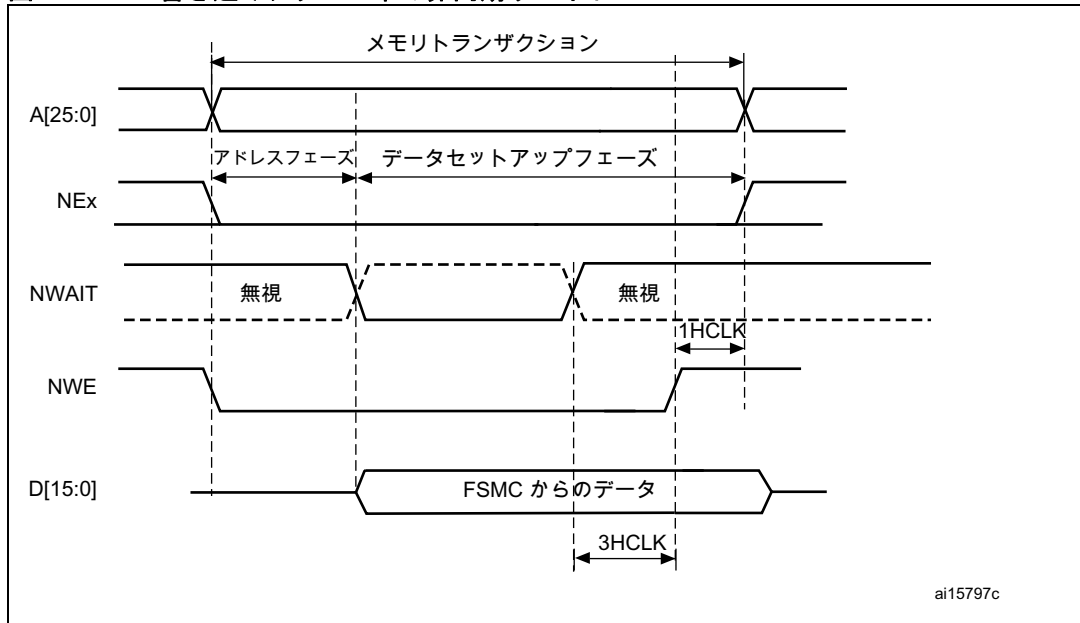
図 203 と 図 204 は、WAIT が解放された後に非同期メモリによってメモリアクセスに追加される HCLK クロックサイクル数を示します (上記のケースとは別)。

図 203. 読み出しアクセス中の非同期ウェイト



1. N WAIT 極性は、FSMC_BCRx レジスタの WAITPOL ビットの設定に依存します。

図 204. 書き込みアクセス中の非同期ウェイト



1. N WAIT 極性は、FSMC_BCRx レジスタの WAITPOL ビットの設定に依存します。

25.5.5 同期トランザクション

メモリクロック (CLK) は、CLKDIV パラメータ値に従った HCLK の約数です。

NOR 型フラッシュメモリは、NADV のアサーションから CLK がハイになるまでの最小時間を指定します。この制約を満たすために、FSMC は、同期アクセスの最初の内部クロックサイクルでは (NADV アサーションの前)、クロックをメモリに供給しません。これにより、メモリクロックの立ち上がりエッジは、必ず NADV ローパルスの途中で発生するようになります。

データ遅延と NOR 型フラッシュ遅延

データ遅延は、データをサンプリングする前のウェイトサイクル数です。DATLAT の値は、NOR 型フラッシュ設定レジスタで指定された遅延の値と一致しなければなりません。FSMC は、NADV がローのときのクロックサイクルをデータ遅延カウントに含めません。

注意： 一部の NOR 型フラッシュメモリは、データ遅延カウントに NADV ローサイクルを含むため、NOR 型フラッシュの遅延と FSMC DATALAT パラメータの厳密な関係は、次のいずれかとなります。

- NOR 型フラッシュ遅延 = (DATLAT + 2) CLK クロックサイクル数
- NOR 型フラッシュ遅延 = (DATLAT + 3) CLK クロックサイクル数

最近のメモリの中には、遅延フェーズにおいて NWAIT をアサートするものがあります。そのような場合には、DATLAT を最小値に設定することができます。結果として、FSMC はデータをサンプリングして、データが有効であるかどうか評価できるだけの時間待つこととなります。このように、FSMC は、メモリが遅延から抜け出し、実際のデータが取得されることを検出します。

その他のメモリは、遅延中に NWAIT をアサートしません。この場合、遅延が FSMC とメモリの両方について正しく設定される必要があります。正しく設定されないと、無効なデータが正しいデータとして取得されたり、有効なデータがメモリアccessの初期フェーズで失われることとなります。

シングルバースト転送

選択されたバンクが同期アクセスのバーストモードで設定されている場合、例えば AHB シングルバーストトランザクションが 16 ビットメモリでリクエストされると、FSMC は長さ 1 (AHB 転送が 16 ビットの場合) または 2 (AHB 転送が 32 ビットの場合) のバーストトランザクションを実行し、最後のデータがストローブされるとチップセレクトをネゲートします。

明らかに、このような転送はサイクル数という点では最も効率的なものではありません (非同期読み出しと比較して)。しかし、ランダム非同期アクセスでは、まずメモリアccessモードを再プログラムする必要があり、すべて合わせるともっと時間がかかることとなります。

ウェイト管理

同期 NOR 型フラッシュメモリの場合、NWAIT はプログラムされた遅延期間である (DATALAT+2) CLK サイクル後に評価されます。

NWAIT がアクティブであることが検出されると (WAITPOL = 0 の場合はローレベル、WAITPOL = 1 の場合はハイレベル)、NWAIT がインアクティブであることが検出されるまで (WAITPOL = 0 の場合はハイレベル、WAITPOL = 1 の場合はローレベル)、ウェイトステートが挿入されます。

NWAIT がインアクティブのときには、データはただちに (ビット WAITCFG = 1) または次のクロックエッジ (ビット WAITCFG = 0) で有効とみなされます。

NWAIT 信号によるウェイトステートの挿入時には、コントローラはメモリへクロックパルスの送信を続け、チップセレクトと出力有効信号を有効に保ち、データを有効と認識しません。

バーストモードでの NOR 型フラッシュの NWAIT 信号には、2 つのタイミング設定があります。

- フラッシュメモリは、ウェイトステートの 1 データサイクル前に NWAIT 信号をアサートします (リセット後のデフォルト)。
- フラッシュメモリは、ウェイトステート中に NWAIT 信号をアサートします。

この 2 つの NOR 型フラッシュウェイトステート設定は、FSMC_BCRx レジスタ (x=0..3) の WAITCFG ビットにより、各チップセレクトごとに FSMC によってサポートされます。

図 205. ウェイト設定

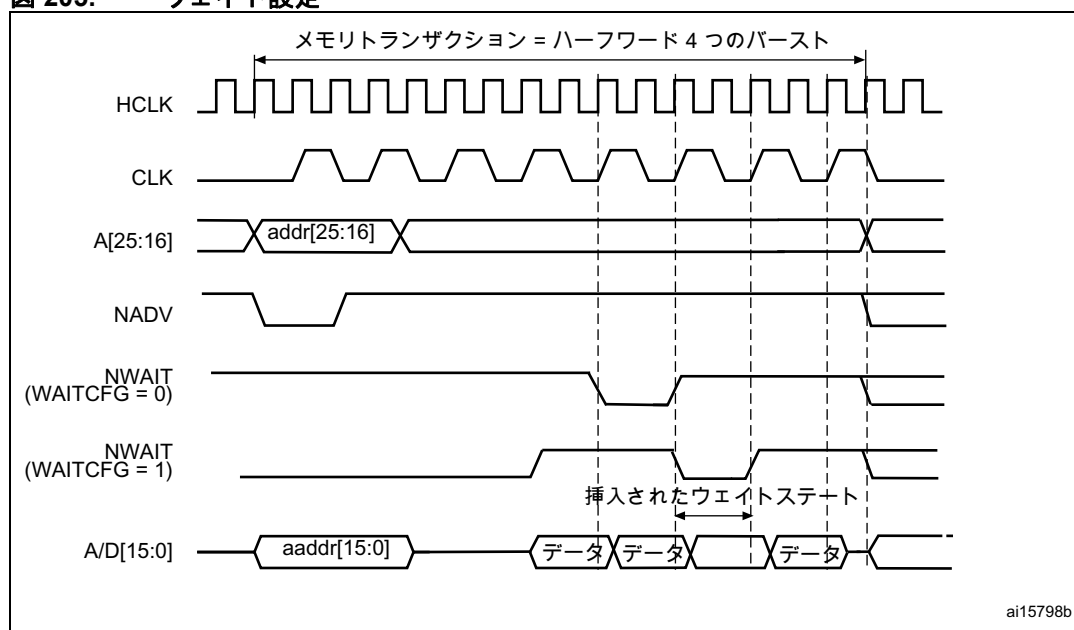
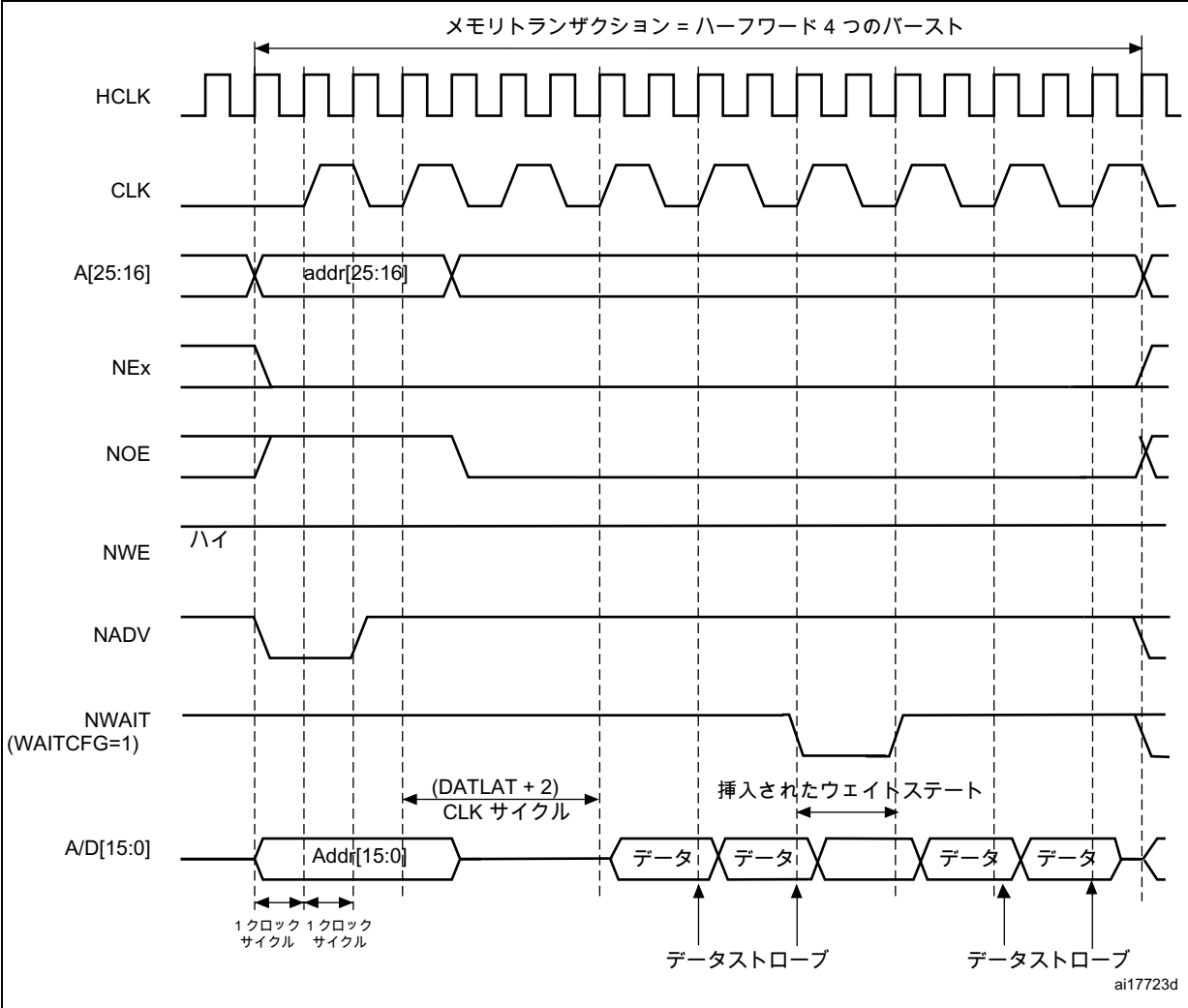


図 206. 同期マルチプレクス読み出しモード-NOR、PSRAM (CRAM)



1. バイトレーン出力 BL は示されていません。NOR アクセス時にはハイに保たれ、PSRAM (CRAM) アクセス時にはローに保たれます。
2. NWAIT 極性は 0 にセットされます。

表 132. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	同期読み出しには影響しません。
18-16	予約済み	0x0
15	ASCYCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリがこの機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	同期読み出しには影響しません。
11	WAITCFG	メモリに応じて設定します。

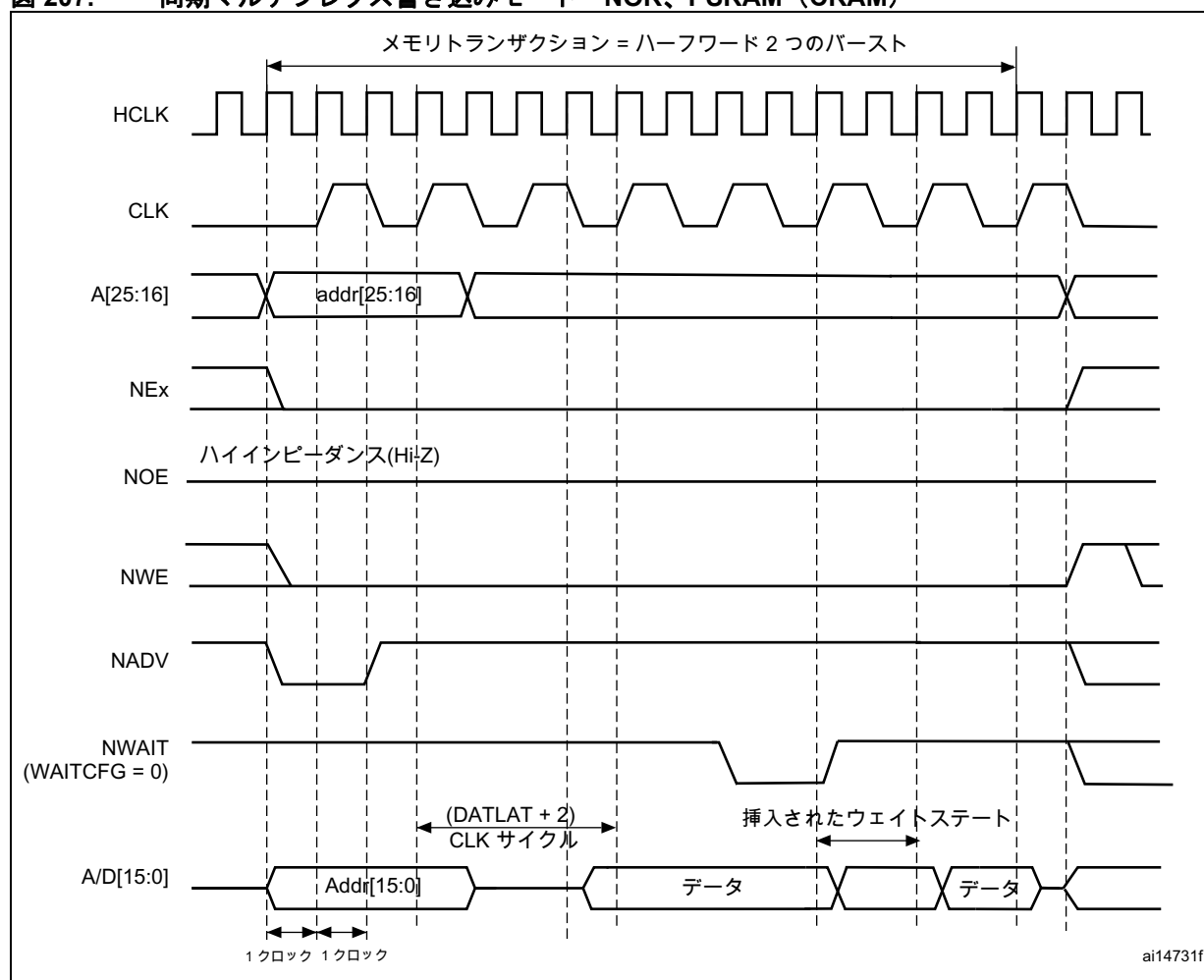
表 132. FSMC_BCRx ビットフィールド (続き)

ビット番号	ビット名	設定値
10	WRAPMOD	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	0x1
7	予約済み	0x1
6	FACCEN	メモリサポート (NOR 型フラッシュメモリ) に従って設定されます。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1 または 0x2
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 133. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
23-20	CLKDIV	0x0 の場合、CLK = HCLK となります (サポートされていません)。 0x1 の場合、CLK = 2×HCLK となります。 ..
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

図 207. 同期マルチプレクス書き込みモード-NOR、PSRAM (CRAM)



1. メモリは、あらかじめ NWAIT 信号を 1 サイクル前に発行しなければならないため、WAITCFG を 0 にプログラムする必要があります。
2. NWAIT 極性は 0 にセットされます。
3. バイトレーン (NBL) 出力は図に示されていません。また、NEx がアクティブの間はローに保たれます。

表 134. FSMC_BCRx ビットフィールド

ビット番号	ビット名	設定値
31-20	予約済み	0x000
19	CBURSTRW	0x1
18-16	予約済み	0x0
15	ASCYCWAIT	0x0
14	EXTMOD	0x0
13	WAITEN	メモリが機能をサポートする場合は 1 にセット、そうでなければ 0 のままとします。
12	WREN	0x1
11	WAITCFG	0x0

表 134. FSMC_BCRx ビットフィールド (続き)

ビット番号	ビット名	設定値
10	WRAPMOD	0x0
9	WAITPOL	メモリに応じて設定します。
8	BURSTEN	同期読み出し時は影響しません。
7	予約済み	0x1
6	FACCEN	メモリのサポート状況に応じて設定します。
5-4	MWID	必要に応じて設定します。
3-2	MTYP	0x1
1	MUXEN	必要に応じて設定します。
0	MBKEN	0x1

表 135. FSMC_BTRx ビットフィールド

ビット番号	ビット名	設定値
31:30	予約済み	0x0
29:28	ACCMOD	0x0
27-24	DATLAT	データ遅延
23-20	CLKDIV	0x0 の場合、CLK = HCLK となります (サポートされていません)。 0x1 の場合、CLK = 2×HCLK となります。
19-16	BUSTURN	NEx ハイから NEx ローまでの時間 (BUSTURN HCLK)
15-8	DATAST	無視
7-4	ADDHLD	無視
3-0	ADDSET	無視

25.5.6 NOR/PSRAM 制御レジスタ

NOR/PSRAM 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。

SRAM/NOR 型フラッシュチップセレクト制御レジスタ 1.4 (FSMC_BCR1.4)

アドレスオフセット：0xA000 0000 + 8 * (x - 1)、x = 1..4

リセット値：0x0000 30DB（バンク 1）、0x0000 30D2（バンク 2 から 4）

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。

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		
予約済み												CBURSTRW	予約済み				ASCYCWAIT	EXTMOD	WAITEN	WREN	WAITCFG	WRAPMOD	WAITPOL	BURSTEN	予約済み	FACCEN	MWID			MTYP		MUXEN	MBKEN
																								RW				RW	RW	RW	RW	RW	RW

ビット 31:20 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 19 **CBURSTRW** : 書き込みバーストイネーブル

セルラー RAM (PSRAM) メモリでは、このビットが書き込み操作における同期バーストプロトコルを有効にします。同期読み出しアクセスのイネーブルビットは、FSMC_BCRx レジスタの BURSTEN ビットです。

0 : 書き込み操作は、常に非同期モードで実行されます。

1 : 書き込み操作は、同期モードで実行されます。

ビット 18:16 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 **ASYNCAWAIT** : 非同期転送時のウェイト信号

このビットは、非同期プロトコル時でも FSMC によるウェイト信号の使用を有効／無効にします。

0 : 非同期プロトコル実行中（リセット後のデフォルト）に NWAIT 信号を考慮しません。

1 : 非同期プロトコル実行中に NWAIT 信号を考慮します。

ビット 14 **EXTMOD** : 拡張モードイネーブル

このビットでは、FSMC が FSMC_BWTR レジスタ内部の同期アクセスの書き込みタイミングをプログラムできるため、読み出しと書き込みのタイミングが異なります。

0 : FSMC_BWTR レジスタ内部の値は考慮されません（リセット後のデフォルト）。

1 : FSMC_BWTR レジスタ内部の値が考慮されます。

注： 拡張モードが無効になっている場合、FSMC は次のようにモード 1 またはモード 2 で動作できます。

- SRAM/PSRAM 型メモリタイプが選択されている場合 (MTYP = 0x0 または 0x01)、モード 1 がデフォルトモードとなります。
- NOR 型メモリタイプが選択されている場合 (MTYP = 0x10)、モード 2 がデフォルトモードとなります。

ビット 13 **WAITEN** : ウェイトイネーブルビット

このビットは、同期モードでフラッシュメモリにアクセスする場合に NWAIT 信号によるウェイトステートの挿入を有効／無効にします。

0 : NWAIT 信号は無効です（レベルは考慮されず、プログラムされたフラッシュ遅延時間後にウェイトステートは挿入されません）。

1 : NWAIT 信号は有効です（アサートされた場合、プログラムされたフラッシュ遅延時間後にレベルが考慮され、ウェイトステートが挿入されます）（リセット後のデフォルト）。



ビット 12 WREN : 書き込みイネーブルビット

このビットは、FSMC によるバンクへの書き込み操作の有効/無効を示します。

0 : FSMC によるバンクへの書き込み操作は無効であり、AHB エラーが報告されます。

1 : FSMC によるバンクへの書き込み操作は有効です (リセット後のデフォルト)。

ビット 11 WAITCFG : ウェイトタイミング設定。

NWAIT 信号は、メモリからのデータが有効であるかどうか、または同期モードでフラッシュメモリにアクセスする際にウェイトステートを挿入する必要があるかどうかを示します。この設定ビットは、ウェイトステートの 1 クロック前、またはウェイトステート中に、メモリによって N WAIT がアサートされるかどうかを定義します。

0 : N WAIT 信号は、ウェイトステートの 1 データサイクル前にアクティブです (リセット後のデフォルト)。

1 : N WAIT 信号は、ウェイトステート中、アクティブです (PRAM には使用しない)。

ビット 10 WRAPMOD : ラップトバーストモードサポート

コントローラが AHB バーストラップアクセスを 2 つのリニアアクセスに分割するかどうかを定義します。バーストモードでメモリにアクセスするときのみ有効です。

0 : ダイレクトラップトバーストは無効です (リセット後のデフォルト)。

1 : ダイレクトラップトバーストは有効です。

注: このビットは、CPU および DMA がラッピングバースト転送を生成できないため無効です。

ビット 9 WAITPOL : ウェイト信号極性ビット。

メモリからのウェイト信号の極性を定義します。バーストモードでメモリにアクセスするときのみ有効です。

0 : N WAIT アクティブローです (リセット後のデフォルト)。

1 : N WAIT アクティブハイです。

ビット 8 BURSTEN : バーストイネーブルビット

このビットは、読み出し操作中の同期アクセスを有効/無効にします。バーストモードで動作する同期メモリに対してのみ有効です。

0 : バーストモードは無効です (リセット後のデフォルト)。読み出しアクセスは、非同期モードで実行されます。

1 : バーストモードは有効です。読み出しアクセスは、同期モードで実行されます。

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 FACCEN : フラッシュアクセスイネーブル**

NOR 型フラッシュメモリアクセス操作を有効にします。

0 : 対応する NOR 型フラッシュメモリアクセスは無効です。

1 : 対応する NOR 型フラッシュメモリアクセスは有効です (リセット後のデフォルト)。

ビット 5:4 MWID : メモリデータバス幅

外部メモリデバイスの幅を定義します。すべてのタイプのメモリに対して有効です。

00 : 8ビット

01 : 16ビット (リセット後のデフォルト)

10 : 予約済み。使用しないでください。

11 : 予約済み。使用しないでください。

ビット 3:2 MTYP : メモリタイプ

対応するメモリバンクに接続される外部メモリのタイプを定義します。

00 : SRAM (バンク 2...4 のリセット後のデフォルト)

01 : PSRAM (CRAM)

10 : NOR 型フラッシュ/OneNAND 型フラッシュ (バンク 1 のリセット後のデフォルト)

11 : 予約済み

ビット 1 MUXEN : アドレス/データマルチプレクスイネーブルビット

このビットがセットされているときには、データバス上でアドレスとデータの値がマルチプレクスされます。NOR および PSRAM 型メモリでのみ有効です。

0 : アドレス/データはマルチプレクスされません。

1 : アドレス/データはデータバス上でマルチプレクスされます (リセット後のデフォルト)。

ビット 0 MBKEN : メモリバンクイネーブルビット

メモリバンクを有効にします。リセット後は、バンク 1 は有効であり、その他はすべて無効です。無効なバンクにアクセスすると、AHB バス上で ERROR が発生します。

0 : 対応するメモリバンクは無効です。

1 : 対応するメモリバンクは有効です。

SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1..4 (FSMC_BTR1..4)

アドレスオフセット : $0xA000\ 0000 + 0x04 + 8 * (x - 1)$ 、 $x = 1..4$

リセット値 : $0x0FFF\ FFFF$

このレジスタには、SRAM、PSRAM、NOR 型フラッシュメモリに使用される各メモリバンクの制御情報が含まれています。FSMC_BCRx レジスタの EXTMOD ビットがセットされている場合、このレジスタは書き込みと読み出しアクセスに分割されます。すなわち、読み出しアクセスを設定するレジスタ（このレジスタ）と、書き込みアクセスを設定するレジスタ（FSMC_BWTRx レジスタ）の 2 つのレジスタが使用できます。

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
予約済み		ACCMOD		DATLAT				CLKDIV				BUSTURN				DATAST								ADDHLD				ADDSET			
		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	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 **ACCMOD** : アクセスモード

タイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FSMC_BCRx レジスタの EXTMOD ビットが 1 のときのみ考慮されます。

00 : アクセスモード A

01 : アクセスモード B

10 : アクセスモード C

11 : アクセスモード D

ビット 27:24 **DATLAT** : 同期 NOR 型フラッシュメモリのデータ遅延（ビット記述テーブルの下にある注記を参照）

バーストモードが有効な同期 NOR 型フラッシュメモリについて、最初のデータを読み書きする前にメモリに発行するメモリクロックサイクル数（+2）を定義します。

このタイミングパラメータは、HCLK 周期ではなく、FSMC_CLK 周期で表されます。PSRAM（CRAM）の場合には、このフィールドを 0 に設定する必要があります。非同期 NOR 型フラッシュまたは SRAM あるいは PSRAM では、この値は無関係です。

0000 : 最初のバーストアクセスでは、2 CLK クロックサイクルのデータ遅延

1111 : 最初のバーストアクセスでは、17 CLK クロックサイクルのデータ遅延（リセット後のデフォルト）

ビット 23:20 **CLKDIV** : クロック分周比（FSMC_CLK 信号）

FSMC_CLK クロック出力信号の周期を HCLK サイクル数で定義します。

0000 : 予約済み

0001 : FSMC_CLK 周期 = 2 x HCLK 周期

0010 : FSMC_CLK 周期 = 3 x HCLK 周期

1111 : FSMC_CLK 周期 = 16 x HCLK 周期（リセット後のデフォルト）

非同期 NOR 型フラッシュ、SRAM、または PSRAM アクセスでは、この値は無関係です。

ビット 19:16 **BUSTURN** : バスターンアラウンドフェーズ時間

これらのビットはソフトウェアによって書き込まれ、書き込み/読み出しトランザクションの最後に遅延を追加します。この遅延によって連続トランザクション間の最小時間（NEx ハイから NEx ローになるまでの t_{EHL} ）と読み出しアクセス後にメモリがデータバスを解放するのに必要な最大時間（ t_{EHQZ} ）を次のように一致させることができます

EXTMOD = "0" の場合、 $(BUSTURN + 1) \text{ HCLK 周期} \geq t_{EHLmin}$ かつ $(BUSTURN + 2) \text{ HCLK 周期} \geq t_{EHQZmax}$

EXTMOD = "1" の場合、 $(BUSTURN + 2) \text{ HCLK 周期} \geq \max(t_{EHLmin}, t_{EHQZmax})$

0000 : BUSTURN フェーズ時間 = 0 HCLK クロックサイクルを追加します。

...

1111 : BUSTURN フェーズ時間 = 15 x HCLK クロックサイクル（リセット後のデフォルト）

ビット 15:8 **DATAST** : データフェーズ時間

これらのビットはソフトウェアによって書き込まれ、非同期アクセスで使用するデータフェーズの時間を定義します (図 190~図 202 を参照)。

0000 0000 : 予約済み

0000 0001 : DATAST フェーズ時間 = 1 x HCLK クロックサイクル

0000 0010 : DATAST フェーズ時間 = 2 x HCLK クロックサイクル

...

1111 1111 : DATAST フェーズ時間 = 255 x HCLK クロックサイクル (リセット後のデフォルト)

各メモリタイプおよびアクセスモードでのデータフェーズ時間については、それぞれの図 (図 190~図 202) を参照してください。

例: モード 1、書き込みアクセス、DATAST = 1 : データフェーズ時間 = DATAST+1 = 2 HCLK クロックサイクル。

注: 同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD** : アドレスホールドフェーズ時間

これらのビットはソフトウェアによって書き込まれ、モード D およびマルチプレクスアクセスで使用するアドレスホールドフェーズの時間を定義します (図 199~図 202 を参照)。

0000 : 予約済み

0001 : ADDHLD フェーズ時間 = 1 x HCLK クロックサイクル

0010 : ADDHLD フェーズ時間 = 2 x HCLK クロックサイクル

...

1111 : ADDHLD フェーズ時間 = 15 x HCLK クロックサイクル (リセット後のデフォルト)

各アクセスモードでのアドレスホールドフェーズ時間については、それぞれの図 (図 199~図 202) を参照してください。

注: 同期アクセスではこの値は使用されず、アドレスホールドフェーズの長さは、常に1メモリクロック周期です。

ビット 3:0 **ADDSET** : アドレスセットアップフェーズ時間

これらのビットはソフトウェアによって書き込まれ、SRAM、ROM、非同期 NOR 型フラッシュアクセスで使用するアドレスセットアップフェーズの時間を定義します (図 190 から 図 202 を参照)。

0000 : ADDSET フェーズ時間 = 0 x HCLK クロックサイクル

...

1111 : ADDSET フェーズ時間 = 15 x HCLK クロックサイクル (リセット後のデフォルト)

各アクセスモードでのアドレスセットアップフェーズ時間については、それぞれの図 (図 190 から 図 202) を参照してください。

注: 同期アクセスでは、この値は無視されます。

注: 内部更新により、PSRAM (CRAM) には可変遅延があります。したがって、これらのメモリは、遅延フェーズ全体で NWAIT 信号を発行して必要に応じて遅延を延長します。

PSRAM (CRAM) では、FSMC がすぐに遅延フェーズを終了し、メモリからの NWAIT 信号のサンプリングを開始して、メモリがレディ状態になったら読み出しまたは書き込みを開始するよう、DATLAT フィールドを 0 に設定する必要があります。

過去のフラッシュメモリとは異なり、この方法は、NWAIT 信号を発行する最新世代の同期フラッシュメモリでも使用できます (使用中の特定のフラッシュメモリのデータシートを参照してください)。

SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1..4 (FSMC_BWTR1..4)

アドレスオフセット : $0xA000\ 0000 + 0x104 + 8 * (x - 1)$ 、 $x = 1..4$

リセット値 : $0x0FFF\ FFFF$

このレジスタは、SRAM、PSRAM、および NOR 型フラッシュメモリで使用される各メモリバンクの制御情報を含みます。このレジスタは、FSMC_BCRx レジスタの EXTMOD ビットがセットされているときのみ、書き込み非同期アクセスに対してアクティブになります。

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
予約済み	ACCMO D		予約済み									BUSTURN				DATAST								ADDHLD				ADDSET			
	rw	rw										rw	rw	rw	rw	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 **ACCMOD** : アクセスモード。

次のタイミング図に示されているように、非同期アクセスモードを指定します。これらのビットは、FSMC_BCRx レジスタの EXTMOD ビットが 1 にセットされている場合にのみ考慮されます。

00 : アクセスモード A

01 : アクセスモード B

10 : アクセスモード C

11 : アクセスモード D

ビット 27:20 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 19:16 **BUSTURN** : バスターンアラウンドフェーズ時間

これらのビットはソフトウェアで書き込まれ、書き込みトランザクションの終わりに遅延を追加し、次のように連続トランザクション間の最小時間 (ENx ハイから ENx ローまでの t_{EHL}) を一致させます。

$(BUSTURN + 1) \text{ HCLK 周期} \geq t_{EHLmin}$

0000 : BUSTURN フェーズ時間 = 0 HCLK クロックサイクルを追加します。

...

1111 : BUSTURN フェーズ時間 = 15 HCLK クロックサイクル (リセット後のデフォルト)

ビット 15:8 **DATAST** : データフェーズ時間

これらのビットはソフトウェアによって書き込まれ、非同期 SRAM、PSRAM、NOR 型フラッシュメモリアクセスで使用するデータフェーズの時間を定義します (図 190~図 202 を参照)。

0000 0000 : 予約済み

0000 0001 : DATAST フェーズ時間 = 1 x HCLK クロックサイクル

0000 0010 : DATAST フェーズ時間 = 2 x HCLK クロックサイクル

...

1111 1111 : DATAST フェーズ時間 = 255 x HCLK クロックサイクル (リセット後のデフォルト)

注: 同期アクセスでは、この値は無視されます。

ビット 7:4 **ADDHLD** : アドレスホールドフェーズ時間

これらのビットはソフトウェアによって書き込まれ、非同期マルチプレクスアクセスで使用するアドレスホールドフェーズの時間を定義します (図 199 から 図 202 を参照)。

0000 : 予約済み

0001 : ADDHLD フェーズ時間 = 1 x HCLK クロックサイクル

0010 : ADDHLD フェーズ時間 = 2 x HCLK クロックサイクル

...

1111 : ADDHLD フェーズ時間 = 15 x HCLK クロックサイクル (リセット後のデフォルト)

注: 同期 NOR 型フラッシュアクセスでは、この値は使用されず、アドレスホールドフェーズの長さは、常に 1 フラッシュクロック周期です。

ビット 3:0 **ADDSET** : アドレスセットアップフェーズ時間

これらのビットはソフトウェアによって書き込まれ、非同期アクセスで使用するアドレスセットアップフェーズの時間を HCLK 周期で定義します (図 199~図 202 を参照)。

0000 : ADDSET フェーズ時間 = 0 x HCLK クロックサイクル

...

1111 : ADDSET フェーズ時間 = 15 x HCLK クロックサイクル (リセット後のデフォルト)

注: 同期 NOR 型フラッシュアクセスでは、この値は無視されます。

25.5.7 FSMC レジスタマップ

次の表は FSMC レジスタの一覧です。

表 136. FSMC レジスタマップ

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
0000	FSMC_BCR1	予約済み												CBURSTRW	予約済み			ASYNCAWAIT	EXTMOD	WAITEN	WREN	WAITCFG	WRAPMOD	WAITPOL	BURSTEN	予約済み	FACCEN	MWID	MTYP			MUXEN	MBKEN
0008	FSMC_BCR2	予約済み												CBURSTRW	予約済み			ASYNCAWAIT	EXTMOD	WAITEN	WREN	WAITCFG	WRAPMOD	WAITPOL	BURSTEN	予約済み	FACCEN	MWID	MTYP			MUXEN	MBKEN
0010	FSMC_BCR3	予約済み												CBURSTRW	予約済み			ASYNCAWAIT	EXTMOD	WAITEN	WREN	WAITCFG	WRAPMOD	WAITPOL	BURSTEN	予約済み	FACCEN	MWID	MTYP			MUXEN	MBKEN
0018	FSMC_BCR4	予約済み												CBURSTRW	予約済み			ASYNCAWAIT	EXTMOD	WAITEN	WREN	WAITCFG	WRAPMOD	WAITPOL	BURSTEN	予約済み	FACCEN	MWID	MTYP			MUXEN	MBKEN
0004	FSMC_BTR1	予約 済み	ACC MOD	DATLAT		CLKDIV		BUSTURN			DATAST			ADDHLD			ADDSET																
000C	FSMC_BTR2	予約 済み	ACC MOD	DATLAT		CLKDIV		BUSTURN			DATAST			ADDHLD			ADDSET																
0014	FSMC_BTR3	予約 済み	ACC MOD	DATLAT		CLKDIV		BUSTURN			DATAST			ADDHLD			ADDSET																
001C	FSMC_BTR4	予約 済み	ACC MOD	DATLAT		CLKDIV		BUSTURN			DATAST			ADDHLD			ADDSET																
0104	FSMC_BWTR1	予約 済み	ACC MOD	予約済み			BUSTURN			DATAST			ADDHLD			ADDSET																	
010C	FSMC_BWTR2	予約 済み	ACC MOD	予約済み			BUSTURN			DATAST			ADDHLD			ADDSET																	
0114	FSMC_BWTR3	予約 済み	ACC MOD	予約済み			BUSTURN			DATAST			ADDHLD			ADDSET																	
011C	FSMC_BWTR4	予約 済み	ACC MOD	予約済み			BUSTURN			DATAST			ADDHLD			ADDSET																	

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

26 I²C(Inter-integrated Circuit) インタフェース

26.1 I²C の概要

I²C (inter-integrated 回路) 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 バス間のインタフェースとして機能します。マルチマスタ機能を提供し、すべての I²C バス固有のシーケンス、プロトコル、アービトレーション、タイミングを制御します。また、標準モード (最大 100 kHz の Sm) と高速モード (最大 400 kHz の Fm) に対応します。

このインタフェースは、CRC の生成と検証、SMBus (System Management Bus)、PMBus (Power Management Bus) など、さまざまな目的に使用できます。

デバイスの実装によっては、CPU の負荷を軽減するために DMA 機能を使用できます。

26.2 I²C の主な機能

- パラレルバス/I²C プロトコル間のコンバータ
- マルチマスタ機能：同じインタフェースがマスタまたはスレーブとして機能
- I²C マスタの機能：
 - － クロック生成
 - － START 信号、STOP 信号の生成
- I²C スレーブ機能：
 - － プログラマブルな I²C アドレス検出
 - － 2 つのスレーブアドレスを確認応答するデュアルアドレス機能
 - － STOP ビット検出
- 7ビット/10ビットアドレス指定および同報（General Call）の生成と検出
- 複数の通信速度をサポート：
 - － 標準モード（最大 100 kHz）
 - － 高速モード（最大 400 kHz）
- アナログノイズフィルタ
- ステータスフラグ：
 - － トランスマッタ/レシーバモードフラグ
 - － 最終バイト送信フラグ
 - － I²C ビジーフラグ
- エラーフラグ：
 - － マスタモードのアービトレーションロスト条件
 - － アドレス/データ送信後の確認応答不具合
 - － START/STOP コンディションの誤配置検出
 - － クロック伸長が無効な場合のオーバーラン/アンダーラン
- 2 つの割り込みベクタ：
 - － 正常なアドレス/データ通信用の割り込み
 - － エラー条件用の割り込み
- オプションのクロック伸長
- DMA 機能付きの 1バイトバッファ
- 設定可能な PEC（Packet Error Checking）の生成/検証：
 - － PEC 値は、送信モードで最終バイトとして送信可能
 - － 最後の受信バイトの PEC エラーチェック
- SMBus 2.0 互換性：
 - － クロックロータイムアウト遅延：25 ms
 - － マスタのクロックロー累積延長時間：10 ms
 - － スレーブのクロックロー累積延長時間：25 ms
 - － ハードウェアによる PEC 生成/検証（ACK 制御付き）
 - － ARP（アドレス解決プロトコル）対応
- PMBus 互換性

注： 製品によっては、上記の機能の一部を使用できないことがあります。I²C インタフェース実装によってサポートされる特定の機能については、製品のデータシートを参照してください。

26.3 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）または高速（最大 400 kHz）の I²C バスと接続できます。

26.3.1 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後はスレーブからマスタに、アービトレーションロストまたは STOP 生成が発生した場合はマスタからスレーブにそれぞれ自動的に切り替わり、マルチマスタ機能を可能にします。

通信の流れ

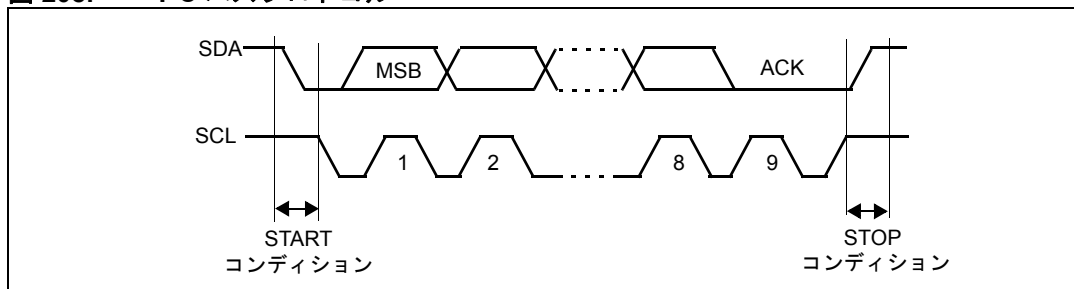
マスタモードでは、I²C インタフェースは、データ転送を開始してクロック信号を生成します。シリアルデータ転送は、常に START コンディションで始まり、STOP コンディションで終わります。START および STOP コンディションは、マスタモードではソフトウェアによって生成されます。

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/無効にできます。

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。[図 208](#) を参照してください。

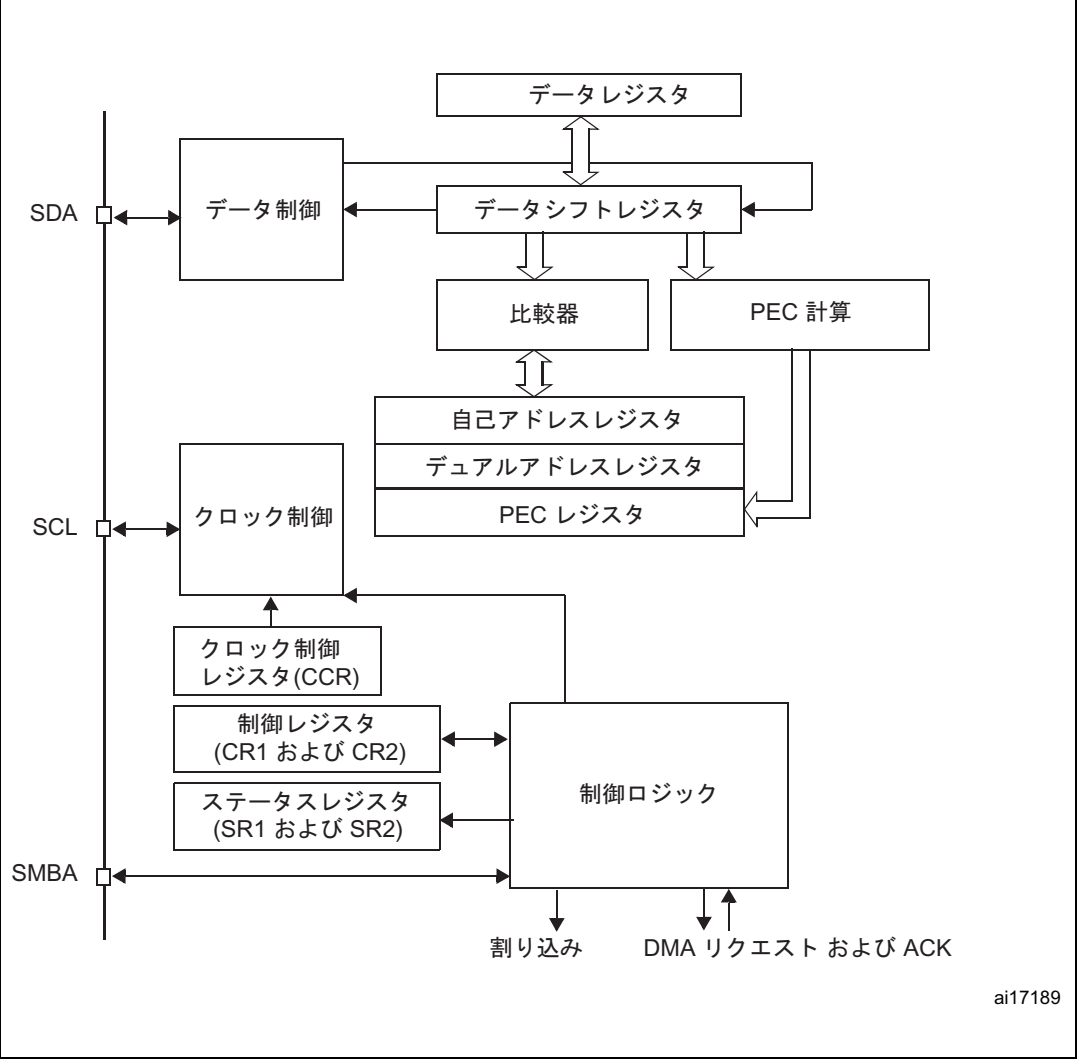
図 208. I²C バスプロトコル



確認応答は、ソフトウェアによって有効/無効にできます。I²C インタフェースアドレス（デュアルアドレス指定の 7 ビット/10 ビットや同報アドレス）は、ソフトウェアによって選択できます。

I²C インタフェースのブロック図を [図 209](#) に示します。

図 209. I²C ブロック図



1. SMBA は、SMBus モードでのオプション信号です。SMBus が無効な場合、この信号は適用されません。

26.3.2 I²C スレーブモード

デフォルトでは、I²C インタフェースはスレーブモードで動作します。デフォルトのスレーブモードからマスタモードに切り替えるには、START コンディションの生成が必要です。

正しいタイミングを生成するには、ペリフェラル入カクロックを I2C_CR2 レジスタにプログラミングする必要があります。ペリフェラル入カクロックは、少なくとも次の周波数以上でなければなりません。

- 標準モード：2 MHz
- 高速モード：4 MHz

注： 電圧スケーリング範囲 3 は高速モードでは許可されていません。詳細は、[セクション 5.1.5：ダイナミック電圧スケーリング管理 \(103 ページ\)](#) を参照してください。

START コンディションが検出されると、アドレスは SDA ラインから受信され、シフトレジスタに送信されます。その後、このアドレスはインタフェースのアドレス (OAR1) および OAR2 (ENDUAL=1 の場合) または同報アドレス (ENGC = 1 の場合) と比較されます。

注： 10ビットアドレスモードでは、この比較にはヘッダシーケンス (11110xx0) が含まれます。ここで、xx はアドレスの上位 2 ビットを示します。

ヘッダまたはアドレスの不一致：インタフェースはこれを無視し、別の START コンディションを待ちます。

ヘッダー一致 (10ビットモードのみ)：インタフェースは、ACK ビットがセットされている場合は確認応答パルスを生成し、8ビットのスレーブアドレスを待ちます。

アドレス一致：インタフェースは次の順序で動作します。

- ACK ビットがセットされている場合、確認応答パルスを生成します。
- ハードウェアによって ADDR ビットがセットされ、ITEVTEN 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。
- ENDUAL=1 の場合、ソフトウェアは、DUALF ビットを読み出して、どのスレーブアドレスが確認応答されたかをチェックする必要があります。

10ビットモードでは、アドレスシーケンスの受信後、スレーブは常にレシーバモードになります。スレーブがトランスミッタモードに入るのは、反復 START コンディションの後、一致するアドレスビットを持ち、最下位ビットがセットされた (11110xx1) ヘッダシーケンスを受信したときです。

TRA ビットは、スレーブがレシーバモードかトランスミッタモードかを示します。

スレーブトランスミッタ

アドレス受信に続いて ADDR をクリアした後、スレーブは、内部シフトレジスタを介して、DR レジスタから SDA ラインにバイトを送信します。

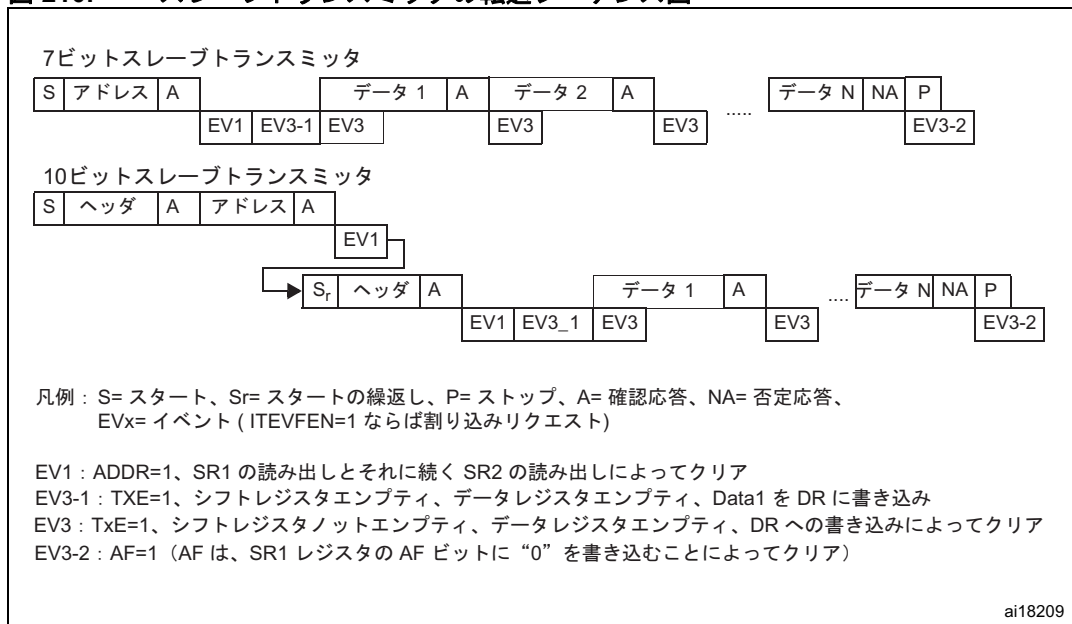
スレーブは、ADDR がクリアされ、DR が送信データで満杯になるまで、SCL のローレベルを伸長します ([図 210](#) の転送シーケンスの EV1 EV3 を参照)。

確認応答パルスが受信されると：

- ITEVTEN および ITBUFEN ビットがセットされている場合はハードウェアによって TxE ビットが割り込みでセットされます。

TxE がセットされ、次のデータ送信の終わりまでに I2C_DR レジスタに一部のデータが書き込まれていない場合、BTF ビットがセットされ、インタフェースは、I2C_SR1 の読み出しと、それに続く I2C_DR レジスタへの書き込みによって BTF がクリアされるまで、SCL のローレベルを延長して待ちます。

図 210. スレーブトランスミッタの転送シーケンス図



1. EV1 および EV3_1 イベントは、対応するソフトウェアシーケンスの終わりまで、SCL のローレベルを伸長します。
2. 次のバイト送信の終わりまでにソフトウェアシーケンスが完了しない場合、EV3 イベントは SCL のローレベルを伸長します。

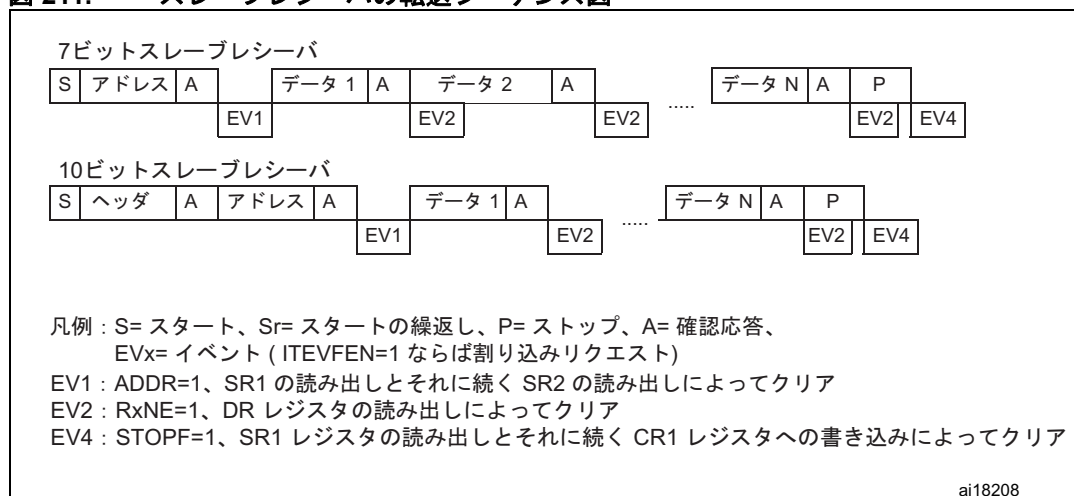
スレーブレシーバ

アドレス受信に続いて ADDR をクリアした後、スレーブは内部シフトレジスタを介して、SDA ラインから DR レジスタにバイトを受信します。各バイトの後、インタフェースは順に以下を生成します。

- ACK ビットがセットされている場合、確認応答パルスを生成します。
- RxNE ビットがハードウェアによってセットされ、ITEVFEN および ITBUFEN 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。

RxNE がセットされ、次のデータ受信の終わりまでに DR レジスタ内のデータが読み出されなかった場合、BTF ビットがセットされます。さらにインタフェースは、I2C_DRの読み出しによって BTF がクリアされるまで、SCL のローレベルを延長して待ちます (図 211 転送シーケンスを参照)。

図 211. スレーブレシーバの転送シーケンス図



- EV1 イベントは、対応するソフトウェアシーケンスの終わりまで SCL のローレベルを伸長します。
- 次のバイト受信の終わりまでにソフトウェアシーケンスが完了しない場合、EV2 イベントは SCL のローレベルを伸長します。
- SR1 レジスタの内容をチェックした後、ユーザは、セットされていることが分かったフラグごとに、完全なクリアシーケンスを実行してください。
 したがって、ADDR フラグと STOPF フラグの場合、I2C 割り込みルーチンの中に次のシーケンスが必要です。

```

READ SR1
if (ADDR == 1) {READ SR1; READ SR2}
if (STOPF == 1) {READ SR1; WRITE CR1}
    
```

 この目的は、ADDR フラグと STOPF フラグがセットされていることが分かった場合、両方を確実にクリアすることです。

スレーブ通信の終了

最後のデータバイトが転送された後、マスタは STOP コンディションを生成します。インタフェースは、このコンディションを検出して、次の動作を行います。

- STOPF ビットをセットし、ITEVFEN 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。

STOPF ビットは、SR1 レジスタの読み出しと、それに続く CR1 レジスタへの書き込みによってクリアされます（図 211：スレーブレシーバの転送シーケンス図 EV4 を参照）。

26.3.3 I²C マスタモード

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て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 ビットによってバス上に START コンディションが生成されると、マスタモードが選択されます。

次に、マスタモードに必要なシーケンスを示します。

- 正しいタイミングを生成するため、I2C_CR2 レジスタにペリフェラル入力クロックをプログラミングします。
- クロック制御レジスタを設定します。
- 立ち上がり時間レジスタを設定します。
- I2C_CR1 レジスタをプログラムして、ペリフェラルを有効にします。
- I2C_CR1 レジスタの START ビットをセットして、START コンディションを生成します。

ペリフェラル入力クロックは、少なくとも次の周波数以上でなければなりません。

- 標準モード : 2 MHz
- 高速モード : 4 MHz

注 : 電圧スケーリング範囲3 は高速モードでは許可されていません。詳細は、[セクション 5.1.5 : ダイナミック電圧スケーリング管理 \(103 ページ\)](#) を参照してください。

SCL マスタクロック生成

CCR ビットを使用して、SCL クロックのハイとローをそれぞれ立ち上がりエッジと立ち下がりエッジの生成から始めて生成します。スレーブが SCL ラインを伸長することがあるため、立ち上がりエッジの生成後、ペリフェラルが TRISE ビットでプログラムされている時間の終了時にバスからの SCL 入力をチェックします。

- SCL ラインがローの場合は、スレーブがバスを伸長していることを意味し、ハイレベルカウンタは SCL ラインがハイであることを検出するまで停止します。これにより、SCL クロックパラメータの最短 HIGH 期間を保証できます。
- SCL ラインがハイの場合、ハイレベルカウンタはカウントを続けます。

実際、スレーブがクロックをまったく伸長していなくても、ペリフェラルによる SCL 立ち上がりエッジ生成からペリフェラルによる SCL 立ち上がりエッジ検出までのフィードバックループには、時間がかかります。このループバックの継続時間は、SCL 立ち上がり時間 (SCL VIH 入力検出に影響する)、SCL 入力パス上にあるノイズフィルタによる遅延、そして APB クロックとの内部 SCL 入力同期による遅延にも関係します。フィードバックループによって使用される最長時間は、SCL 立ち上がり時間が何であっても SCL 周波数が安定するように TRISE ビットでプログラムされます。

START コンディション

BUSY ビットがクリアされるときに START ビットをセットすると、インタフェースが START コンディションを生成し、マスタモードに切り替わります (MSL ビットをセット)。

注 : マスタモードで START ビットをセットすると、インタフェースは現在のバイト転送の最後に RESTART コンディションを生成します。

START コンディションが送信されると、

- SB ビットがハードウェアによってセットされ、ITEVFEN 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。

次に、マスタは、SR1 レジスタの読み出しと、それに続く DR レジスタへのスレーブアドレスの書き込みを待ちます ([図 212](#) および [図 213](#) の転送シーケンスの EV5 を参照)。

スレーブアドレス送信

次に、スレーブアドレスが内部シフトレジスタを介して SDA ラインに送信されます。

- 10ビットアドレスモードでは、ヘッダシーケンスの送信によって、次のイベントが発生します。
 - ADDR10 ビットがハードウェアによってセットされ、ITEVFEN 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。

次に、マスタは SR1 レジスタの読み出しと、それに続く DR レジスタへの 2 番目のアドレスバイトの書き込みを待ちます (図 212 および 図 213 の転送シーケンスを参照)。

- ADDR ビットがハードウェアによってセットされ、ITEVFEN 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。

次に、マスタは SR1 レジスタの読み出しと、それに続く SR2 レジスタの読み出しを待ちます (図 212 および 図 213 の転送シーケンスを参照)。

- 7ビットアドレスモードでは、1 個のアドレスバイトが送信されます。

アドレスバイトが送信されると、

- ADDR ビットがハードウェアによってセットされ、ITEVFEN 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。

次に、マスタは SR1 レジスタの読み出しと、それに続く SR2 レジスタの読み出しを待ちます (図 212 および 図 213 の転送シーケンスを参照)。

マスタは、送信されたスレーブアドレスの LSB に応じて、トランスミッタモードに入るか、レシーバモードに入るかを決定できます。

- 7ビットアドレスモードでは、
 - トランスミッタモードに入るには、マスタが LSB をリセットしたスレーブアドレスを送信します。
 - レシーバモードに入るには、マスタが LSB をセットしたスレーブアドレスを送信します。
- 10ビットアドレスモードでは、
 - トランスミッタモードに入るには、マスタはヘッダ (11110xx0) を送信してから、スレーブアドレスを送信します (xx はアドレスの上位 2 ビットです)。
 - レシーバモードに入るには、マスタはヘッダ (11110xx0) を送信してから、スレーブアドレスを送信します。さらに、反復 START コンディションと、それに続いてヘッダ (11110xx1) を送信します (xx はアドレスの上位 2 ビットです)。

TRA ビットは、マスタがレシーバモードにあるか、トランスミッタモードにあるかを示します。

マスタトランスミッタ

アドレス送信に続いて、ADDR がクリアされた後、マスタは、内部シフトレジスタを介して DR レジスタから SDA ラインにバイトを送信します。

マスタは、最初のデータバイトが I2C_DR に書き込まれるまで待ちます (図 212 転送シーケンスの EV8_1 を参照)。

確認応答パルスが受信されると、TxE ビットがハードウェアによってセットされ、ITEVFEN および ITBUFEN 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。

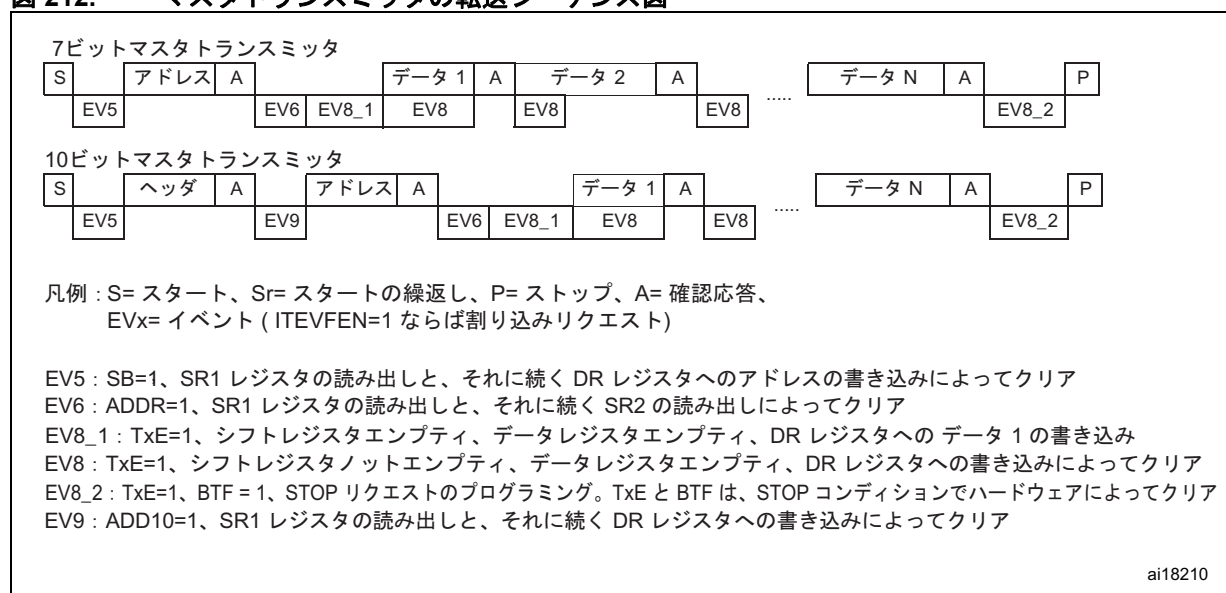
TxE がセットされていて、最後のデータ送信が終わるまでにデータバイトが DR レジスタに書き込まなかった場合、BTF がセットされ、インタフェースは、I2C_DR への書き込みによって BTF がクリアされるまで、SCL のローレベルを伸長して待ちます。

通信の終了

最後のバイトが DR レジスタに書き込まれた後、STOP ビットがソフトウェアによってセットされ、STOP コンディションを生成します (図 212 の転送シーケンスの EV8_2 を参照)。インタフェースは自動的にスレーブモードに戻ります (MSL ビットがクリアされます)。

注： STOP コンディションは、TxE または BTF ビットがセットされたとき、EV8_2 イベント時にプログラミングしてください。

図 212. マスタトランスミッタの転送シーケンス図



- EV5、EV6、EV9、EV8_1、および EV8_2 の各イベントは、対応するソフトウェアシーケンスの終わりまで、SCL のローレベルを伸長します。
- 次のバイト送信の終わりまでにソフトウェアシーケンスが完了しない場合、EV8 イベントは SCL のローレベルを伸長します。

マスタレシーバ

アドレス送信に続いて ADDR がクリアされると、I²C インタフェースはマスタレシーバモードに入ります。このモードでは、インタフェースは内部シフトレジスタを介して SDA ラインから DR レジスタにバイトを受信します。各バイトの後、インタフェースは順に以下を生成します。

1. ACK ビットがセットされている場合、確認応答パルスを生成します。
2. RxNE ビットがセットされ、ITEVTEN および ITBUFEN 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（図 213 転送シーケンスの EV7 を参照）。

RxNE ビットがセットされ、最後のデータ受信の終わりまでに DR レジスタ内のデータが読み出されなかった場合、BTF ビットがハードウェアによってセットされ、インタフェースは、DR レジスタの読み出しによって BTF がクリアされるまで、SCL のローレベルを伸長して待ちます。

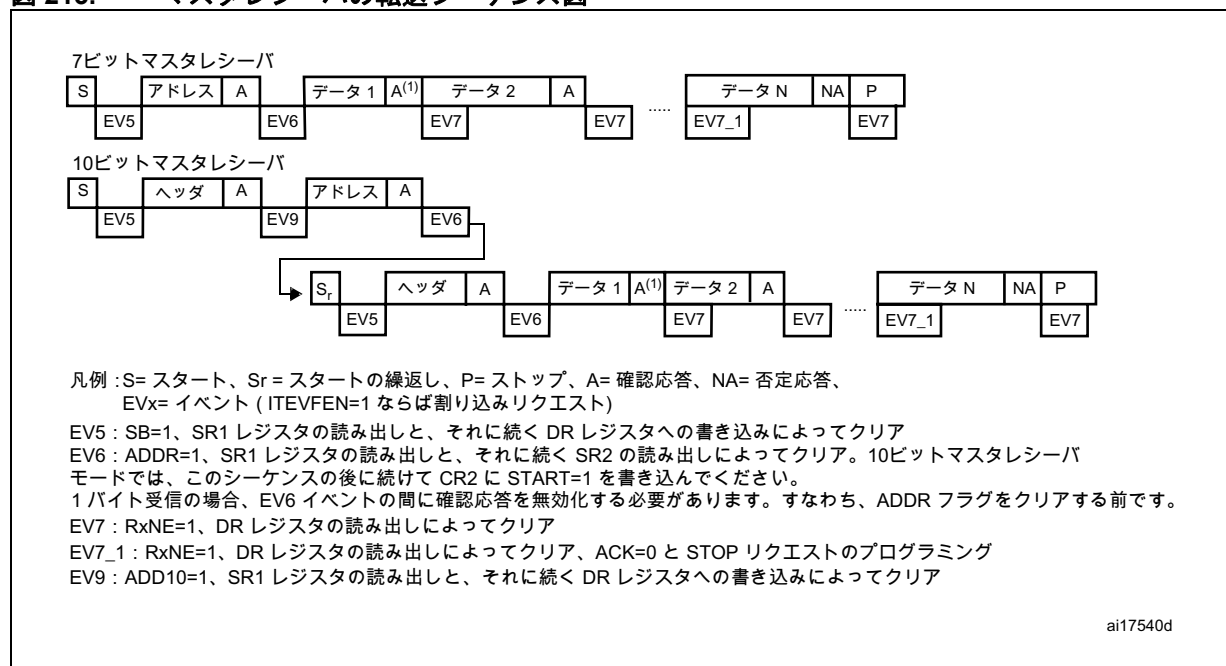
通信の終了

マスタは、スレーブから受信した最終バイトに対して NACK を送信します。この NACK を受信したスレーブは、SCL ラインと SDA ラインの制御を解除します。これによって、マスタは STOP/RESTART コンディションを送信できます。

1. 最後の受信データバイトの後で否定応答パルスを生成するには、最後から 2 番目の RxNE イベントの後で最後から 2 番目のデータバイトの読み出し直後に、ACK ビットをクリアする必要があります。
2. STOP/RESTART コンディションを生成するには、ソフトウェアは、最後から 2 番目の RxNE イベントの後で最後から 2 番目のデータバイトの読み出し直後に、STOP/START ビットをセットする必要があります。
3. 1バイトの受信に備えて、確認応答は EV6 中（ADDR フラグがクリアされる前）に無効化され、STOP コンディションは EV6 の後に生成されます。

STOP コンディションの生成後、インタフェースは自動的にスレーブモードに戻ります（MSL ビットがクリアされます）。

図 213. マスタレシーバの転送シーケンス図



1. 1 バイトが受信された場合、それは NA です。
2. EV5、EV6、および EV9 イベントは、対応するソフトウェアシーケンスの終わりまで、SCL のローレベルを伸長します。
3. 次のバイト受信の終わりまでにソフトウェアシーケンスが完了しない場合、EV7 イベントは SCL のローレベルを伸長します。
4. EV7_1 ソフトウェアシーケンスは、現在のバイト転送の ACK パルスより前に完了する必要があります。

現在のバイト転送の ACK パルスより前に EV7-1 ソフトウェアシーケンスが完了しない場合、以下の手順を推奨します。

次の条件を実現するには、この手順に従う必要があります。

- 最後のデータ受信が終わる前に、ACK ビットが遅れずにローにセットされます。
- 最後のデータ受信の後、補助データの受信なしに STOP ビットがハイにセットされます。

2バイト受信の場合 :

- ADDR = 1 になるまで待ちます (ADDR フラグがクリアされるまで SCL のローレベルを伸長します)。
- ACK をローにセットし、POS をハイにセットします。
- ADDR フラグをクリアします。
- BTF = 1 になるまで待ちます (DR にデータ 1、シフトレジスタにデータ 2、データ 1 が読み出されるまで SCL のローレベルを伸長)。
- STOP をハイにセットします。
- データ 1 と 2 を読み出します。

N > 2バイト受信の場合 (N-2 データ受信から) :

- BTF = 1 になるまで待ちます (DR にデータ N-2、シフトレジスタにデータ N-1、データ N-2 が読み出されるまで SCL のローレベルを伸長)。
- ACK をローにセットします。
- データ N-2 を読み出します。
- BTF = 1 になるまで待ちます (DR にデータ N-1、シフトレジスタにデータ N、データ N-1 が読み出されるまで SCL のローレベルを伸長)。
- STOP をハイにセットします。
- データ N-1 と N を読み出します。

26.3.4 エラー条件

以下は、通信障害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)

このエラーは、I²C インタフェースがアドレスやデータの転送中に外部の STOP または START コンディションを検出したときに発生します。このとき、

- BERR ビットがセットされ、ITERREN 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。
- スレーブモードの場合：データは破棄され、ラインがハードウェアによって解放されます。
 - START の誤配置である場合、スレーブはこれを RESTART とみなし、アドレスまたは STOP コンディションを待ちます。
 - STOP の誤配置である場合、スレーブは STOP コンディションの場合のように動作し、ラインがハードウェアによって解放されます。
- マスタモードの場合：ラインは解放されず、現在の送信の状態は影響を受けません。現在の送信を中止するかどうかは、ソフトウェア次第です。

確認応答障害 (AF)

このエラーは、インタフェースが否定応答ビットを検出したときに発生します。このとき、

- AF ビットがセットされ、ITERREN 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。
- NACK を受信したトランスミッタは、通信をリセットする必要があります。
 - スレーブの場合：ラインがハードウェアによって解放されます。
 - マスタの場合：ソフトウェアが STOP または反復 START コンディションを生成する必要があります。

アービトレーションロスト (ARLO)

このエラーは、I²C インタフェースがアービトレーションロスト条件を検出したときに発生します。このとき、

- ARLO ビットがハードウェアによってセットされます (ITERREN 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)。
- I²C インタフェースは自動的にスレーブモードに戻ります (MSL ビットがクリアされます)。I²C がアービトレーションを失うと、同じ転送ではそのスレーブアドレスを確認応答できませんが、通信を許可されたマスタからの反復 START 後に確認応答できます。
- ラインがハードウェアによって解放されます。

オーバーラン/アンダーランエラー (OVR)

スレーブモードでオーバーランエラーが発生するのは、クロック伸長が無効であり、I²C インタフェースがデータを受信しているときです。インタフェースが次のバイトを受信する前、インタフェースはすでにバイトを受信し (RxNE=1)、DR 内のデータはまだ読み出されていません。このとき、

- 最後に受信されたバイトは失われます。
- オーバーランエラーが発生した場合、ソフトウェアで RxNE ビットをクリアし、トランスミッターで最後の受信バイトを再送信してください。

スレーブモードでアンダーランエラーが発生するのは、クロック伸長が無効であり、I²C インタフェースがデータを送信しているときです。次のバイト用のクロックが到着する前、インタフェースはまだ次のバイト (TxNE=1) で DR を更新していません。このとき、

- DR レジスタ内の同じバイトが再送信されます。
- ユーザは、アンダーランエラー中にレシーバ側で受信されたデータが破棄され、I²C バス規格で指定されたクロックロー時間内に次のバイトが書き込まれるようにしてください。

送信される最初のバイトに関しては、ADDR がクリアされてから最初の SCL 立ち上がりエッジまでに DR を書き込む必要があります。これができない場合、レシーバは最初のデータを破棄する必要があります。

26.3.5 SDA/SCL ライン制御

- クロック伸長が有効な場合：
 - トランスミッターモード：TxNE=1 および BTF=1 の場合：インタフェースは、送信前にクロックラインをローレベルに保ち、マイクロコントローラがデータレジスタにバイトを書き込むのを待ちます (バッファとシフトレジスタは、いずれも空)。
 - レシーバモード：RxNE=1 および BTF=1 の場合：インタフェースは、受信後にクロックラインをローレベルに保ち、マイクロコントローラがデータレジスタ内のバイトを読み出すのを待ちます (バッファとシフトレジスタは、いずれも満杯)。
- スレーブモードでクロック伸長が無効な場合：
 - オーバーランエラー：RxNE=1 で、次のバイトが受信されるまでに DR が読み出されていない場合。最後に受信されたバイトは失われます。
 - アンダーランエラー：TxNE=1 で、次のバイトを送信するまでに DR への書き込みが行われていない場合。同じバイトが再送信されます。
 - 書き込み衝突は管理されません。

26.3.6 SMBus

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。これは I²C の動作原理をベースにしています。SMBus は、システムおよびパワーマネジメント関連タスク向けの制御バスを提供します。SMBus を使用すると、システムは個々の制御ラインをトグルする代わりに、各デバイスとの間でメッセージをやり取りできます。

SMBus 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。

SMBus と I²C との類似点

- 2 線バスプロトコル（1 クロック、1 データ）+ SMBus アラートライン（オプション）
- マスタ／スレーブ通信（マスタがクロックを供給します）
- マルチマスタ機能
- I²C 7ビットアドレス指定フォーマットに類似した SMBus データフォーマット（図 208）。

SMBus と I²C との相違点

次の表に、SMBus と I²C との相違点を示します。

表 137. SMBus と I²C

SMBus	I ² C
最大速度：100 kHz	最大速度：400 kHz
最小クロック速度：10 kHz	最小クロック速度の規定なし
クロックロータイムアウト：35 ms	タイムアウトなし
論理レベルは固定	論理レベルは V _{DD} に依存
種々のアドレスタイプ（予約、ダイナミック、など）	7ビット、10ビット、および同報のスレーブアドレスタイプ
種々のバスプロトコル（クイックコマンド、プロセスコール、など）	バスプロトコルなし

SMBus アプリケーションの用途

SMBus を使用すれば、デバイスは製造者情報を提供し、モデル／部品番号をシステムに通知し、サスペンドイベントに備えてその状態を保存し、種々のエラーを報告し、制御パラメータを受け付け、そのステータスを返すことができます。SMBus は、システムおよびパワーマネジメント関連タスク向けの制御バスを提供します。

デバイス識別

SMBus 上にスレーブとして存在するすべてのデバイスには、スレーブアドレスと呼ばれるユニークなアドレスがあります。予約済みのスレーブアドレスの一覧については、SMBus 仕様 バージョン2.0 (<http://smbus.org/>) を参照してください。

バスプロトコル

SMBus 仕様では、最大 9 種のバスプロトコルをサポートします。これらのプロトコルと SMBus アドレスタイプの詳細については、SMBus 仕様 バージョン2.0 (<http://smbus.org/>) を参照してください。これらのプロトコルは、ユーザのソフトウェアで実装してください。

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。アドレス解決プロトコル (ARP) には、以下の属性があります。

- アドレス割り当てには、標準の SMBus 物理層アービトレーション機構を使用します。
- 割り当てられたアドレスは、デバイスに電源が供給されている間は一定しています。デバイスの電源切断後もアドレスの保持は可能です。
- アドレス割り当ての後、新たな SMBus パケットのオーバーヘッドは生じません。(つまり、割り当てられたスレーブアドレスへのその後のアクセスでは、固定アドレスのデバイスへのアクセスと同じオーバーヘッドになります。)
- いずれの SMBus マスタも、バスをエニュメレーションできます。

ユニークデバイス識別子 (UDID)

アドレス割り当てを目的として各デバイスを分離する仕組みを提供するには、各デバイスがユニークデバイス識別子 (UDID) を実装する必要があります。

128ビット UDID と ARP の詳細については、SMBus 仕様 バージョン 2.0 (<http://smbus.org/>) を参照してください。

SMBus アラートモード

SMBus Alert は、マスタ機能をピンに置き換えたいデバイス用の割り込みラインを持つオプション信号です。SMBA は、SCL 信号や SDA 信号と同じく、ワイヤード AND 信号です。SMBA は、SMBus 同報アドレスとともに使用されます。SMBus で呼び出されるメッセージの長さは 2 バイトです。

スレーブ専用デバイスは、I2C_CR1 レジスタの ALERT ビットをセットして、ホストに接続したいことを SMBA を通じて通知できます。ホストは割り込みを処理し、アラート応答アドレス (ARA と呼ばれ、値は 0001 100X) を通じて全 SMBA デバイスに同時にアクセスします。SMBA をローレベル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。このステータスは、I2C_SR1 レジスタの SMBALERT ステータスフラグを使用して識別されます。ホストは、モディファイド受信バイト動作を行います。スレーブ送信デバイスによって提供された 7 ビットのデバイスアドレスは、バイトの上位 7 ビットに配置されます。8 番目のビットは 0 または 1 にできます。

複数のデバイスが SMBA をローレベルに引き下げている場合、スレーブアドレス転送時の標準的なアービトレーションによって、最高優先順位 (最小アドレス) のデバイスが通信権を獲得します。スレーブアドレスの確認応答後、デバイスは、その SMBA の引き下げを解放する必要があります。メッセージ転送の完了時に、SMBA が依然としてローレベルであることにホストが気づいた場合、再び ARA の読み出しが行われます。

SMBA 信号を実装していないホストは、周期的に ARA にアクセスできます。

SMBus アラートモードの詳細については、SMBus 仕様 バージョン 2.0 (<http://smbus.org/>) を参照してください。

タイムアウトエラー

I²C と SMBus では、タイミング仕様に違いがあります。

SMBus では、クロックロータイムアウトとして 35 ms の TIMEOUT が定義されています。さらに SMBus では、スレーブデバイスのクロックロー累積延長時間として TLOW:SEXT を規定しています。また、マスタデバイスのクロックロー累積延長時間として TLOW:MEXT も規定しています。これらのタイムアウトの詳細については、SMBus 仕様 バージョン 2.0 (<http://smbus.org/>) を参照してください。

I2C_SR1 レジスタのステータスフラグ Timeout や Tlow Error は、この機能のステータスを示します。

SMBus モードでのインタフェースの使い方

I²C モードから SMBus モードに切り替えるには、次のシーケンスを実行してください。

- I2C_CR1 レジスタの SMBus ビットをセットします。
- アプリケーションの必要に応じて、I2C_CR1 レジスタの SMBTYPE ビットと ENARP ビットを設定します。

デバイスをマスタとして設定したい場合、[セクション 26.3.3 : I2C マスタモード](#)に記載されている START コンディションの生成手順に従ってください。そうでない場合は、[セクション 26.3.2 : I2C スレーブモード](#)に記載されている手順に従ってください。

アプリケーションは、ソフトウェアによってさまざまな SMBus プロトコルを制御する必要があります。

- ENARP=1 および SMBTYPE=0 の場合 : SMB デバイスのデフォルトアドレスが確認応答されます。
- ENARP=1 および SMBTYPE=1 の場合 : SMB ホストヘッダが確認応答されます。
- SMBALERT=1 の場合 : SMB アラート応答アドレスが確認応答されます。

26.3.7 DMA リクエスト

DMA リクエストは、データ転送用にのみ生成されます（有効な場合）。DMA リクエストが生成されるのは、送信中にデータレジスタが空になったとき、および受信中にデータレジスタが満杯になったときです。DMA は、I2C データ転送前に初期化し、有効にする必要があります。I2C_CR2 レジスタの DMAEN ビットは、ADDR イベント前にセットされなければなりません。マスタモードまたはスレーブモードでクロック伸長が有効の場合は、ADDR フラグをクリアする前に ADDR イベント中に DMAEN ビットをセットすることもできます。DMA リクエストは、現在のバイト転送の終わりまでに処理される必要があります。対応する DMA ストリームにプログラミングされたデータ転送回数に達したとき、DMA コントローラは、I²C インタフェースに転送終了（EOT）信号を送信し、転送完了割り込みを生成します（有効な場合）。

- マスタトランスミッタ : EOT 割り込み後の割り込みルーチンでは、DMA リクエストを無効にし、BTF イベントの発生を待ってから、STOP コンディションをプログラミングします。
- マスタレシーバ
 - － 受信するバイト数が 2 以上であるとき、DMA コントローラは、最後から 2 番目のデータバイト（number_of_bytes - 1）に対応するハードウェア信号（EOT_1）を送信します。I2C_CR2 レジスタで LAST ビットがセットされた場合、I²C は、EOT_1 に続く次のバイトの後に、自動的に NACK を送信します。ユーザは、DMA 転送完了割り込みルーチンで STOP コンディションを生成できます（有効な場合）。
 - － 1バイトを受信しなければならない場合 : EV6 イベント中に NACK をプログラムする必要があります。つまり、ADDR=1 のときに ADDR フラグをクリアする前に ACK=0 をプログラムします。その後、ADDR フラグをクリアする前または DMA 転送完了割り込みルーチン中に STOP コンディションをプログラムできます。

DMA を使用した送信

DMA モードでの送信を有効にするには、I2C_CR2 レジスタの DMAEN ビットをセットします。データは、TxE ビットがセットされるたびに、DMA ペリフェラルを使用して設定されるメモリ領域 (DMA 仕様を参照) から I2C_DR レジスタにロードされます。I²C 送信の DMA ストリーム x (x はストリーム番号) を割り付けるには、次のシーケンスを実行します。

1. I2C_DR レジスタのアドレスを DMA_SxPAR レジスタにセットします。データは、毎回の 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_SxMA0R レジスタにメモリアドレスをセットします (バブルバッファモードの場合は、DMA_SxMA1R レジスタも)。データは、毎回の TxE イベント後に、このメモリから I2C_DR にロードされます。
3. 転送されるバイトの総数を DMA_SxNDTR レジスタに設定します。毎回の TxE イベント後に、この値はデクリメントされます。
4. DMA_SxCR レジスタの PL[0:1] ビットを使用して、DMA ストリームの優先順位を設定します。
5. DMA_SxCR レジスタの DIR ビットをセットし、アプリケーション要件に応じて 1/2 転送終了後または転送完了後に、割り込みを設定します。
6. DMA_SxCR レジスタの EN ビットをセットすることによって、ストリームをアクティブにします。

DMA コントローラのレジスタにプログラミングされたデータ転送回数に達すると、DMA コントローラは I²C インタフェースに転送終了 (EOT/EOT_1) 信号を送信し、DMA は DMA ストリーム割り込みベクタ上に割り込みを生成します (有効な場合)。

注： DMA が送信に使用される場合、I2C_CR2 レジスタの ITBUFEN ビットを有効にしないでください。

DMA を使用した受信

DMA モードでの受信を有効にするには、I2C_CR2 レジスタの DMAEN 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たびに、DMA ペリフェラルを使用して設定されるメモリ領域 (DMA 仕様を参照) に I2C_DR レジスタからロードされます。I²C 受信の DMA ストリーム x (x はストリーム番号) を割り付けるには、次のシーケンスを実行します。

1. I2C_DR レジスタのアドレスを DMA_SxPAR レジスタにセットします。データは、毎回の 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_SxMA0R レジスタにメモリアドレスをセットします (バブルバッファモードの場合は、DMA_SxMA1R レジスタも)。データは、毎回の RxNE イベント後に、I2C_DR レジスタからこのメモリ領域にロードされます。
3. 転送されるバイトの総数を DMA_SxNDTR レジスタに設定します。毎回の RxNE イベント後に、この値はデクリメントされます。
4. DMA_SxCR レジスタの PL[0:1] ビットを使用して、ストリームの優先順位を設定します。
5. DIR ビットをリセットし、アプリケーション要件に応じて 1/2 転送終了後または転送完了後に、DMA_SxCR レジスタに割り込みを設定します。
6. DMA_SxCR レジスタの EN ビットをセットすることによって、ストリームをアクティブにします。

DMA コントローラのレジスタにプログラミングされたデータ転送回数に達すると、DMA コントローラは I²C インタフェースに転送終了 (EOT/EOT_1) 信号を送信し、DMA は DMA ストリーム割り込みベクタ上に割り込みを生成します (有効な場合)。

注： DMA が受信に使用される場合、I2C_CR2 レジスタの ITBUFEN ビットを有効にしないでください。

26.3.8 パケットエラーチェック

通信の信頼性を高めるため、PEC 計算回路が実装されています。PEC の計算には、 $C(x) = x^8 + x^2 + x + 1$ という CRC-8 多項式を各ビットに連続的に適用します。

- PEC 計算を有効にするには、I2C_CR1 レジスタの ENPEC ビットをセットします。PEC は、アドレスと R/W ビットを含むすべてのメッセージバイトに基づいて計算される CRC-8 コードです。
 - － 送信時：最後のバイトに対応する TxE イベントの後、I2C_CR1 レジスタの PEC 転送ビットをセットします。PEC は、最後に送信されたバイトの後で転送されます。
 - － 受信時：最後のバイトに対応する RxNE イベントの後、I2C_CR1 レジスタの PEC ビットをセットし、次に受信するバイトが内部的に計算した PEC と等しくない場合にレシーバが NACK を送信するようにします。マスタレシーバの場合、チェックの結果にかかわらず、PEC の後には NACK が続く必要があります。PEC は、スレーブモードで CRC 受信の ACK よりも前にセットする必要があります。これは、マスタモードで ACK がローにセットされる時にセットされなければなりません。
- PECERR エラーフラグ／割り込みは、I2C_SR1 レジスタでも使用できます。
- DMA と PEC 計算機能がともに有効な場合：
 - － 送信時：I²C インタフェースが DMA コントローラから EOT 信号を受信すると、最終バイトの後で自動的に PEC を送信します。
 - － 受信時：I²C インタフェースが DMA コントローラから EOT_1 信号を受信すると、次のバイトを自動的に PEC とみなし、その検査を行います。PEC 受信後に DMA リクエストが生成されます。
- 中間での PEC 転送を可能にするため、I2C_CR2 レジスタには制御ビット（LAST ビット）が用意されており、本当に最後の DMA 転送かどうかを判定します。マスタレシーバの最後の DMA リクエストである場合、最後に受信されたバイトの後に自動的に NACK が送信されます。
- PEC 計算は、アービトレーションロスによって損なわれます。

26.4 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 138. 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	有効制御ビット
START ビット送信 (マスタ)	SB	ITEVFEN
アドレス送信 (マスタ) またはアドレス一致 (スレーブ)	ADDR	
10ビットヘッダ送信 (マスタ)	ADD10	
STOP 受信 (スレーブ)	STOPF	
データバイト転送終了	BTF	
受信バッファノットエンプティ	RxNE	ITEVFEN および ITBUFEN
送信バッファエンプティ	TxE	
バスエラー	BERR	ITERREN
アービトラションロス (マスタ)	ARLO	
確認応答障害	AF	
オーバーラン/アンダーラン	OVR	
PEC エラー	PECERR	
Timeout/Tlow エラー	TIMEOUT	
SMBus アラート	SMBALERT	

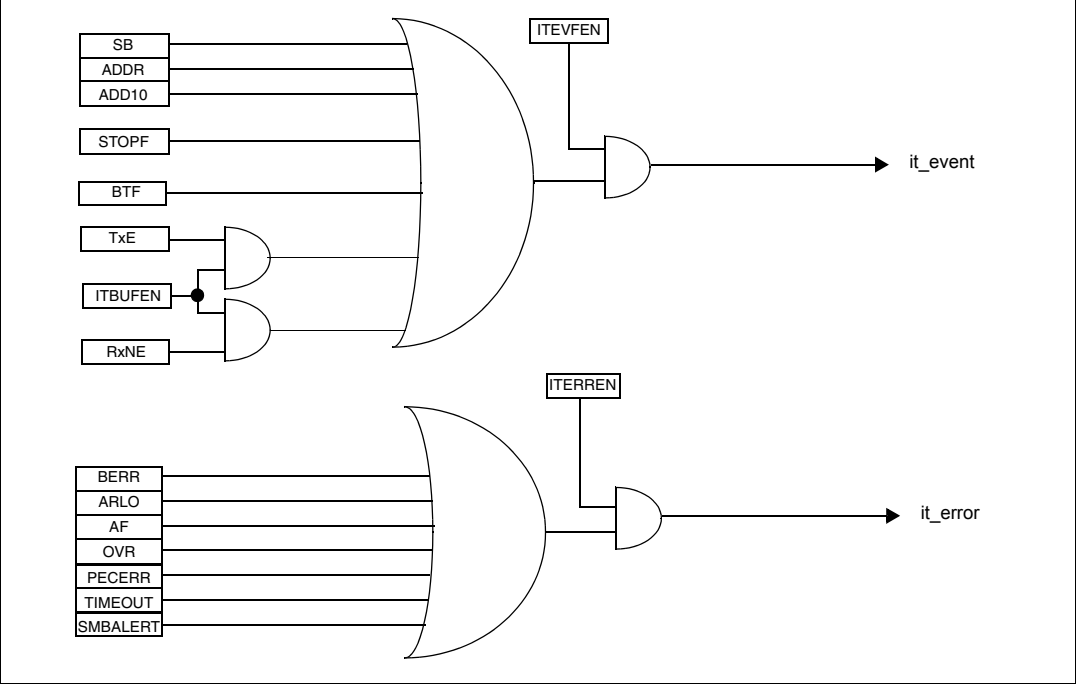
注 : SB、ADDR、ADD10、STOPF、BTF、RxNE、および TxE の各ビットは、同じ割り込みチャンネル上で論理和をとります。

BERR、ARLO、AF、OVR、PECERR、TIMEOUT、および SMBALERT の各ビットは、同じ割り込みチャンネル上で論理和をとります。



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。

図 214. I²C 割り込みマッピング図



26.5 I²C 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（Cortex®-M3 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。詳細については、[セクション 30.16.2：タイマ、ウォッチドッグ、および I2C のデバッグサポート \(904 ページ\)](#) を参照してください。

26.6 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、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。

26.6.1 I²C 制御レジスタ 1(I2C_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
SWRST	予約済み	ALERT	PEC	POS	ACK	STOP	START	X STRETCH	ENGCR	ENPEC	ENARP	SMB TYPE	予約済み	SMBUS	PE
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw

- ビット 15 **SWRST**：ソフトウェアリセット
- このビットがセットされると、I2C はリセット状態になります。このビットをリセットする前に、I2C ラインが解放されていて、バスが空いていることを確認してください。
- 0：I²C ペリフェラルはリセット状態ではありません。
- 1：I²C ペリフェラルはリセット状態です。
- 注：** このビットは、エラーまたはロック状態後にペリフェラルの再初期化に使用できます。例として、**BUSY** ビットがセットされ、バス上にグリッチがあるためにロックされたままになる場合、**SWRST** ビットを使用してこの状態から抜け出すことができます。

- ビット 14 予約済み、リセット値に保持する必要があります。
- ビット 13 **ALERT**：SMBus アラート
- このビットは、ソフトウェアによってセット/クリアされ、PE=0 のときにハードウェアによってクリアされます。
- 0：SMBA ピンをハイレベルに解放します。アラート応答アドレスヘッダの後には NACK が続きます。
- 1：SMBA ピンをローレベルに駆動します。アラート応答アドレスヘッダの後には ACK が続きます。

- ビット 12 **PEC**：パケットエラーチェック
- このビットは、ソフトウェアによってセット/クリアされ、PEC が転送されたとき、START/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。
- 0：PEC は転送されません。
- 1：PEC が転送されます（送信または受信モード）。
- 注：** PEC 計算は、アービトレーションロスによって損なわれます。

ビット 11 POS : 確認応答/PEC 位置 (データ受信)

このビットは、ソフトウェアによってセット/クリアされ、PE=0 のときにハードウェアによってクリアされます。

0 : ACK ビットは、シフトレジスタで受信されている現在のバイトの (N)ACK を制御します。PEC ビットは、シフトレジスタ内の現在のバイトが PEC であることを示します。

1 : ACK ビットは、シフトレジスタで受信される次のバイトの (N)ACK を制御します。PEC ビットは、シフトレジスタ内の次のバイトが PEC であることを示します。

注: POS ビットは、マスタモードの 2 バイト受信設定でのみ使用する必要があります。また、[セクション : マスタレシーバ \(689 ページ\)](#) で推奨する 2 バイト受信手順に説明されているように、データ受信の開始前に設定する必要があります。

ビット 10 ACK : 確認応答有効

このビットは、ソフトウェアによってセット/クリアされ、PE=0 のときにハードウェアによってクリアされます。

0 : 確認応答は返されません。

1 : バイトの受信後 (アドレスまたはデータの一致)、確認応答が返されます。

ビット 9 STOP : STOP 生成

このビットは、ソフトウェアによってセット/クリアされ、STOP コンディションの検出時にハードウェアによってクリアされ、タイムアウトエラーの検出時にハードウェアによってセットされます。

マスタモード :

0 : STOP は生成されません。

1 : 現在のバイト転送後、または現在の START コンディションの送信後に、STOP が生成されます。

スレーブモード :

0 : STOP は生成されません。

1 : 現在のバイト転送後に SCL ラインと SDA ラインを解放します。

ビット 8 START : START 生成

このビットは、ソフトウェアによってセット/クリアされ、START の送信時または 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 : START は生成されません。

1 : 反復 START が生成されます。

スレーブモード :

0 : START は生成されません。

1 : バスが空いていれば、START が生成されます。

ビット 7 NOSTRETCH : クロック伸長ディセーブル (スレーブモード)

このビットは、スレーブモードで ADDR または BTF フラグがセットされているとき、ソフトウェアによってリセットされるまで、クロック伸長を無効にするために使用されます。

0 : クロック伸長は有効です。

1 : クロック伸長は無効です。

ビット 6 ENGC : 同報イネーブル

0 : 同報は無効です。アドレス 00h は NACK されます。

1 : 同報は有効です。アドレス 00h は ACK されます。

ビット 5 ENPEC : PEC イネーブル

0 : PEC 計算は無効です。

1 : PEC 計算は有効です。

ビット 4 ENARP : ARP イネーブル

0 : ARP は無効です。

1 : ARP は有効です。

SMBTYPE=0 の場合、SMBus デバイスのデフォルトアドレスが認識されます。

SMBTYPE=1 の場合、SMBus ホストのアドレスが認識されます。

ビット 3 **SMBTYPE** : SMBus タイプ

- 0 : SMBus デバイス
- 1 : SMBus ホスト

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 **SMBUS** : SMBus モード

- 0 : I²C モード
- 1 : SMBus モード

ビット 0 **PE** : ペリフェラルイネーブル

- 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： 通信中にこのビットがリセットされた場合、現在の通信の最後にアイドル状態に戻ったとき、ペリフェラルが無効になります。
通信の最後には PE=0 となるため、すべてのビットがリセットされます。
マスタモードでは、通信の最後まで、このビットをリセットしてはなりません。

注： STOP、START、または PEC ビットがセットされると、このビットがハードウェアによってクリアされるまで、ソフトウェアは I2C_CR1 に書き込みアクセスを実行してはいけません。さもないと、第 2 の STOP、START、または PEC リクエストをセットする危険性があります。

26.6.2 I²C 制御レジスタ 2(I2C_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
予約済み			LAST	DMA EN	ITBUFE N	ITEVTE N	ITERRE N	予約済み		FREQ[5:0]					
			rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

ビット 15:13 予約済み、リセット値に保持する必要があります。

ビット 12 **LAST** : DMA 最終転送

0 : 次の DMA EOT は最終転送ではありません。

1 : 次の DMA EOT は最終転送です。

注 : このビットは、最後の受信データでの NACK の生成を可能にするために、マスタレシーバモードで使います。

ビット 11 **DMAEN** : DMA リクエストイネーブル

0 : DMA リクエストは無効です。

1 : TxE=1 または RxNE =1 のとき、DMA リクエストは有効です。

ビット 10 **ITBUFEN** : バッファ割り込みイネーブル

0 : TxE = 1 または RxNE = 1 で、割り込みは生成されません。

1 : TxE = 1 または RxNE = 1 で、イベント割り込みを生成します (DMAEN ビットの状態には依存しません)。

ビット 9 **ITEVTEN** : イベント割り込みイネーブル

0 : イベント割り込みは無効です。

1 : イベント割り込みは有効です。

この割り込みは、以下の場合に生成されます。

- SB = 1 (マスタ)
- ADDR = 1 (マスタ/スレーブ)
- ADD10= 1 (マスタ)
- STOPF = 1 (スレーブ)
- BTF = 1 (ただし、TxE または RxNE イベントが発生していないとき)
- ITBUFEN = 1 で TxE イベントが 1 になったとき
- ITBUFEN = 1 で RxNE イベントが 1 になったとき

ITERREN : 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。

この割り込みは、以下の場合に生成されます。

- BERR = 1
- ARLO = 1
- AF = 1
- OVR = 1
- PECERR = 1
- TIMEOUT = 1
- SMBALERT = 1

ビット 7:6 予約済み、リセット値に保持する必要があります。

ビット 5:0 **FREQ[5:0]** : ペリフェラルクロック周波数

FREQ ビットは、APB クロック周波数値 (I2C ペリフェラルは APB に接続) を使用して設定する必要があります。FREQ フィールドはペリフェラルが使用し、I2C 仕様に準拠したデータセットアップおよびホールド時間を生成します。最小許容周波数は 2 MHz であり、最大周波数は最大 APB 1 周波数 (32 MHz) と本質的な制限 (46 MHz) によって制約されます。

0b000000 : 許可されません。

0b000001 : 許可されません。

0b000010 : 2 MHz

...

0b100000 : 32 MHz

0b100000 を超える場合 : 許可されません。

26.6.3 I²C 自己アドレスレジスタ 1 (I2C_OAR1)

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
ADD MODE	予約済み					ADD[9:8]		ADD[7:1]							ADD0
rw						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 **ADDMODE** : アドレスモード (スレーブモード)
0 : 7ビットスレーブアドレス (10ビットアドレスは確認応答されません)
1 : 10ビットスレーブアドレス (7ビットアドレスは確認応答されません)

ビット 14 ソフトウェアによって常に 1 に保持してください。

ビット 13:10 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 **ADD[9:8]** : インタフェースアドレス
7ビットアドレスモード : 無視
10ビットアドレスモード : アドレスのビット 9:8

ビット 7:1 **ADD[7:1]** : インタフェースアドレス
アドレスのビット 7:1

ビット 0 **ADD0** : インタフェースアドレス
7ビットアドレスモード : 無視
10ビットアドレスモード : アドレスのビット 0

26.6.4 I²C 自己アドレスレジスタ 2 (I2C_OAR2)

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
予約済み								ADD2[7:1]							ENDUAL
								rw	rw	rw	rw	rw	rw	rw	rw

ビット 15:8 予約済み、リセット値に保持する必要があります。

ビット 7:1 **ADD2[7:1]** : インタフェースアドレス
デュアルアドレスモードでアドレスのビット 7:1

ビット 0 **ENDUAL** : デュアルアドレスモードイネーブル
0 : 7ビットアドレスモードで、OAR1 のみが認識されます。
1 : 7ビットアドレスモードで、OAR1 と OAR2 の両方が認識されます。



26.6.5 I²C データレジスタ (I2C_DR)

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
予約済み								DR[7:0]							
								rW	rW	rW	rW	rW	rW	rW	rW

ビット 15:8 予約済み、リセット値に保持する必要があります。

ビット 7:0 **DR[7:0]** : 8ビットデータレジスタ

- 受信されたバイト、またはバスに送信されるバイトです。
- トランスマッタモード : DR レジスタにバイトが書き込まれると、バイト送信が自動的に始まります。送信が始まり (TxE=1)、次に送信されるデータが DR に書き込まれると、連続的な送信ストリームを維持できます。
 - レシーバモード : 受信したバイトが DR にコピーされます (RxNE=1)。次のデータバイトを受信する前に DR が読み出されると (RxNE=1)、連続的な送信ストリームを維持できます。
- 注 :** スレーブモードでは、アドレスは DR にコピーされません。
- 注 :** 書き込み衝突は管理されません (TxE=0 の場合、DR は書き込み可能です)。
- 注 :** ACK パルスで ARLO イベントが発生した場合、受信したバイトは DR にコピーされないため、読み出しできません。

26.6.6 I²C ステータスレジスタ 1(I2C_SR1)

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
SMB ALERT	TIMEO UT	予約済み	PEC ERR	OVR	AF	ARLO	BERR	TxE	RxNE	予約済み	STOPF	ADD10	BTf	ADDR	SB
rc_w0	rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	r	r		r	r	r	r	r

ビット 15 SMBALERT : SMBus アラート

- SMBus ホストモードで、
 - 0 : SMBALERT はありません。
 - 1 : SMBALERT イベントがピンで発生しました。
- SMBus スレーブモードで、
 - 0 : SMBALERT 応答アドレスヘッダはありません。
 - 1 : SMBALERT LOW への SMBALERT 応答アドレスヘッダを受信しました。
- ソフトウェアによって 0 を書き込んでクリアするか、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 TIMEOUTタイムアウトまたはT ローエラー

- 0 : タイムアウトエラーは発生していません。
- 1 : SCL は 25 ms の間ローレベルを維持しました (タイムアウト)。
- または
- マスタのクロックロー累積延長時間が 10 ms を超えました (Tlow:mext)。
- または
- スレーブのクロックロー累積延長時間が 25 ms を超えました (Tlow:sext)。
- スレーブモードに設定した場合 : スレーブは通信をリセットし、ラインはハードウェアによって解放されます。
- マスタモードに設定した場合 : STOP コンディションがハードウェアによって送信されます。
- ソフトウェアによって 0 を書き込んでクリアするか、PE=0 のときにハードウェアによってクリアされます。

注 : この機能はSMBus モードでのみ使用できます。

ビット 13 予約済み、リセット値に保持する必要があります。

ビット 12 PECERR : 受信時の PEC エラー

- 0 : PEC エラーなし。レシーバは PEC 受信後に ACK を返します (ACK=1 の場合)。
- 1 : PEC エラーあり。レシーバは PEC 受信後に NACK を返します (ACK は任意)。
- ソフトウェアによって 0 を書き込んでクリアするか、PE=0 のときにハードウェアによってクリアされます。
- 注記 : 受信した CRC が正しくないとき、PEC 制御ビットが CRC 受信終了前にセットされなければ、PECERR はスレーブモードでセットされません。それでは、PEC 値を読み出すと受信した CRC が正しいか正しくないかが判断されます。



ビット 11 OVR : オーバーラン/アンダーラン

0 : オーバーラン/アンダーランはありません。

1 : オーバーランまたはアンダーランがあります。

– NOSTRETCH=1 の場合、ハードウェアによってスレーブモードでセットされます。

– 新しいバイト (ACK パルスを含む) を受信し、DR レジスタがまだ読み出されていないときは受信となります。新しく受信したバイトは失われます。

– 新しいバイトを送信すべきときに、DR レジスタへの書き込みがまだ行われていないときは送信となります。同じバイトが 2 回送信されます。

– ソフトウェアによって 0 を書き込んでクリアするか、PE=0 のときにハードウェアによってクリアされます。

注： DR の書き込みが SCL の立ち上がりエッジのごく近くで行われた場合、送信されるデータは不定で、ホールドタイミングエラーが発生します。

ビット 10 AF : 確認応答障害

0 : 確認応答障害はありません。

1 : 確認応答障害があります。

– 確認応答が返されないとき、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 0 を書き込んでクリアするか、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 9 ARLO : アービトレーションロスト (マスタモード)

0 : アービトレーションロストは検出されていません。

1 : アービトレーションロストが検出されました。

インタフェースがバスのアービトレーションを他のマスタに明け渡したとき、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 0 を書き込んでクリアするか、PE=0 のときにハードウェアによってクリアされます。

ARLO イベントの後、インタフェースは自動的にスレーブモード (MSL=0) に切り替わります。

注： SMBUS では、スレーブモードでのデータのアービトレーションは、データフェーズ中または確認応答の送信中にのみ行われます (アドレスの確認応答では行われません)。

ビット 8 BERR : バスエラー

0 : START または STOP コンディションの誤配置はありません。

1 : START または STOP コンディションが誤配置されました。

– バイト転送中に SCL がハイである間、SDA の立ち上がりまたは立ち下がりエッジが有効でない位置に発生していることをインタフェースが検出したとき、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 0 を書き込んでクリアするか、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 7 TxE : データレジスタエンプティ (トランスミッタ)

0 : データレジスタは空ではありません。

1 : データレジスタは空です。

– 送信中に DR が空になったときにセットされます。TxE はアドレスフェーズではセットされません。

– DR レジスタへのソフトウェア書き込みによってクリアされるか、START/STOP コンディションの後または PE=0 のときにハードウェアによってクリアされます。

NACK が受信された場合、または次に送信されるバイトが PEC である場合 (PEC=1)、TxE はセットされません。

注： 送信される最初のデータを書き込んだり、BTF がセットされているときにデータを書き込んだりしても、TxE はクリアされません。いずれの場合も、データレジスタはまだ空だからです。

ビット 6 **RxNE** : データレジスタノットエンプティ (レシーバ)

0 : データレジスタは空です。

1 : データレジスタは空ではありません。

– レシーバモードで、データレジスタが空でないときにセットされます。RxNE は、アドレスフェーズではセットされません。

– ソフトウェアで DR レジスタへの読み書きを行ってクリアするか、PE=0 のときにハードウェアによってクリアされます。

ARLO イベントの場合、RxNE はセットされません。

注: **BTF がセットされているとき、データレジスタはまだ満杯であるため、データを読み出しても RxNE はクリアされません。**

ビット 5 予約済み、リセット値に保持する必要があります。

ビット 4 **STOPF** : STOP 検出 (スレーブモード)

0 : STOP コンディションは検出されていません。

1 : STOP コンディションが検出されました。

– 確認応答後に、スレーブがバス上で STOP コンディションを検出したとき、ハードウェアによってセットされます (ACK=1 の場合)。

– ソフトウェアで SR1 レジスタを読み出し、続いて CR1 レジスタに書き込むとクリアされます。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。

注: **STOPF ビットは、NACK 受信後にはセットされません。**

STOPF がセットされた後、完全なクリアシーケンス (READ SR1、続いて WRITE CR1) を実行することを推奨します。図 211 : スレーブレシーバの転送シーケンス図 (685 ページ) を参照してください。

ビット 3 **ADD10** : 10ビットヘッダの送信 (マスターモード)

0 : ADD10 イベントは発生していません。

1 : マスタは最初のアドレスバイト (ヘッダ) を送信しました。

– マスタが 10ビットアドレスモードで最初のバイトを送信すると、ハードウェアによってセットされます。

– ソフトウェアで SR1 レジスタを読み出し、続いて DR レジスタに 2 番目のアドレスバイトを書き込むとクリアされます。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。

注: **ADD10 ビットは、NACK 受信後にはセットされません。**

ビット 2 BTF : バイト転送終了

- 0 : データバイト転送は完了していません。
- 1 : データバイト転送が正常に終了しました。
- NOSTRETCH=0 の場合、ハードウェアによってセットされます。
- 新しいバイト (ACK パルスを含む) を受信し、DR レジスタがまだ読み出されていないときは受信となります (RxNE=1)。
- 新しいバイトを送信すべきときに、DR レジスタへの書き込みがまだ行われていないときは送信となります (TxNE=1)。
- ソフトウェアで DR レジスタへの読み書きを行ったときにクリアされます。または、送信時の START/STOP コンディションの後や PE=0 のときに、ハードウェアによってクリアされます。

注: BTF ビットは、NACK 受信後にはセットされません。

BTF ビットは、次に送信されるバイトが PEC である場合 (I2C_SR2 レジスタの TRA=1 かつ I2C_CR1 レジスタの PEC=1)、セットされません。

ビット 1 ADDR : アドレス送信 (マスタモード) / アドレス一致 (スレーブモード)

このビットは、ソフトウェアで SR1 レジスタを読み出し、続いて SR2 レジスタを読み出すことでクリアされます。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。

- アドレス一致 (スレーブ)
- 0 : アドレスは受信されていないか、一致しませんでした。
 - 1 : 受信したアドレスが一致しました。
 - 受信したスレーブアドレスが OAR レジスタの内容と一致したとき、あるいは同報 / SMBus デバイスデフォルトアドレスまたは SMBus ホスト / SMBus アラートが認識されたとき、ハードウェアによってセットされます (設定に応じて、それぞれ有効である場合)。

注: スレーブモードでは、ADDR がセットされた後、完全なクリアシーケンス (READ SR1、続いて READ SR2) を実行することを推奨します。図 211 : スレーブレシーバの転送シーケンス図 (685 ページ) を参照してください。

- アドレス送信 (マスタ)
- 0 : アドレス送信は終了していません。
 - 1 : アドレス送信は終了しました。
 - 10ビットアドレス指定では、2 番目のバイトの ACK 後にビットがセットされます。
 - 7ビットアドレス指定では、バイトの ACK 後にビットがセットされます。

注: ADDR は、NACK 受信後にはセットされません。

ビット 0 SB : START ビット (マスタモード)

- 0 : START コンディションはありません。
- 1 : START コンディションが生成されました。
- START コンディションが生成されたときにセットされます。
- ソフトウェアで SR1 レジスタを読み出し、続いて DR レジスタへの書き込みを行うとクリアされます。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。

26.6.7 I²C ステータスレジスタ 2(I2C_SR2)

アドレスオフセット : 0x18
リセット値 : 0x0000

注 : I2C_SR1 の読み出し後に ADDR フラグがセットされた場合でも、I2C_SR1 の読み出し後に I2C_SR2 を読み出すと、ADDR フラグはクリアされます。したがって、I2C_SR2 は、I2C_SR1 で ADDR がセットされていると分かったとき、または STOPF ビットがクリアされているときにのみ読み出す必要があります。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PEC[7:0]								DUALF	SMB HOST	SMBDEF AULT	GEN CALL	予約済み	TRA	BUSY	MSL
r	r	r	r	r	r	r	r	r	r	r	r		r	r	r

- ビット 15:8 **PEC[7:0]** : パケットエラーチェックレジスタ
ENPEC=1 のとき、このレジスタは内部 PEC を含みます。
- ビット 7 **DUALF** : デュアルフラグ (スレーブモード)
0 : 受信アドレスが OAR1 と一致します。
1 : 受信アドレスが OAR2 と一致します。
– STOP コンディションまたは反復 START コンディションの後、あるいは 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 6 **SMBHOST** : SMBus ホストヘッダ (スレーブモード)
0 : SMBus ホストアドレスは受信されていません。
1 : SMBTYPE=1 かつ ENARP=1 のとき、SMBus ホストアドレスが受信されました。
– STOP コンディションまたは反復 START コンディションの後、あるいは 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 5 **SMBDEFAULT** : SMBus デバイスデフォルトアドレス (スレーブモード)
0 : SMBus デバイスデフォルトアドレスを受信していません。
1 : ENARP=1 のとき、SMBus デバイスデフォルトアドレスを受信しました。
– STOP コンディションまたは反復 START コンディションの後、あるいは 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 4 **GENCALL** : 同報アドレス (スレーブモード)
0 : 同報アドレスを受信していません。
1 : ENGC=1 のとき、同報アドレスを受信しました。
– STOP コンディションまたは反復 START コンディションの後、あるいは 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 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 **TRA** : トランスミッタ/レシーバ

0 : データバイトを受信しました。

1 : データバイトが送信されました。

このビットは、全アドレスフェーズの最後に、アドレスバイトの R/W ビットに応じてセットされます。

また、STOP コンディション (STOPF=1)、反復 START コンディション、バスアービトラーションロスト (ARLO=1) の検出後、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 1 **BUSY** : バスビジー

0 : バス上で通信は行われていません。

1 : バス上で通信が行われています。

– SDA または SCL のローレベル検出時に、ハードウェアによってセットされます。

– STOP コンディションの検出時に、ハードウェアによってクリアされます。

バス上で通信が実施されていることを示します。この情報は、インタフェースを無効にしても (PE=0) 更新されます。

ビット 0 **MSL** : マスタ/スレーブ

0 : スレーブモード

1 : マスタモード

– インタフェースがマスタモードになると (SB=1)、ハードウェアによってセットされます。

– バス上での STOP コンディションまたはアービトラーションロスト (ARLO=1) を検出した後、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。

注 : *I2C_SR1 の読み出し後に ADDR フラグがセットされた場合でも、I2C_SR1 の読み出し後に I2C_SR2 を読み出すと、ADDR フラグはクリアされます。したがって、I2C_SR2 は、I2C_SR1 で ADDR がセットされていると分かったとき、または STOPF ビットがクリアされているときにのみ読み出す必要があります。*

26.6.8 I²C クロック制御レジスタ (I2C_CCR)

アドレスオフセット : 0x1C

リセット値 : 0x0000

注 : 標準モード I²C 周波数を達成するには、 f_{PCLK1} が 2 MHz 以上である必要があります。高速モード I²C 周波数を達成するには、4 MHz 以上である必要があります。400 kHz の最大 I²C 高速モードクロックに到達するには、10 MHz の倍数である必要があります。

CCR レジスタは、I2C が無効 (PE=0) の場合にのみ設定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
F/S	DUTY	予約済み		CCR[11:0]											
rw	rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 15 **F/S** : I2C マスタモード選択

0 : 標準モード I2C

1 : 高速モード I2C

ビット 14 **DUTY** : 高速モードデューティサイクル

0 : 高速モード $t_{low}/t_{high} = 2$

1 : 高速モード $t_{low}/t_{high} = 16/9$ (CCR を参照)

ビット 13:12 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 **CCR[11:0]** : 高速/標準モード (マスタモード) でのクロック制御レジスタ

マスタモードで SCL クロックを制御します。

標準モードまたは SMBus :

$$T_{high} = CCR * T_{PCLK1}$$

$$T_{low} = CCR * T_{PCLK1}$$

高速モード :

DUTY = 0 の場合 :

$$T_{high} = CCR * T_{PCLK1}$$

$$T_{low} = 2 * CCR * T_{PCLK1}$$

DUTY = 1 の場合 : (400 kHz に到達するには)

$$T_{high} = 9 * CCR * T_{PCLK1}$$

$$T_{low} = 16 * CCR * T_{PCLK1}$$

例 : 標準モードで 100 kHz の SCL 周波数を生成するには、

FREQR = 08、 $T_{PCLK1} = 125$ ns の場合、CCR には 0x28 をプログラミングする必要があります。

$$(0x28 \Leftrightarrow 40d \times 125 \text{ ns} = 5000 \text{ ns})$$

注 : 最小許容値は 0x04 です。ただし、FAST DUTY モードでは最小許容値は 0x01 になります。

注 : $t_{high} = t_{r(SCL)} + t_{w(SCLH)}$ 。パラメータの定義については、デバイスのデータシートを参照してください。

$t_{low} = t_{f(SCL)} + t_{w(SCLL)}$ 。パラメータの定義については、デバイスのデータシートを参照してください。

I2C 通信速度は f_{SCL} から $1/(t_{high} + t_{low})$ です。アナログフィルタ入力遅延によって、実際の周波数は異なることがあります。

注 : CCR レジスタは、I²C が無効 (PE=0) の場合にのみ設定する必要があります。

26.6.9 I²C TRISE レジスタ (I2C_TRISE)

アドレスオフセット : 0x20
リセット値 : 0x0002

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み										TRISE[5:0]					
										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 5:0 **TRISE[5:0]** : 高速/標準モード（マスタモード）における最大立ち上がり時間

これらのビットは、マスタモードでの SCL フィードバックループの最大継続時間を提供します。目的は、SCL 立ち上がりエッジの継続時間にかかわらず、安定した SCL 周波数を維持することです。これらのビットには、I²C バス仕様で定められている SCL 最大立ち上がり時間より 1 だけ大きな値をプログラミングする必要があります。

例：標準モードで許容される SCL の最大立ち上がり時間は 1000 ns です。
I2C_CR2 レジスタの **FREQ[5:0]** ビットの値が 0x08 で、**T_{PCLK1}** = 125 ns である場合、**TRISE[5:0]** ビットには 09h をプログラミングする必要があります。
(1000 ns / 125 ns = 8 + 1)
フィルタ値も **TRISE[5:0]** に加算することができます。
結果が整数でない場合、**t_{HIGH}** パラメータを考慮して、その整数部を **TRISE[5:0]** にプログラミングする必要があります。

注： **TRISE[5:0]** の設定は、I2C が無効なとき (PE=0) にのみ行う必要があります。

26.6.10 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。

レジスタ境界アドレステーブルについては、[表 5 \(47 ページ\)](#) を参照してください。

表 139. 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		
0x00	I2C_CR1	予約済み																SWRST	予約済み	ALERT	PEC	POS	ACK	STOP	START	NOSTRETCH	ENGCG	ENPEC	ENARP	SMBTYPE	予約済み	SMBUS	PE		
	リセット値																	0		0	0	0	0	0	0	0	0	0	0	0	0	0			
0x04	I2C_CR2	予約済み																				LAST	DMAEN	ITBUFEN	ITEVTEN	ITERREN	予約済み	FREQ[5:0]							
	リセット値																					0	0	0	0	0		0	0	0	0	0	0		
0x08	I2C_OAR1	予約済み																ADDMODE	予約済み				ADD[9:8]		ADD[7:1]					ADD0					
	リセット値																	0					0	0		0	0	0	0	0	0	0			
0x0C	I2C_OAR2	予約済み																								ADD2[7:1]					ENDUAL				
	リセット値																									0	0	0	0	0	0	0	0		
0x10	I2C_DR	予約済み																								DR[7:0]									
	リセット値																									0	0	0	0	0	0	0	0		
0x14	I2C_SR1	予約済み																SMBALERT	TIMEOUT	予約済み	PECERR	OVR	AF	ARLO	BERR	TxE	RxNE	予約済み	STOPF	ADD10	BTF	ADDR	SB		
	リセット値																	0	0		0	0	0	0	0	0	0	0		0	0	0	0	0	
0x18	I2C_SR2	予約済み																PEC[7:0]								DUALF	SMBHOST	SMBDEFAULT	GENCALL	予約済み	TRA	BUSY	MSL		
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0		0	0	0	
0x1C	I2C_CCR	予約済み																F/S	DUTY	予約済み	CCR[11:0]														
	リセット値																	0	0		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x20	I2C_TRISE	予約済み																								TRISE[5:0]									
	リセット値																									0	0	0	0	0	1	0			

27 USART (Universal synchronous asynchronous receiver transmitter)

27.1 USART の概要

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 Interconnection 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可能です。

マルチバッファ設定用の DMA を使用して、高速データ通信が可能です。

27.2 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、速度とクロックの柔軟な許容誤差を実現
- フラクショナルボーレートジェネレータシステム
 - － プログラマブルな送受信共通のボーレート：APB 周波数= 32MHzで、8 倍オーバーサンプリングのとき最大 4 Mbit/s
- プログラム可能なデータワード長（8 または 9 ビット）
- 設定可能なストップビット：1 個または 2 個のストップビットをサポート
- 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
- 同期送信用のトランスミッタクロック出力
- IrDA SIR エンコーダデコーダ
 - － 通常モード用に 3/16 ビット持続時間のサポート
- スマートカードエミュレーション機能
 - － スマートカードインタフェースは、ISO 7816-3 規格で定義された非同期プロトコルのスマートカードをサポート
 - － スマートカード動作用に 0.5、1.5 のストップビット
- 単線半二重通信
- DMA (Direct Memory Access) を使用した設定可能なマルチバッファ通信
 - － 集中型 DMA を使用して、送受信バイトを専用 SRAM にバッファリング
- トランスミッタとレシーバ用に別個のイネーブルビット
- 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
 - － 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 種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
- 10 種の割り込みソース（フラグ付き）
 - － CTS の変化
 - － LIN ブレーク検出
 - － 送信データレジスタエンプティ
 - － 送信完了
 - － 受信データレジスタフル

- アイドルラインの受信
- オーバーランエラー
- フレーミングエラー
- ノイズエラー
- パリティエラー
- マルチプロセッサ通信：アドレス不一致の場合、ミュートモードに遷移
- 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／アドレスマーク検出）
- 2種類のレシーバウェイクアップモード：アドレスビット（MSB、9番目のビット）、アイドルライン

27.3 USART 機能の説明

インタフェースは、3本のピンによって他のデバイスに外部接続されます（[図 215](#)を参照）。USART 双方向通信では、少なくとも受信データ入力（RX）と送信データ出力（TX）の2個のピンが必要です。

RX：受信データ入力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には、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。単線モードとスマートカードモードでは、この入出力を使用してデータを送受信します（USART レベルでは、続いてデータは SW_RX で受信されます）。

通常の USART モードでは、フレームは以下の部分から構成され、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
- START ビット
- データワード（8 または 9 ビット）、LSB ファースト
- フレームの完了を示す 0.5、1、1.5、2 のストップビット
- このインタフェースはフラクショナルボーレートジェネレータを使用します - 12 ビットの仮数部と 4 ビットの小数部
- ステータスレジスタ（USART_SR）
- データレジスタ（USART_DR）
- ボーレートレジスタ（USART_BRR）- 12 ビットの仮数部と 4 ビットの小数部
- ガードタイムレジスタ（USART_GTPR）- スマートカードモードの場合

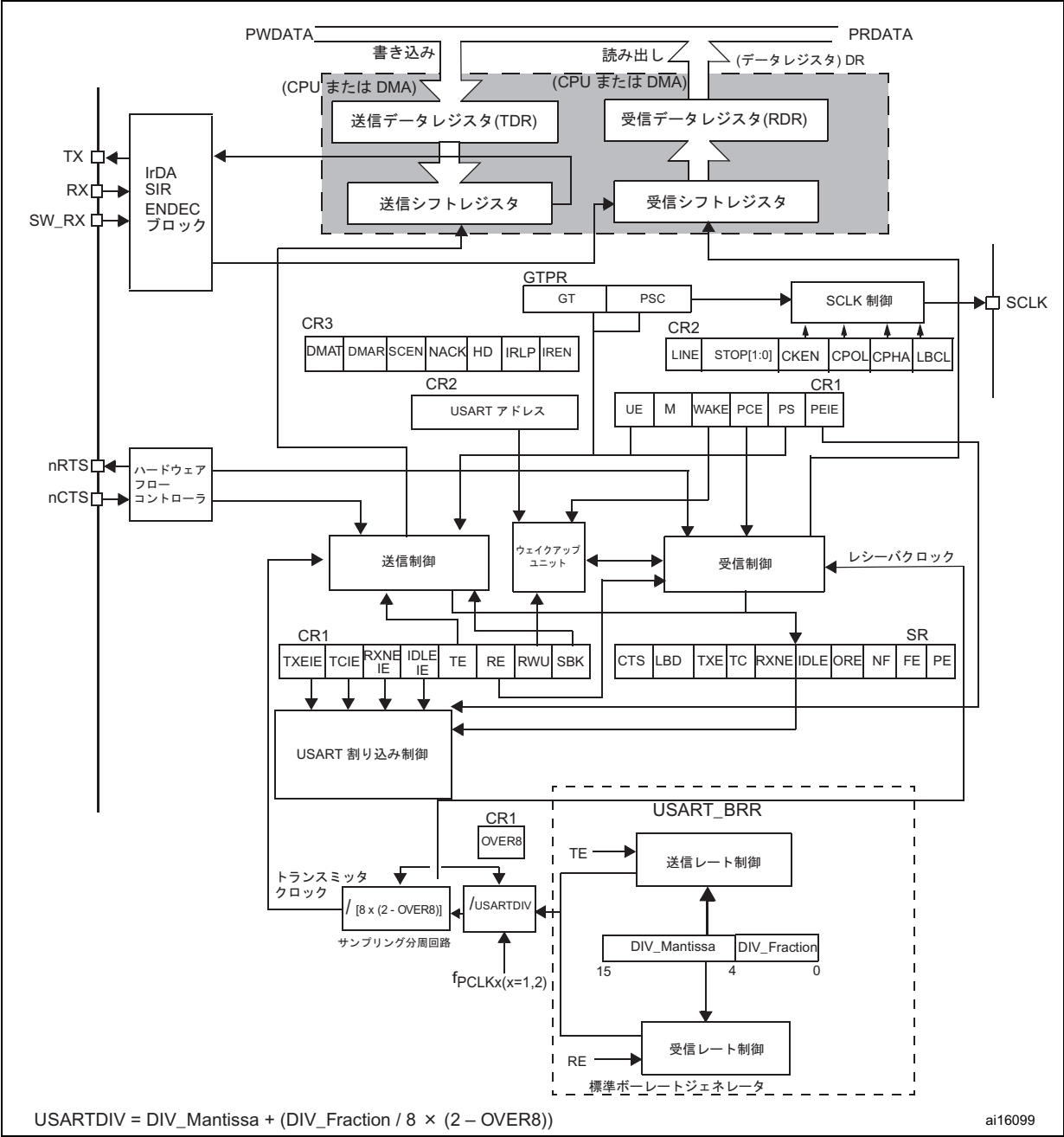
各ビットの定義については、[セクション 27.6：USART レジスタ \(757 ページ\)](#)を参照してください。

同期モードでのインタフェースには次のピンが必要になります。

- **SCLK**：トランスミッタクロック出力。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/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 上の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ミングできます。スマートカードモードでは、CK はスマートカードへのクロックを供給できます。

- ハードウェアフロー制御モードでは次のピンが必要になります。
- **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 がデータを受信できることを示します。

図 215. USART のブロック図



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。

27.3.1 USART キャラクタの説明

ワード長は、USART_CR1 レジスタの M ビットをプログラミングすることによって、8 ビットまたは 9 ビットとして選択できます（図 216 を参照）。

TX ピンは、スタートビットの処理中にはロー状態です。また、ストップビットの処理中にはハイ状態です。

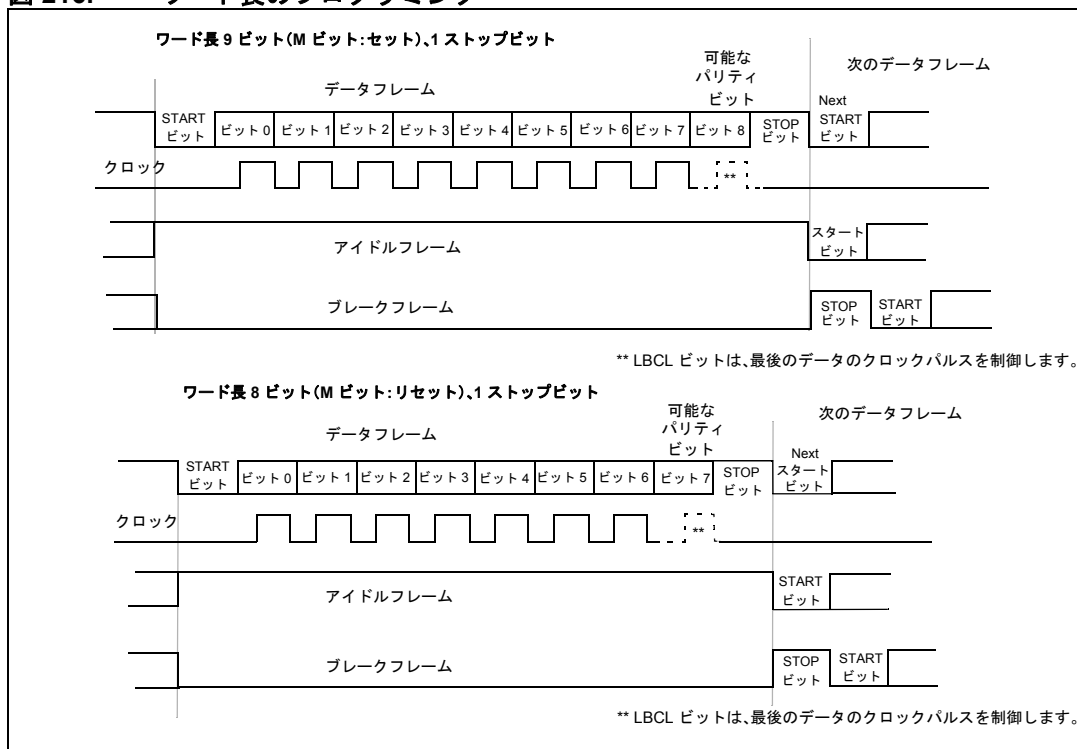
アイドルキャラクタは 1 フレーム全体が“1”と解釈され、この後にデータが格納された次のフレームのスタートビットが続きます（“1”の数にはストップビットの数も含まれます）。

ブレイクキャラクタは、フレーム周期中に“0”を受信することであると解釈されます。ブレイクフレームの終わりに、スタートビットを確認応答するために、トランスミッタは 1 個または 2 個のストップビット（論理“1”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イネーブル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 216. ワード長のプログラミング



27.3.2 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、8 または 9 ビットのデータワードを送信できます。送信イネーブルビット (TE) がセットされているとき、送信シフトレジスタ内のデータは TX ピンに出力され、対応するクロックパルスは CK ピンに出力されます。

キャラクタ送信

USART 送信時、データは LSB ファーストで TX ピンにシフトアウトされます。このモードでは、USART_DR 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 215 を参照)。

各キャラクタの前に START ビット (1 ビット周期の間、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によって終端されます。

USART では、0.5、1、1.5、および 2 のストップビットがサポートされます。

注： 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凍結されるため、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 ビットが有効にされた後、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

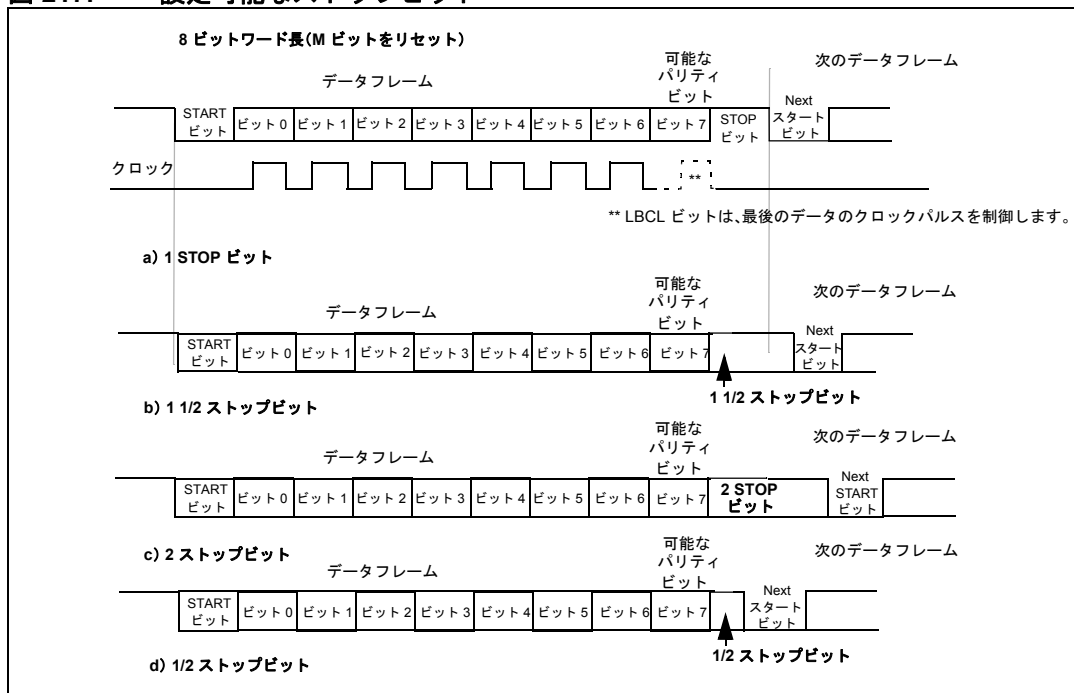
キャラクタとともに送信されるストップビットの数は、制御レジスタ 2 のビット 13、12 でプログラミングできます。

- **1 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 ストップビット：** 通常 USART モード、1 線式モード、モデムモードでサポートされます。
- **0.5 ストップビット：** スマートカードモードでのデータの受信に使用されます。
- **1.5 ストップビット：** スマートカードモードでのデータの送受信に使用されます。

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。

ブレーク送信では、10 個 (m = 0 の場合) または 11 個 (m = 1 の場合) のローレベルビットの後に、設定された数のストップビットが続きます。ローレベルビットの数が 10/11 個を超える長いブレークを送信することはできません。

図 217. 設定可能なストップビット



手順：

1. USART_CR1 レジスタの UE ビットに 1 を書き込むことによって、USART を有効にします。
2. USART_CR1 の M ビットをプログラミングして、ワード長を定義します。
3. USART_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. マルチバッファ通信を行う場合は、USART_CR3 レジスタの DMA イネーブル (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。
5. USART_BRR レジスタを使用して、希望するボーレートを選択します。
6. USART_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 USART_DR 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、送信されるデータごとにこれを繰り返します。
8. USART_DR 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 が無効にされたり HALT モードに入ったりする場合に必要な操作です。

1 バイト通信

TXE ビットのクリアは、常にデータレジスタへの書き込みによって行われます。

TXE ビットは、ハードウェアによってセットされ、次のことを示します。

- データが TDR からシフトレジスタに移動され、データ送信が開始されました。
- TDR レジスタは空です。
- 次のデータを、前のデータを上書きすることなく、USART_DR 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、データは、USART_DR レジスタへの書き込み命令によって 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、データは、USART_DR レジスタへの書き込み命令によってシフトレジスタに直接格納され、データ送信が開始され、TXE ビットがただちにセットされます。

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た場合、TC ビットはハイレベルになります。USART_CR1 レジスタの TCIE 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。

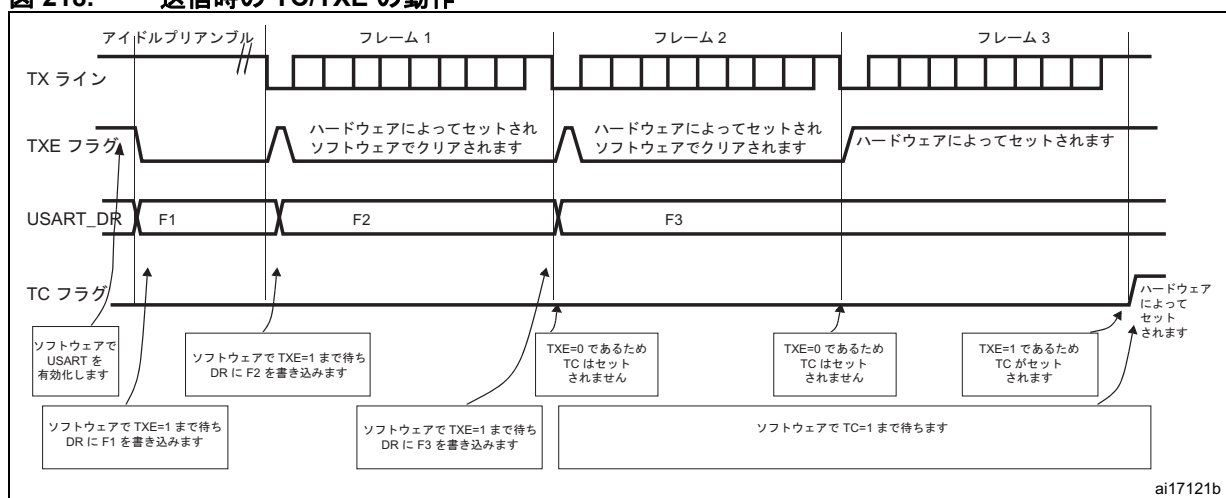
USART_DR レジスタに最後のデータを書き込んだ後、USART を無効にしたり、マイクロコントローラを低電力モードにするには、その前に TC=1 になるまで待つ必要があります（[図 218：送信時の TC/TXE の動作](#)を参照）。

TC ビットは、次のソフトウェアシーケンスによってクリアされます。

1. USART_SR レジスタからの読み出し
2. USART_DR レジスタへの書き込み

注： TC ビットは、“0”を書き込むことによってもクリアされます。このクリアシーケンスは、マルチバッファ通信の場合にのみ推奨されます。

図 218. 送信時の TC/TXE の動作



ブレークキャラクタ

SBK ビットをセットすると、ブレークキャラクタが送信されます。ブレークフレーム長は、M ビットに依存します（図 216 を参照）。

SBK ビットが“1”にセットされると、現在のキャラクタ送信が完了してから、TX ラインにブレークキャラクタが送信されます。このビットは、ブレークキャラクタが完了したとき（ブレークキャラクタのストップビット）、ハードウェアによってリセットされます。USART は、次のフレームの START ビットが正しく認識されるように、最後のブレークフレームの終わりに論理 1 ビットを挿入します。

注： ソフトウェアがブレーク送信の開始より前に SBK ビットをリセットした場合、ブレークキャラクタは送信されません。2 つの連続したブレークの場合、前のブレークのストップビットの後で SBK ビットをセットしてください。

アイドルキャラクタ

TE ビットをセットすると、USART 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。

27.3.3 レシーバ

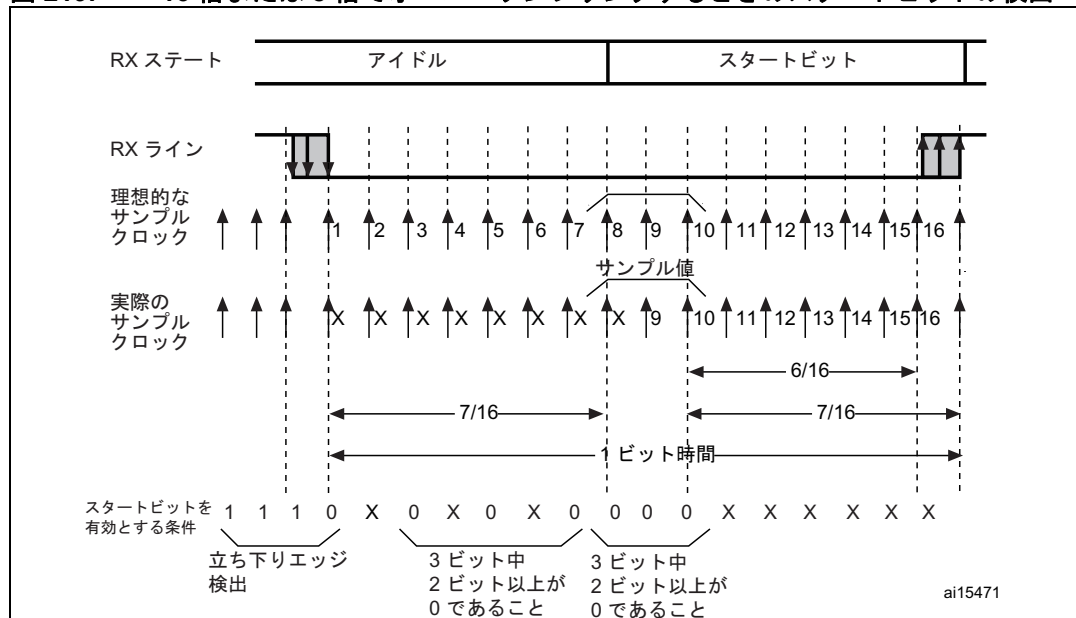
USART は、USART_CR1 レジスタの M ビットに応じて、8 ビットまたは 9 ビットのデータワードを受信できます。

START ビットの検出

16 倍または 8 倍でオーバーサンプリングするとき、スタートビットの検出シーケンスは同じです。

USART では、特定のサンプルシーケンスが認識されると、START ビットが検出されます。シーケンス：1 1 1 0 X 0 X 0 X 0 0 0 0。

図 219. 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)。

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)、両方のサンプリングで、サンプリングされた3つのビットのうち、少なくとも2つが0の場合 (第3、第5、および第7ビットでのサンプリングと、第8、第9、および第10ビットでのサンプリング)、NE ノイズフラグがセットされます。この条件が満たされない場合、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)。

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、スタートビットは確認されますが、NE ノイズフラグビットがセットされます。

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、USART_DR 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。

手順：

1. USART_CR1 レジスタの UE ビットに 1 を書き込むことによって、USART を有効にします。
2. USART_CR1 の M ビットをプログラミングして、ワード長を定義します。
3. USART_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. マルチバッファ通信を行う場合には、USART_CR3 レジスタの DMA イネーブル (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。STEP 3
5. ボーレートレジスタ USART_BRR を使用して、希望するボーレートを選択します。
6. USART_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、START 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る場合、割り込みが生成されます。
- 受信時にフレームエラー、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。
- マルチバッファでは、RXNE ビットは、バイトが受信されるたびにセットされ、データレジスタへの 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、USART_DR レジスタへのソフトウェア読み出しによって行われます。RXNE フラグは、0 を書き込むことによってクリアできます。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。

注： データの受信中には、RE ビットをリセットしないでください。受信中に RE ビットが無効にされた場合、現在のバイトの受信は中止されます。

ブレークキャラクタ

ブレークキャラクタが受信されると、USART は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た場合、データ受信キャラクタと同じ手順があり、IDLEIE 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。

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が受信されると、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、バイトが受信されるたびにセットされます。次のデータが受信されたときに 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。USART_DR への読み出しが行われると、前回のデータが読み出されます。
- 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た場合、または EIE ビットと DMAR ビットの両方がセットされた場合は、割り込みが生成されます。
- ORE ビットは、USART_SR レジスタの読み出しと、それに続く USART_DR レジスタの読み出し動作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、有効な最後のデータは、受信レジスタ RDR に格納され、読み出しが可能です。
- RXNE=0 の場合、有効な最後のデータはすでに読み出されているため、RDR レジスタには読み出すべきデータがありません。このケースは、有効な最後のデータの RDR への読み込みと、新しい（そして失われた）データの受信が同時に行われた場合に発生します。また、読み出しシーケンス中 (USART_SR レジスタと USART_DR レジスタの読み出しアクセスの間) に新しいデータが受信された場合にも発生します。

適切なオーバーサンプリング方式の選択

レシーバは、有効な受信データとノイズを区別することによって、データ復旧用に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）。

オーバーサンプリング方式は、USART_CR1 レジスタの OVER8 ビットをプログラミング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 倍とすることができます（[図 220](#) および [図 221](#)）。

アプリケーションに応じて、

- 高い速度（最大 $f_{PCLK}/8$ ）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が減少します（[セクション 27.3.5：クロック偏差に対する USART レシーバの許容誤差 \(739 ページ\)](#) 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 $f_{PCLK}/16$ に制限されます。

論理レベルの評価方法を選択するには、USART_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 140 を参照) は、サンプリング中にグリッチが発生しているので、そのデータを除去します。
- ラインにノイズがないときは 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 27.3.5: クロック偏差に対する USART レシーバの許容誤差 (739 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 USART_DR レジスタへ転送されます。
- 1 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、USART_CR3 レジスタの EIE ビットがセットされている場合、割り込みが発行されます。

NF ビットは、USART_SR レジスタの読み出し動作と、それに続く USART_DR レジスタの読み出し動作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 220. データサンプリング (16 倍のオーバーサンプリング)



図 221. データサンプリング (8 倍のオーバーサンプリング)

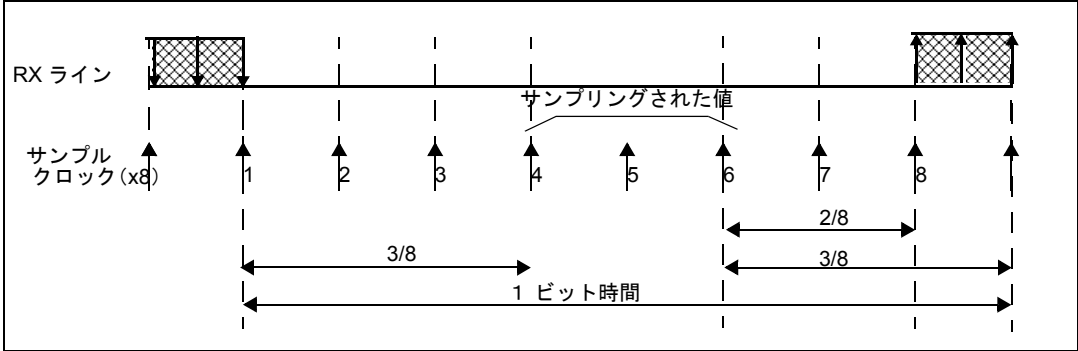


表 140. サンプリングされたデータからのノイズ検出

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。

同期外れ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：

- ハードウェアによって FE 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 USART_DR レジスタへ転送されます。
- 1 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、USART_CR3 レジスタの EIE ビットがセットされている場合、割り込みが発行されます。

FE ビットは、USART_SR レジスタの読み出し動作と、それに続く USART_DR レジスタの読み出し動作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、スマートカードモードでは 0.5 または 1.5 にできます。

1. **0.5 ストップビット (スマートカードモードでの受信)**: 0.5 ストップビットでは、サンプリングは行われません。したがって、0.5 ストップビットが選択されている場合、フレーミングエラーやブレークフレームの検出は行われません。
2. **1 ストップビット**: 1 ストップビットのサンプリングは、8 番目、9 番目、および 10 番目のサンプルで行われます。
3. **1.5 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には、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 (USART_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示される NACK 信号です)。その後、1.5 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。1.5 ストップビットのサンプリングは、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 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 27.3.11 : スマートカード \(748 ページ\)](#) を参照してください。
4. **2 ストップビット**: 2 ストップビットのサンプリングは、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。

27.3.4 フラクショナルボーレート生成

レシーバとトランスミッタ (Rx と Tx) のボーレートには、USARTDIV の仮数部と小数部にプログラミングされたものと同じ値が設定されます。

Equation 1 : 標準 USART のボーレート (SPI モードを含む)

$$Tx/Rx \text{ ボー} = \frac{f_{CK}}{8 \times (2 - \text{OVER8}) \times \text{USARTDIV}}$$

Equation 2 : スマートカード、LIN、および IrDA モードのボーレート

$$Tx/Rx \text{ ボー} = \frac{f_{CK}}{16 \times \text{USARTDIV}}$$

USARTDIV は、符号なしの固定小数点数であり、USART_BRR レジスタにコード化されます。

- OVER8=0 のとき、小数部は 4 ビットにコード化され、USART_BRR レジスタの DIV_fraction[3:0] ビットによってプログラミングされます。
- OVER8=1 のとき、小数部は 3 ビットにコード化され、USART_BRR レジスタの DIV_fraction[2:0] ビットによってプログラミングされます。ビット DIV_fraction[3] はクリアされた状態に保つ必要があります。

注 : ボーカウンタは、USART_BRR への書き込み後、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、通信中には、ボーレートレジスタの値を変更しないでください。

OVER8=0 のとき、USART_BRR レジスタの値から USARTDIV を得る方法**例 1:**

DIV_Mantissa = 0d27 かつ DIV_Fraction = 0d12 (USART_BRR = 0x1BC) の場合、

仮数部 (USARTDIV) = 0d27

小数部 (USARTDIV) = $12/16 = 0d0.75$

したがって、USARTDIV = 0d27.75

例 2:

USARTDIV = 0d25.62 をプログラミングするには

次のようにします。

$DIV\_Fraction = 16 * 0d0.62 = 0d9.92$

最も近い実数は 0d10 = 0xA

DIV_Mantissa = 仮数部 (0d25.620) = 0d25 = 0x19

USART_BRR = 0x19A であるため、USARTDIV = 0d25.625

例 3:

USARTDIV = 0d50.99 をプログラミングするには

次のようにします。

$DIV\_Fraction = 16 * 0d0.99 = 0d15.84$

最も近い実数は 0d16 = 0x10 => DIV_frac[3:0] のオーバーフロー => キャリーを仮数部に加算する必要があります。

DIV_Mantissa = 仮数部 (0d50.990 + キャリー) = 0d51 = 0x33

USART_BRR = 0x330 であるため、USARTDIV = 0d51.000

OVER8=1 のとき、USART_BRR レジスタの値から USARTDIV を得る方法**例 1 :**

DIV_Mantissa = 0x27 かつ DIV_Fraction[2:0] = 0d6 (USART_BRR = 0x1B6) の場合、

仮数部 (USARTDIV) = 0d27

小数部 (USARTDIV) = $6/8 = 0d0.75$

したがって、USARTDIV = 0d27.75

例 2:

USARTDIV = 0d25.62 をプログラミングするには

次のようにします。

$DIV\_Fraction = 8 * 0d0.62 = 0d4.96$

最も近い実数は 0d5 = 0x5

DIV_Mantissa = 仮数部 (0d25.620) = 0d25 = 0x19

これによって、USART_BRR = 0x195 => USARTDIV = 0d25.625

例 3:

USARTDIV = 0d50.99 をプログラミングするには

次のようにします。

$$\text{DIV\_Fraction} = 8 \times 0d0.99 = 0d7.92$$

最も近い実数は 0d8 = 0x8 => DIV_frac[2:0] のオーバーフロー => キャリーを仮数部に加算する必要があります。

$$\text{DIV\_Mantissa} = \text{仮数部} (0d50.990 + \text{キャリー}) = 0d51 = 0x33$$

これによって、USART_BRR = 0x0330 => USARTDIV = 0d51.000

**表 141. プログラミングされたボーレートの誤差計算 (f_{PCLK} = 8 MHz または f_{PCLK} = 12 MHz)、
16 倍のオーバーサンプリング⁽¹⁾**

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート 7		f _{PCLK} = 8 MHz			f _{PCLK} = 12 MHz		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 KBps	1.2 KBps	416.6875	0	1.2 KBps	625	0
2	2.4 KBps	2.4 KBps	208.3125	0.01	2.4 KBps	312.5	0
3	9.6 KBps	9.604 KBps	52.0625	0.04	9.6 KBps	78.125	0
4	19.2 KBps	19.185 KBps	26.0625	0.08	19.2 KBps	39.0625	0
5	38.4 KBps	38.462 KBps	13	0.16	38.339 KBps	19.5625	0.16
6	57.6 KBps	57.554 KBps	8.6875	0.08	57.692 KBps	13	0.16
7	115.2 KBps	115.942 KBps	4.3125	0.64	115.385 KBps	6.5	0.16
8	230.4 KBps	228.571 KBps	2.1875	0.79	230.769 KBps	3.25	0.16
9	460.8 KBps	470.588 KBps	1.0625	2.12	461.538 KBps	1.625	0.16
10	921.6 KBps	NA	NA	NA	NA	NA	NA
11	2 MBps	NA	NA	NA	NA	NA	NA
12	3 MBps	NA	NA	NA	NA	NA	NA

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。



表 142. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 8 \text{ MHz}$ または $f_{PCLK} = 12 \text{ MHz}$)、8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 8 \text{ MHz}$			$f_{PCLK} = 12 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 Kbps	1.2 Kbps	833.375	0	1.2 Kbps	1250	0
2	2.4 Kbps	2.4 Kbps	416.625	0.01	2.4 Kbps	625	0
3	9.6 Kbps	9.604 Kbps	104.125	0.04	9.6 Kbps	156.25	0
4	19.2 Kbps	19.185 Kbps	52.125	0.08	19.2 Kbps	78.125	0
5	38.4 Kbps	38.462 Kbps	26	0.16	38.339 Kbps	39.125	0.16
6	57.6 Kbps	57.554 Kbps	17.375	0.08	57.692 Kbps	26	0.16
7	115.2 Kbps	115.942 Kbps	8.625	0.64	115.385 Kbps	13	0.16
8	230.4 Kbps	228.571 Kbps	4.375	0.79	230.769 Kbps	6.5	0.16
9	460.8 Kbps	470.588 Kbps	2.125	2.12	461.538 Kbps	3.25	0.16
10	921.6 Kbps	888.889 Kbps	1.125	3.55	923.077 Kbps	1.625	0.16
11	2 Mbps	NA	NA	NA	NA	NA	NA
12	3 Mbps	NA	NA	NA	NA	NA	NA

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。

表 143. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 24 \text{ MHz}$)、16 倍のオーバーサンプリング⁽¹⁾

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 24 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 Kbps	1.2 Kbps	833.3125	0	1.2	1250	0
2	2.4 Kbps	2.4 Kbps	416.6875	0	2.4	625	0
3	9.6 Kbps	9.598 Kbps	104.1875	0.02	9.6	156.25	0
4	19.2 Kbps	19.208 Kbps	52.0625	0.04	19.2	78.125	0
5	38.4 Kbps	38.369 Kbps	26.0625	0.08	38.4	39.0625	0
6	57.6 Kbps	57.554 Kbps	17.375	0.08	57.554	26.0625	0.08
7	115.2 Kbps	115.108 Kbps	8.6875	0.08	115.385	13	0.16
8	230.4 Kbps	231.884 Kbps	4.3125	0.64	230.769	6.5	0.16
9	460.8 Kbps	457.143 Kbps	2.1875	0.79	461.538	3.25	0.16



表 143. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 24 \text{ MHz}$)、
16 倍のオーバーサンプリング⁽¹⁾ (続き)

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 24 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
10	921.6 KBps	941.176 KBps	1.0625	2.12	923.077	1.625	0.16
11	2 MBps	NA	NA	NA	NA	NA	NA
12	3 MBps	NA	NA	NA	NA	NA	NA

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。

表 144. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 24 \text{ MHz}$)、
8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 24 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 KBps	1.2 KBps	1666.625	0	1.2 KBps	2500	0
2	2.4 KBps	2.4 KBps	833.375	0	2.4 KBps	1250	0
3	9.6 KBps	9.598 KBps	208.375	0.02	9.6 KBps	312.5	0
4	19.2 KBps	19.208 KBps	104.125	0.04	19.2 KBps	156.25	0
5	38.4 KBps	38.369 KBps	52.125	0.08	38.4 KBps	78.125	0
6	57.6 KBps	57.554 KBps	34.75	0.08	57.554 KBps	52.125	0.08
7	115.2 KBps	115.108 KBps	17.375	0.08	115.385 KBps	26	0.16
8	230.4 KBps	231.884 KBps	8.625	0.64	230.769 KBps	13	0.16
9	460.8 KBps	457.143 KBps	4.375	0.79	461.538 KBps	6.5	0.16
10	921.6 KBps	941.176 KBps	2.125	2.12	923.077 KBps	3.25	0.16
11	2 MBps	2000 KBps	1	0	2000 KBps	1.5	0
12	3 MBps	NA	NA	NA	3000 KBps	1	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。

表 145. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、16 倍のオーバーサンプリング⁽¹⁾

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート		$f_{PCLK} = 1 \text{ MHz}$			$f_{PCLK} = 8 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
1	1.2 Kbps	1.2 Kbps	52.0625	0.04	1.2 Kbps	416.6875	0
2	2.4 Kbps	2.398 Kbps	26.0625	0.08	2.4 Kbps	208.3125	0.01
3	9.6 Kbps	9.615 Kbps	6.5	0.16	9.604 Kbps	52.0625	0.04
4	19.2 Kbps	19.231 Kbps	3.25	0.16	19.185 Kbps	26.0625	0.08
5	38.4 Kbps	38.462 Kbps	1.625	0.16	38.462 Kbps	13	0.16
6	57.6 Kbps	58.824 Kbps	1.0625	2.12	57.554 Kbps	8.6875	0.08
7	115.2 Kbps	NA	NA	NA	115.942 Kbps	4.3125	0.64
8	230.4 Kbps	NA	NA	NA	228.571 Kbps	2.1875	0.79
9	460.8 Kbps	NA	NA	NA	470.588 Kbps	1.0625	2.12
10	921.6 Kbps	NA	NA	NA	NA	NA	NA
11	2 Mbps	NA	NA	NA	NA	NA	NA
12	4 Mbps	NA	NA	NA	NA	NA	NA

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。

表 146. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 1 \text{ MHz}$			$f_{PCLK} = 8 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
1	1.2 Kbps	1.2 Kbps	104.125	0.04	1.2 Kbps	833.375	0
2	2.4 Kbps	2.398 Kbps	52.125	0.08	2.4 Kbps	416.625	0.01
3	9.6 Kbps	9.615 Kbps	13	-0.16	9.604 Kbps	104.125	0.04
4	19.2 Kbps	19.231 Kbps	6.5	0.16	19.185 Kbps	52.125	0.08
5	38.4 Kbps	38.462 Kbps	3.25	0.16	38.462 Kbps	26	0.16
6	57.6 Kbps	58.824 Kbps	2.125	2.12	57.554 Kbps	17.375	0.08
7	115.2 Kbps	111.111 Kbps	1.125	3.55	115.942 Kbps	8.625	0.64
8	230.4 Kbps	NA	NA	NA	228.571 Kbps	4.375	0.79
9	460.8 Kbps	NA	NA	NA	470.588 Kbps	2.125	2.12
10	921.6 Kbps	NA	NA	NA	888.889 Kbps	1.125	3.55



表 146. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、8 倍のオーバーサンプリング⁽¹⁾ (続き)

8 倍のオーバーサンプリング (OVER8=1)							
ボーレート		$f_{PCLK} = 1 \text{ MHz}$			$f_{PCLK} = 8 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
11	2 MBps	NA	NA	NA	NA	NA	NA
12	4 MBps	NA	NA	NA	NA	NA	NA

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。

表 147. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、16 倍のオーバーサンプリング⁽¹⁾

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 32 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
1	1.2 KBps	1.2 KBps	833.3125	0	1.2 KBps	1666.6875	0
2	2.4 KBps	2.4 KBps	416.6875	0	2.4 KBps	833.3125	0
3	9.6 KBps	9.598 KBps	104.1875	0.02	9.601 KBps	208.3125	0.01
4	19.2 KBps	19.208 KBps	52.0625	0.04	19.196 KBps	104.1875	0.02
5	38.4 KBps	38.369 KBps	26.0625	0.08	38.415 KBps	52.0625	0.04
6	57.6 KBps	57.554 KBps	17.375	0.08	57.554 KBps	34.75	0.08
7	115.2 KBps	115.108 KBps	8.6875	0.08	115.108 KBps	17.375	0.08
8	230.4 KBps	231.884 KBps	4.3125	0.64	230.216 KBps	8.6875	0.08
9	460.8 KBps	457.143 KBps	2.1875	0.79	463.768 KBps	4.3125	0.64
10	921.6 KBps	941.176 KBps	1.0625	2.12	914.286 KBps	2.1875	0.79
11	2 MBps	NA	NA	NA	2000 KBps	1	0
12	4 MBps	NA	NA	NA	NA	NA	NA

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。

表 148. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、
8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 32 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 Kbps	1.2 Kbps	1666.625	0	1.2 Kbps	3333.375	0
2	2.4 Kbps	2.4 Kbps	833.375	0	2.4 Kbps	1666.625	0
3	9.6 Kbps	9.598 Kbps	208.375	0.02	9.601 Kbps	416.625	0.01
4	19.2 Kbps	19.208 Kbps	104.125	0.04	19.196 Kbps	208.375	0.02
5	38.4 Kbps	38.369 Kbps	52.125	0.08	38.415 Kbps	104.125	0.04
6	57.6 Kbps	57.554 Kbps	34.75	0.08	57.554 Kbps	69.5	0.08
7	115.2 Kbps	115.108 Kbps	17.375	0.08	115.108 Kbps	34.75	0.08
8	230.4 Kbps	231.884 Kbps	8.625	0.64	230.216 Kbps	17.375	0.08
9	460.8 Kbps	457.143 Kbps	4.375	0.79	463.768 Kbps	8.625	0.64
10	921.6 Kbps	941.176 Kbps	2.125	2.12	914.286 Kbps	4.375	0.79
11	2 MBps	2000 Kbps	1	0	2000 Kbps	2	0
12	4 MBps	NA	NA	NA	4000 Kbps	1	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。

表 149. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、
8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 1 \text{ MHz}$			$f_{PCLK} = 8 \text{ MHz}$		
S.No	目標	実際	ボーレート レジスタに プログラミング された値	誤差 (%) = (計算値 - 目標値) B レート / 目標の B レート	実際	ボーレート レジスタに プログラミング された値	誤差
1	1.2 Kbps	1.2 Kbps	104.125	0.04	1.2 Kbps	833.375	0
2	2.4 Kbps	2.398 Kbps	52.125	0.08	2.4 Kbps	416.625	0.01
3	9.6 Kbps	9.615 Kbps	13	-0.16	9.604 Kbps	104.125	0.04
4	19.2 Kbps	19.231 Kbps	6.5	0.16	19.185 Kbps	52.125	0.08
5	38.4 Kbps	38.462 Kbps	3.25	0.16	38.462 Kbps	26	0.16
6	57.6 Kbps	58.824 Kbps	2.125	2.12	57.554 Kbps	17.375	0.08
7	115.2 Kbps	111.111 Kbps	1.125	3.55	115.942 Kbps	8.625	0.64
8	230.4 Kbps	NA	NA	NA	228.571 Kbps	4.375	0.79



表 149. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 1 \text{ MHz}$ または $f_{PCLK} = 8 \text{ MHz}$)、8 倍のオーバーサンプリング⁽¹⁾ (続き)

8 倍のオーバーサンプリング (OVER8=1)							
ボーレート		$f_{PCLK} = 1 \text{ MHz}$			$f_{PCLK} = 8 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
9	460.8 KBps	NA	NA	NA	470.588 KBps	2.125	2.12
10	921.6 KBps	NA	NA	NA	888.889 KBps	1.125	3.55
11	2 MBps	NA	NA	NA	NA	NA	NA
12	4 MBps	NA	NA	NA	NA	NA	NA

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。

表 150. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、16 倍のオーバーサンプリング⁽¹⁾

16 倍のオーバーサンプリング (OVER8=0)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 32 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
1	1.2 KBps	1.2 KBps	833.3125	0	1.2 KBps	1666.6875	0
2	2.4 KBps	2.4 KBps	416.6875	0	2.4 KBps	833.3125	0
3	9.6 KBps	9.598 KBps	104.1875	0.02	9.601 KBps	208.3125	0.01
4	19.2 KBps	19.208 KBps	52.0625	0.04	19.196 KBps	104.1875	0.02
5	38.4 KBps	38.369 KBps	26.0625	0.08	38.415 KBps	52.0625	0.04
6	57.6 KBps	57.554 KBps	17.375	0.08	57.554 KBps	34.75	0.08
7	115.2 KBps	115.108 KBps	8.6875	0.08	115.108 KBps	17.375	0.08
8	230.4 KBps	231.884 KBps	4.3125	0.64	230.216 KBps	8.6875	0.08
9	460.8 KBps	457.143 KBps	2.1875	0.79	463.768 KBps	4.3125	0.64
10	921.6 KBps	941.176 KBps	1.0625	2.12	914.286 KBps	2.1875	0.79
11	2 MBps	NA	NA	NA	2000 KBps	1	0
12	4 MBps	NA	NA	NA	NA	NA	NA

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。

表 151. プログラミングされたボーレートの誤差計算 ($f_{PCLK} = 16 \text{ MHz}$ または $f_{PCLK} = 32 \text{ MHz}$)、8 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 (OVER8=1)							
ボーレート		$f_{PCLK} = 16 \text{ MHz}$			$f_{PCLK} = 32 \text{ MHz}$		
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た値	誤差 (%) = (計算値 - 目標値) / 目標の B レート	実際	ボーレートレジスタにプログラミングされた値	誤差
1	1.2 Kbps	1.2 Kbps	1666.625	0	1.2 Kbps	3333.375	0
2	2.4 Kbps	2.4 Kbps	833.375	0	2.4 Kbps	1666.625	0
3	9.6 Kbps	9.598 Kbps	208.375	0.02	9.601 Kbps	416.625	0.01
4	19.2 Kbps	19.208 Kbps	104.125	0.04	19.196 Kbps	208.375	0.02
5	38.4 Kbps	38.369 Kbps	52.125	0.08	38.415 Kbps	104.125	0.04
6	57.6 Kbps	57.554 Kbps	34.75	0.08	57.554 Kbps	69.5	0.08
7	115.2 Kbps	115.108 Kbps	17.375	0.08	115.108 Kbps	34.75	0.08
8	230.4 Kbps	231.884 Kbps	8.625	0.64	230.216 Kbps	17.375	0.08
9	460.8 Kbps	457.143 Kbps	4.375	0.79	463.768 Kbps	8.625	0.64
10	921.6 Kbps	941.176 Kbps	2.125	2.12	914.286 Kbps	4.375	0.79
11	2 Mbps	2000 Kbps	1	0	2000 Kbps	2	0
12	4 Mbps	NA	NA	NA	4000 Kbps	1	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。

27.3.5 クロック偏差に対する USART レシーバの許容誤差

USART の非同期レシーバが正しく機能するのは、クロックシステムの合計偏差が USART レシーバの許容誤差を下回る場合に限りです。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トランシーバに起因)

$DTRA + DQUANT + DREC + DTCL < \text{USART レシーバの許容誤差}$

データを正しく受信するための USART レシーバの許容誤差は、許容される最大偏差と等しく、以下の選択に依存します。

- USART_CR1 レジスタの M ビットによって定義される 10/11 ビットのキャラクタ長
- USART_CR1 レジスタの OVER8 ビットによって定義される、8 倍または 16 倍のオーバーサンプリング
- フラクショナルボーレートを使用するかどうか
- 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 (USART_CR3 レジスタの ONEBIT ビットの値に依存)



表 152. USART レシーバの許容誤差 (DIV Fraction = 0 の場合)

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT=0	ONEBIT=1	ONEBIT=0	ONEBIT=1
0	3.75%	4.375%	2.50%	3.75%
1	3.41%	3.97%	2.27%	3.41%

表 153. USART レシーバの許容誤差 (DIV Fraction が 0 でない場合)

M ビット	OVER8 ビット = 0		OVER8 ビット = 1	
	ONEBIT=0	ONEBIT=1	ONEBIT=0	ONEBIT=1
0	3.33%	3.88%	2%	3%
1	3.03%	3.53%	1.82%	2.73%

注: 表 152 と表 153 に示されている数値は、受信したフレームが正確に 10 ビット時間 (M=0 の場合) または 11 ビット時間 (M=1 の場合) のアイドルフレームを含んでいる場合、若干異なる場合があります。

27.3.6 マルチプロセッサ通信

複数の USART をネットワークに接続することによって、マルチプロセッサ通信を行える可能性があります。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、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い場合が多く、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置かれます。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。
- すべての受信割り込みは禁止されます。
- USART_CR1 レジスタの RWU ビットが 1 にセットされます。RWU ビットは、ハードウェアによって自動的に制御されますが、特定の条件下ではソフトウェアによる書き込みが可能です。

USART は、USART_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の出入りを制御できます。

- 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

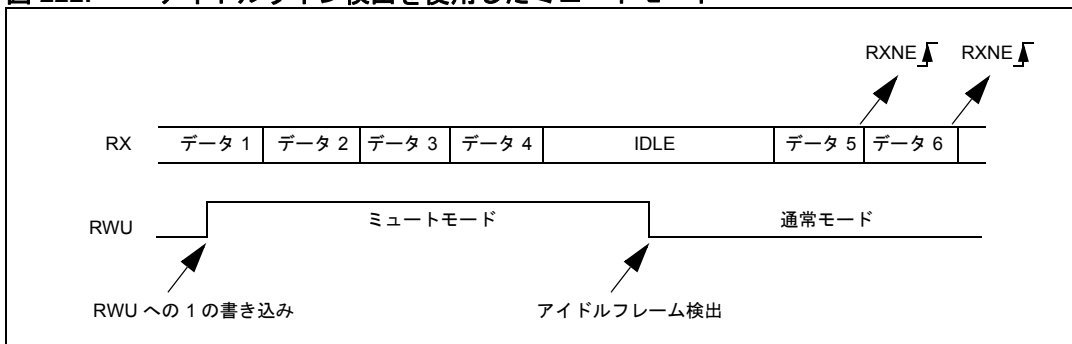
アイドルライン検出 (WAKE=0)

RWU ビットに 1 が書き込まれると、USART はミュートモードに入ります。

USART は、アイドルフレームの検出によって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、USART_SR レジスタの IDLE ビットはセットされません。RWU ビットには、ソフトウェアによって 0 を書き込むこともできます。

図 222 に、アイドルライン検出を使用したミュートモード動作の例を示します。

図 222. アイドルライン検出を使用したミュートモード



アドレスマーク検出 (WAKE=1)

このモードでは、MSB が“1”のバイトはアドレスとして、そうでないバイトはデータとして、それぞれ認識されます。アドレスバイトでは、ターゲットレシーバのアドレスは下位 4 ビットに置かれます。レシーバは、この 4 ビットワードを、USART_CR2 レジスタの ADD ビットにプログラミングされている自分のアドレスと比較します。

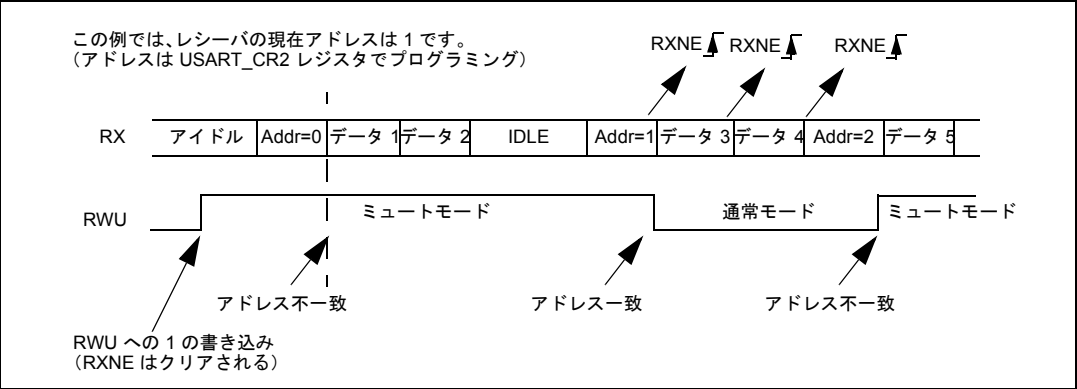
プログラミングされたアドレスと一致しないアドレスキャラクタが受信されると、USART はミュートモードに入ります。この場合、RWU ビットはハードウェアによってセットされます。USART はミュートモードに入っているため、このアドレスバイトに対して RXNE フラグはセットされず、割り込みも DMA リクエストも発行されません。

プログラミングされたアドレスと一致するアドレスキャラクタが受信されると、USART はミュートモードから抜けます。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ため、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。

レシーババッファにデータが存在しない場合 (USART_SR レジスタの RXNE=0)、RWU ビットに 0 または 1 を書き込むことができます。そうでない場合、書き込みは無視されます。

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 223 に示します。

図 223. アドレスマーク検出を使用したミュートモード



27.3.7 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、USART_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るフレーム長に応じて、予想される USART フレームフォーマットを表 154 に示します。

表 154. フレームフォーマット

M ビット	PCE ビット	USART フレーム ⁽¹⁾
0	0	SB 8 ビットデータ STB
0	1	SB 7 ビットデータ PB STB
1	0	SB 9 ビットデータ STB
1	1	SB 8 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

偶数パリティ

下位 7 ビットまたは 8 ビット (M=0 または M=1 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”のビットが偶数になるように、パリティビットが計算されます。

例：データ = 00110101、4 ビットがセット => 偶数パリティが選択された場合 (USART_CR1 の PS=0)、パリティビットは 0 になります。

奇数パリティ

下位 7 ビットまたは 8 ビット (M=0 または M=1 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”のビットが奇数になるように、パリティビットが計算されます。

例：データ = 00110101、4 ビットがセット => 奇数パリティが選択された場合 (USART_CR1 の PS=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

パリティ検査に失敗した場合、USART_SR レジスタの PE フラグがセットされ、USART_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 フラグはソフトウェアシーケンスによってクリアされます(ステータスレジスタの読み出しと、それに続く USART_DR データレジスタへの読み出しまたは書き込みアクセス)。

注： アドレスマークによるウェイクアップの場合、データの最上位ビットを使ってアドレスをパリティビットから区別します。そしてレシーバは、アドレスデータのパリティをチェックしません(パリティエラーの場合、PE はセットされません)。

送信中のパリティ生成

USART_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”)。

注： 送信を管理するソフトウェアルーチンは、PE フラグをクリアするソフトウェアシーケンス(ステータスレジスタの読み出しに続くデータレジスタへの読み出しまたは書き込みアクセス)を起動できます。半二重モードで動作するとき、ソフトウェアに依存しますが、これによって PE フラグが不意にクリアされることがあります。

27.3.8 LIN (local interconnection network) モード

LIN モードを選択するには、USART_CR2 レジスタの LINEN ビットをセットします。LIN モードでは、次のビットは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 STOP[1:0] と CLKEN
- USART_CR3 レジスタの SCEN、HDSEL、IREN

LIN 送信

LIN マスタの送信では、通常の USART 送信の手順ではなく、[セクション 27.3.2](#) で説明したものと同一手順を適用する必要がありますが、次の点が異なります。

- 8 ビットのワード長を設定するには M ビットをクリアします。
- LIN モードに入るには LINEN ビットをセットします。この場合、SBK ビットをセットすると、13 個の“0”ビットがブレークキャラクタとして送信されます。その後、値“1”のビットが送信され、次の START 検出が可能になります。

LIN 受信

ブレーク検出回路が USART インタフェースに内蔵されています。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、ブレークは発生の際に検出できます。

レシーバが有効になると(USART_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(USART_CR2 レジスタの LBDL=0)または 11 個(USART_CR2 レジスタの LBDL=1)の連続した“0”ビットが検出され、その後にデリミタキャラクタが続く場合、USART_SR レジスタの LBD 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 10/11 個が発生する前に“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、レシーバはすぐに停止し、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 224: LIN モードでのブレーク検出 \(11 ビットブレーク長、LBDL=1\)](#) (744 ページ) に示します。

ブレークフレームの例を [図 225: LIN モードでのブレーク検出とフレーミングエラー検出 \(745 ページ\)](#) に示します。

図 224. LIN モードでのブレーク検出 (11 ビットブレーク長、LBDL=1)

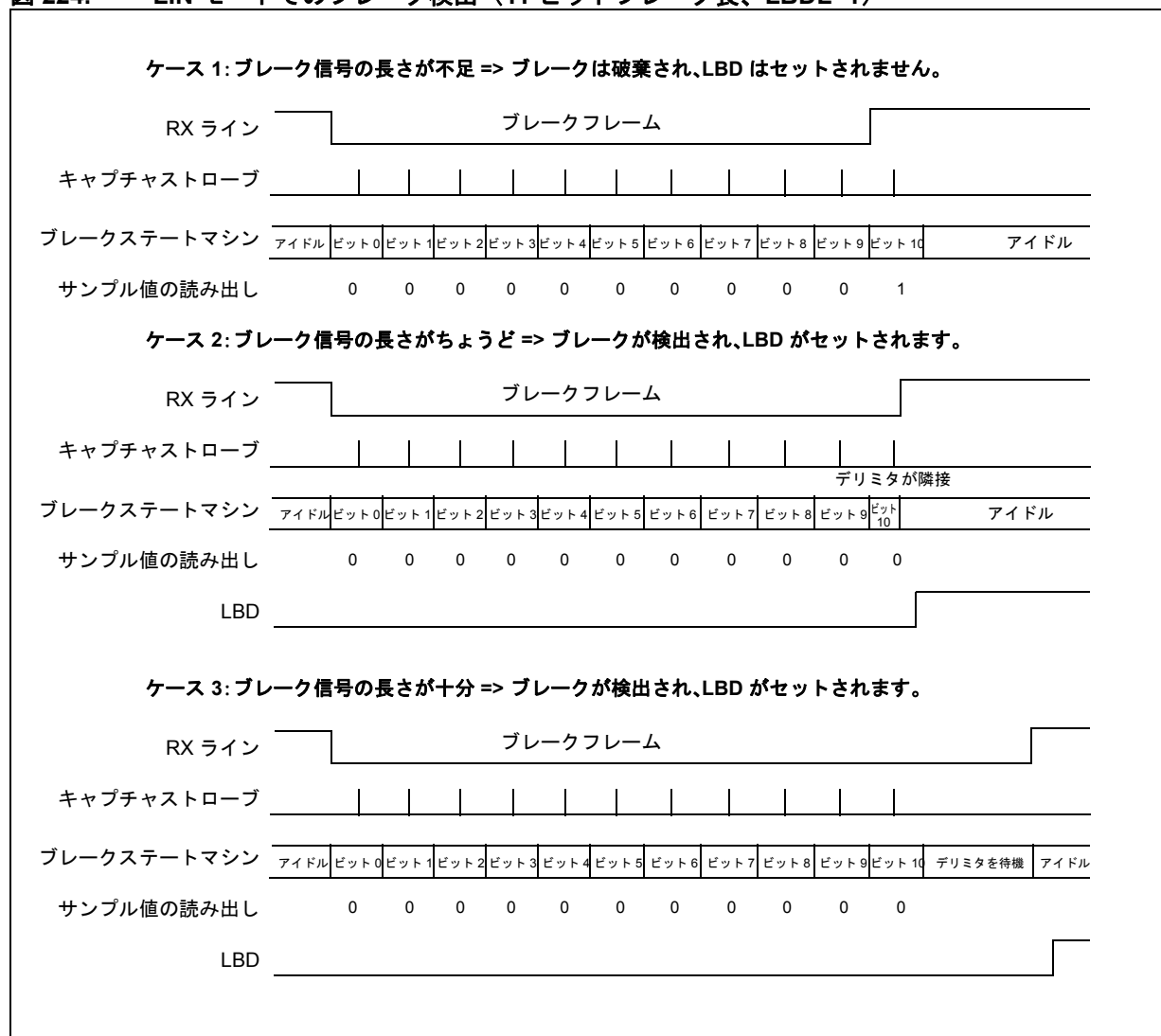
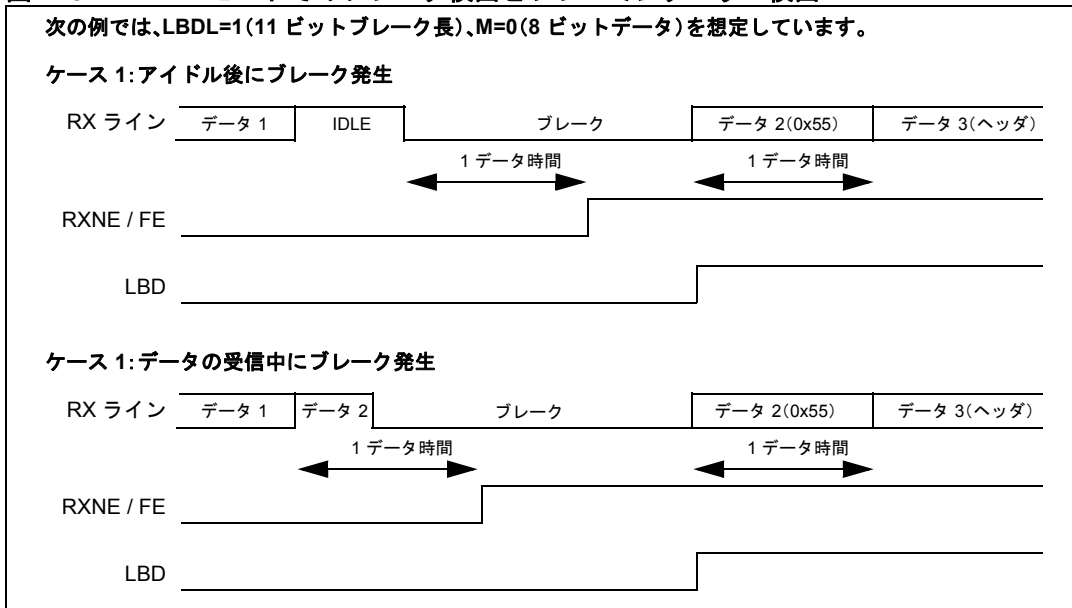


図 225. LIN モードでのブレーク検出とフレーミングエラー検出



27.3.9 USART 同期モード

同期モードを選択するには、USART_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、次のビットは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 SCEN、HDSEL、および IREN ビット

USART では、マスタモードで双方向の同期シリアル通信を制御できます。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。USART_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。USART_CR2 レジスタの CPOL ビットによってクロック極性を選択でき、同じレジスタの CPHA ビットによって外部クロックの位相を選択できます（図 226、図 227、および図 228 を参照）。

アイドル時、プリアンブル処理時、およびブレーク送信時には、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 に応じて立ち上がりまたは立ち下がりエッジ）でサンプリングされます。このとき、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 USART_DR 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。

トランスミッタとレシーバが無効にされた場合 (TE=RE=0)、クロックパルスが正常に機能するには、LBCL、CPOL、および CPHA の各ビットを選択する必要があります。これらのビットは、トランスミッタまたはレシーバが有効である間は、変更しないでください。

レシーバのセットアップ時間とホールド時間を最小限に抑えるには、TE ビットと RE ビットを同じ命令でセットすることをお勧めします。

USART は、マスタモードのみをサポートし、入力クロックに関連するデータを送受信することはできません (SCLK は常に出力です)。

図 226. USART の同期送信の例

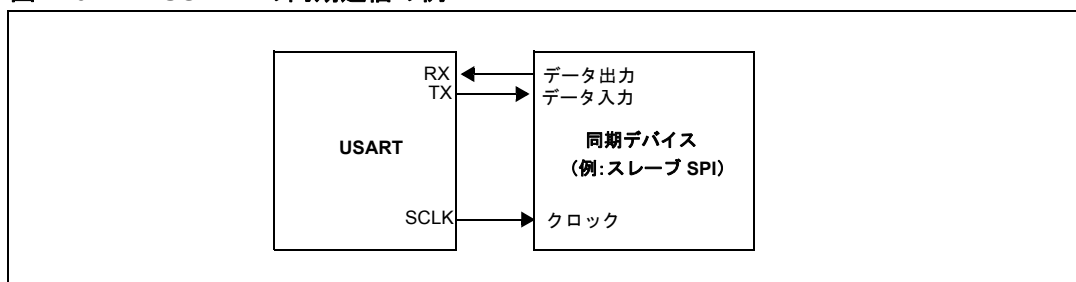


図 227. USART データクロックタイミング図 (M=0)

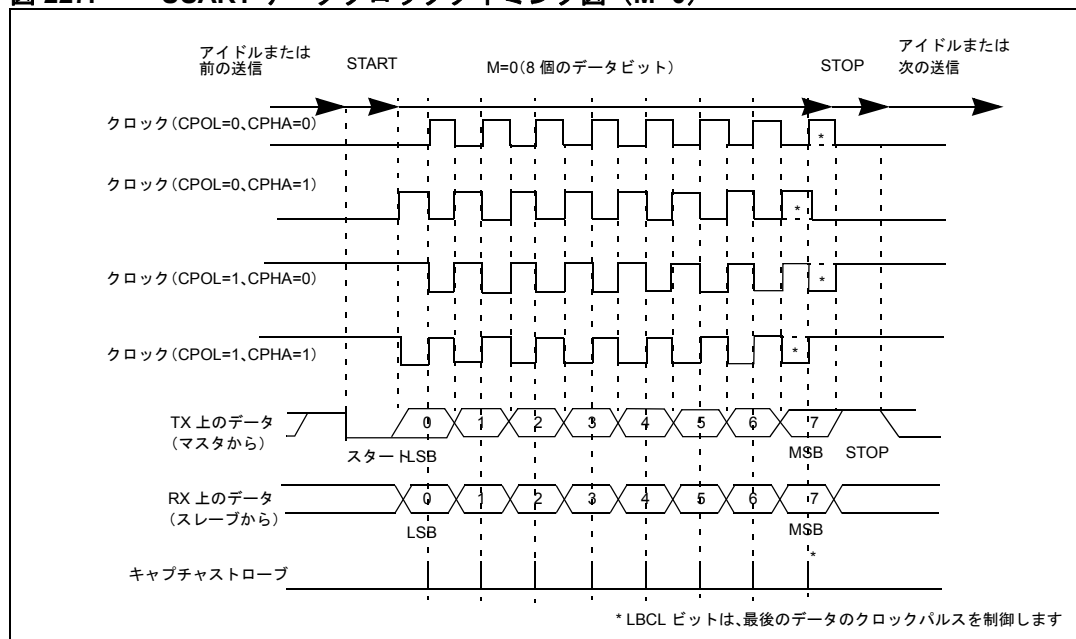


図 228. USART データクロックタイミング図 (M=1)

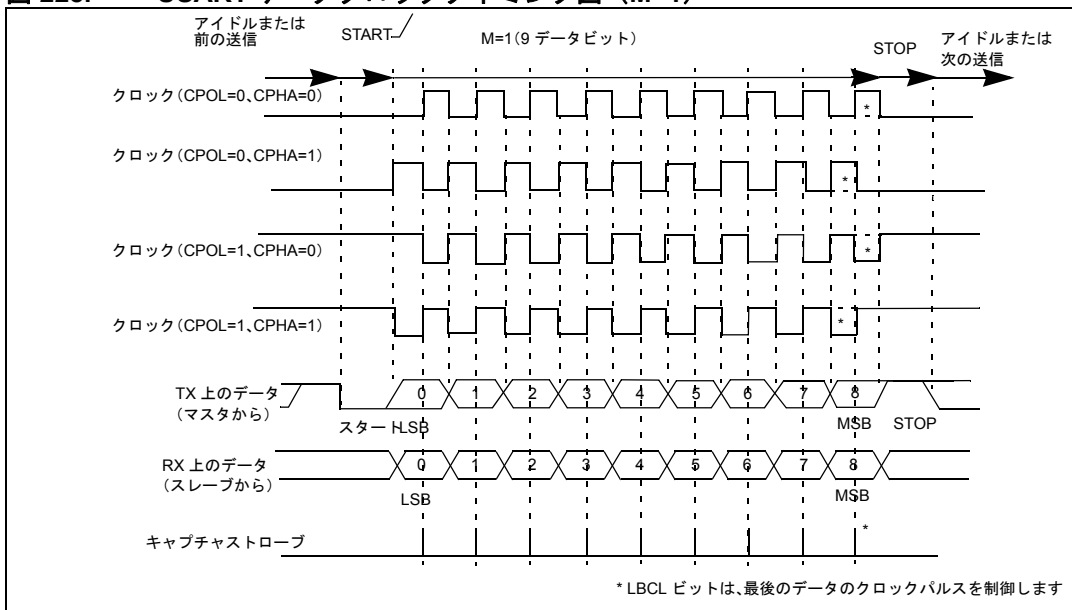
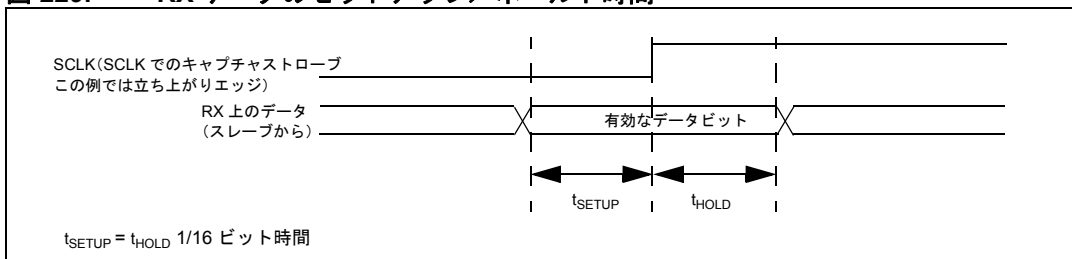


図 229. RX データのセットアップ/ホールド時間



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、スマートカードモードの説明を参照してください。

27.3.10 単線半二重通信

単線半二重モードを選択するには、USART_CR3 レジスタの HDSEL ビットをセットします。このモードでは、次のビットは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 および CLKEN ビット
- USART_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 ラインは内部的に接続されます。半二重通信と全二重通信の選択は、制御ビット“HALF DUPLEX SEL”(USART_CR3 レジスタの 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 ラインが内部的に接続されます。
- RX ピンは使用されなくなります。
- TX ピンは、データが送信されないときに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 USART によって駆動されないときはフローティング入力（または出力ハイのオープンドレイン）として設定されるように、入出力を設定する必要があります。

この点を除くと、通信は通常の USART モードの場合と同様です。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）。特に、TE ビットがセットされている間は、送信はハードウェアによってブロックされることなく、データがデータレジスタに書き込まれると送信を続行します。

27.3.11 スマートカード

スマートカードモードを選択するには、USART_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、次のビットは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 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。

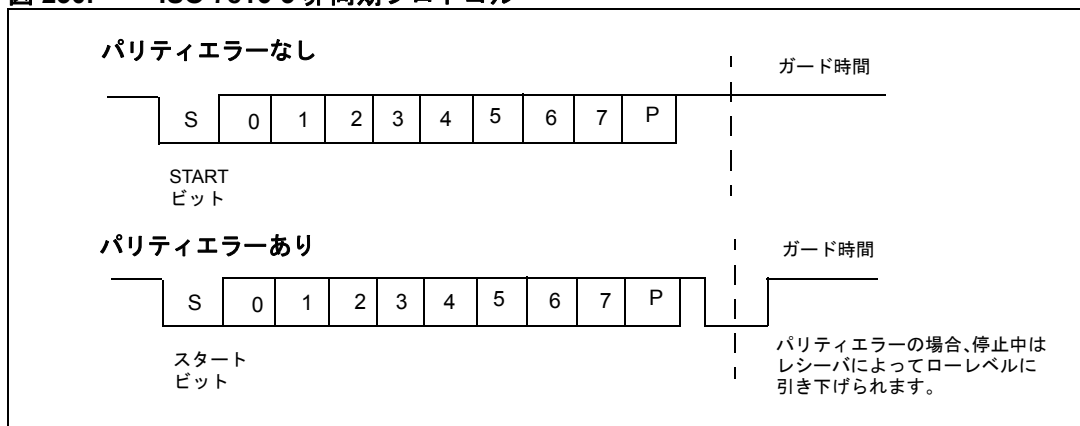
スマートカードインタフェースは、ISO 7816-3 標準で定義された非同期プロトコルスマートカードをサポートするように設計されています。USART は次のように設定してください。

- 8 ビット+パリティ：USART_CR1 レジスタの M=1、PCE=1
- 送受信では 1.5 ストップビット：USART_CR2 レジスタの STOP= 11。

注： 受信には 0.5 ストップビットを選択することも可能です。しかし、2 つの設定間の切り替えを避けるために、送信と受信の両方に 1.5 ストップビットを使用することを推奨します。

図 230 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 230. 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に設定する必要があります。

スマートカードは、単線半二重通信プロトコルです。

- 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。
- 0.5 または 1.5 のストップビット周期がプログラミング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に 1 ボークロック周期の間、送信ラインはローにプルされます。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。この NACK 信号（1 ボークロックの間、送信ラインをローにプル）は、1.5 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。アプリケーションは、プロトコルに基づいて、データの再送信に対処できます。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”されます。そうでない場合、NACK は送信されません。

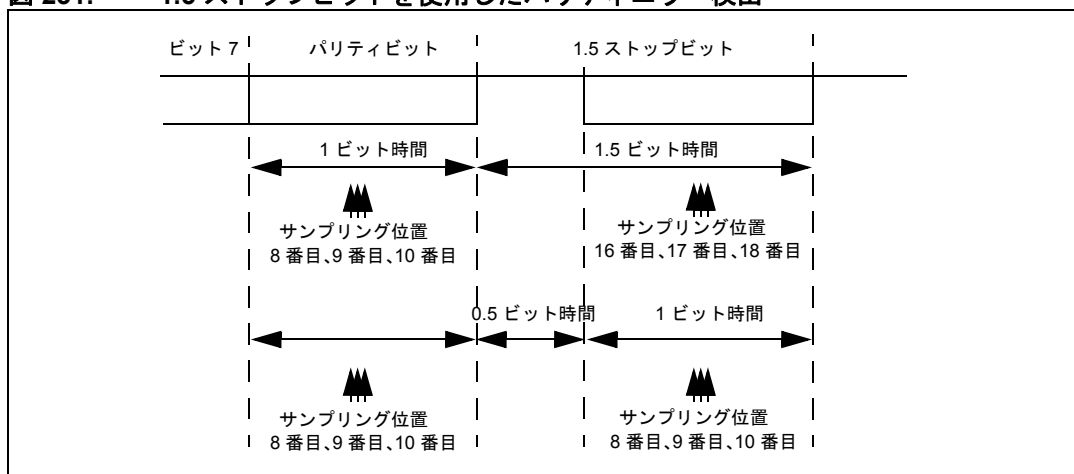
- TC フラグのアサーションは、ガードタイムレジスタのプログラミングによって遅らせることができます。通常動作では、TC がアサートされるのは、送信シフトレジスタが空であり、他に未処理の送信リクエストがない場合です。スマートカードモードでは、空の送信シフトレジスタは、ガードタイムカウンタをトリガして、ガードタイムレジスタにプログラミングされた値までカウントアップします。この間、TC は強制的にローレベルに保たれます。ガードタイムカウンタがプログラミングされた値に達すると、TC がハイにアサートされます。
- TC フラグのネゲートは、スマートカードモードの影響を受けません。
- レシーバからの 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 ボークロック周期にできます。
- 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、ブレイクキャラクタは重要ではありません。フレーミングエラー発生時のデータ 0x00 は、ブレイクではなくデータとして処理されます。

TE ビットをトグルしたとき、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 231 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、1.5 ストップビットが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 231. 1.5 ストップビット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力クロックから得られます。この分周比は、プリスケアラレジスタ USART_GTPR で設定されます。SCLK 周波数は $f_{CK}/2 \sim f_{CK}/62$ に設定することができます。ここで、 f_{CK} はペリフェラル入力クロック。

27.3.12 IrDA SIR ENDEC ブロック

IrDA モードを選択するには、USART_CR3 レジスタの IREN ビットをセットします。IrDA モードでは、次のビットは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、STOP、および CLKEN ビット
- USART_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 232 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。デコーダ入力が高レベルのとき、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 (つまり、USART が IrDA エンコーダにデータを送信している) 場合、IrDA デコーダは IrDA 受信ライン上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。データの受信中には、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 233 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 μ s 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、プログラミング可能です。レシーバ側のグリッチ検出回路は、2 PSC 周期 (PSC は IrDA 低電力ボーレジスタ USART_GTPR にプログラミング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。1 PSC 周期より小さな幅のパルスは常に除かれますが、1 から 2 周期の幅のパルスは受け入れられることも、除か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、USART_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 倍です。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マブル分周器は、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 (USART_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: 1~2 PSC 周期の幅のパルスは、除去されることも、除去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 フィジカルレイヤ仕様では、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 232. IrDA SIR ENDEC- ブロック図

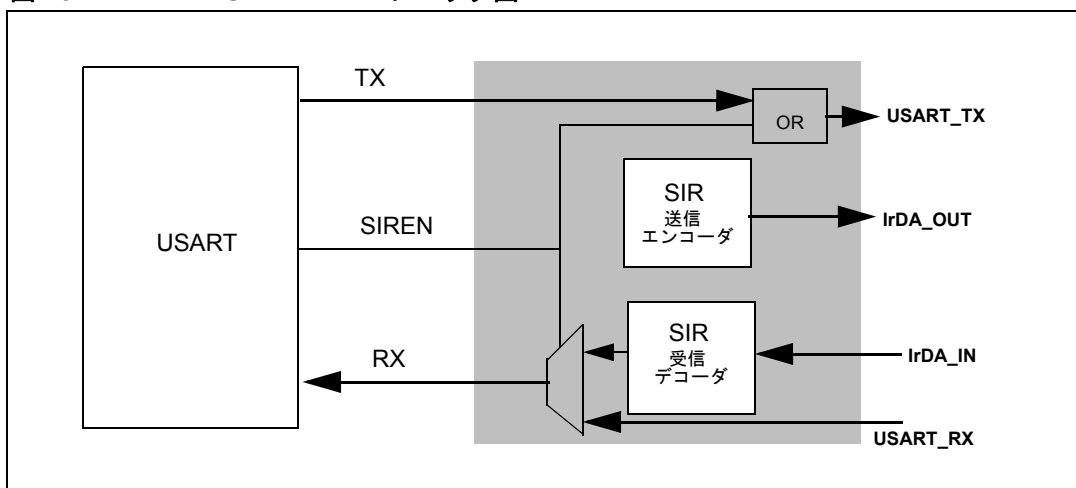
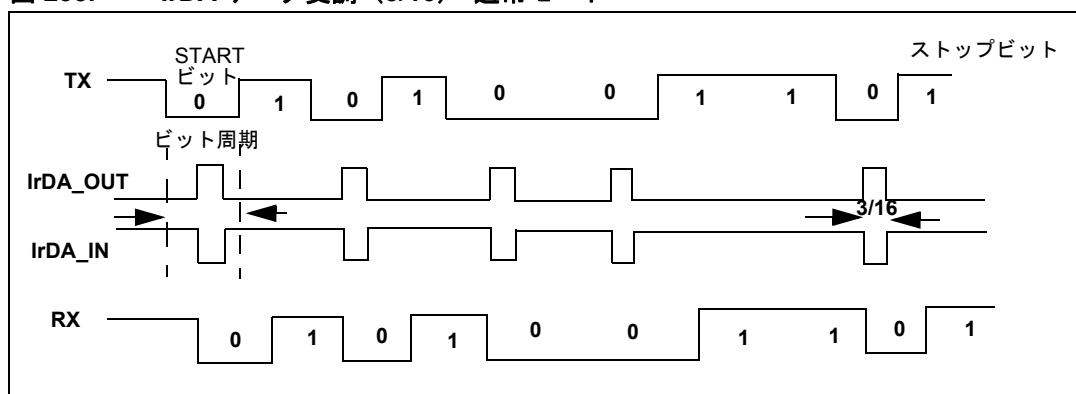


図 233. IrDA データ変調 (3/16) - 通常モード



27.3.13 DMA を使用した連続通信

USART は、DMA を使用して通信を継続することができます。受信バッファと送信バッファに対する DMA リクエストは、それぞれ独立して生成できます。

DMA を使用した送信

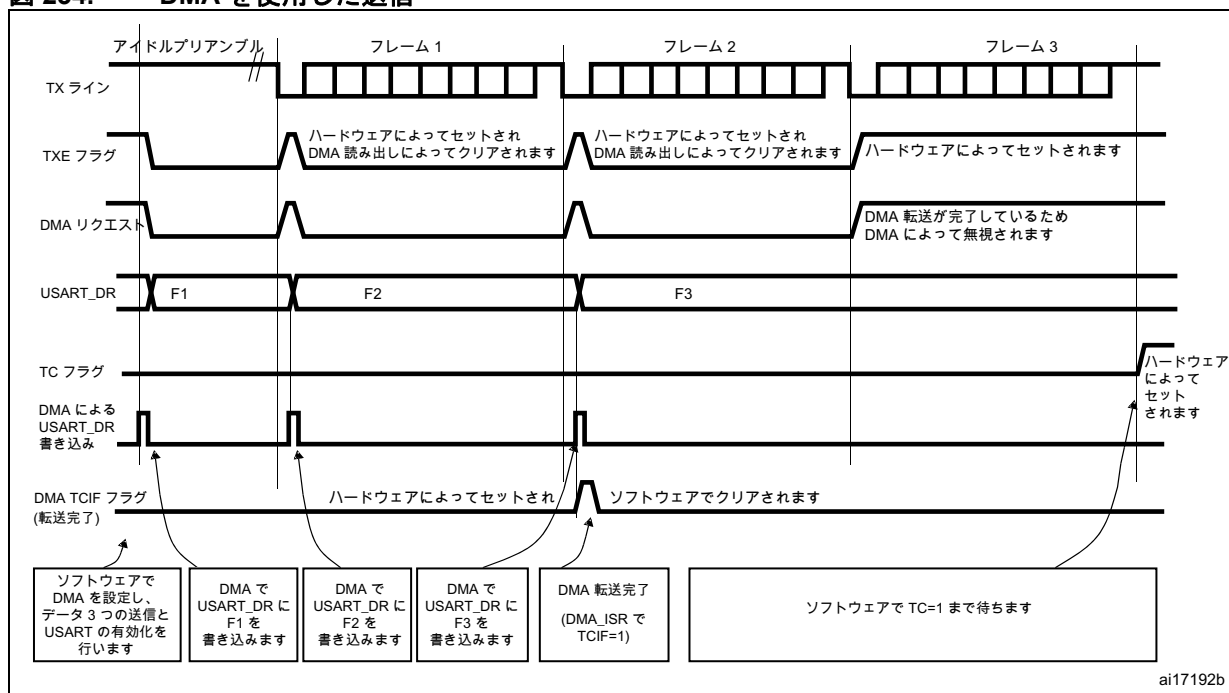
DMA モードでの送信を有効にするには、USART_CR3 レジスタの DMAT ビットをセットします。TXE ビットがセットされるたびに、データは、DMA ペリフェラル（DMA 仕様を参照）を使用して設定された SRAM 領域から USART_DR レジスタにロードされます。DMA チャンネルを USART 送信用に配置するには、次の手順で行います（x はチャンネル番号を示します）

1. DMA 制御レジスタに USART_DR レジスタのアドレスを書き込んで、これを転送先として設定します。データは、毎回の TXE イベント後に、メモリからこのアドレスに移されます。
2. DMA 制御レジスタにメモリアドレスを書き込んで、これを転送元として設定します。データは、毎回の TXE イベント後に、このメモリ領域から USART_DR 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 レジスタに設定します。
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. SR レジスタの TC ビットに 0 を書き込んでクリアします。
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したとき、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 フラグを観察することで USART 通信の完了を確認できます。これは、USART を無効にしたり STOP モードに入ったりする前に、最後の送信を破壊から守るために必要です。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、データ転送中は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 234. DMA を使用した送信



DMA を使用した受信

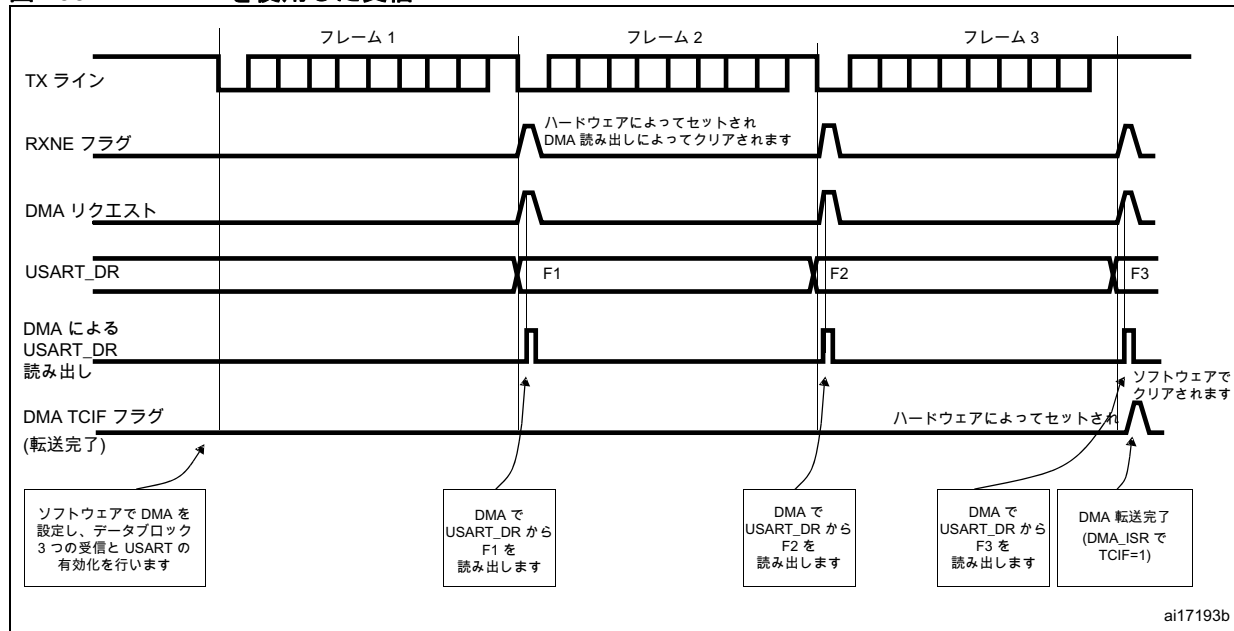
DMA モードでの受信を有効にするには、USART_CR3 レジスタの DMAR ビットをセットします。データバイトが受信されるたびに、データは、USART_DR レジスタから、DMA ペリフェラル (DMA 仕様を参照) を使用して設定された SRAM 領域にロードされます。DMA チャンネルを USART 受信用に配置するには、次の手順で行います。

1. DMA 制御レジスタに USART_DR レジスタのアドレスを書き込んで、これを転送元として設定します。データは、毎回の RXNE イベントの後に、このアドレスからメモリに移されます。
2. DMA 制御レジスタにメモリアドレスを書き込んで、これを転送先として設定します。データは、毎回の RXNE イベントの後に、USART_DR レジスタ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したとき、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。割り込み処理サブルーチンの中では、USART_CR3 レジスタの DMAR ビットをソフトウェアでクリアしなければなりません。

注： DMA を受信に使用する場合、RXNEIE ビットを有効にしてはいけません。

図 235. DMA を使用した受信



ai17193b

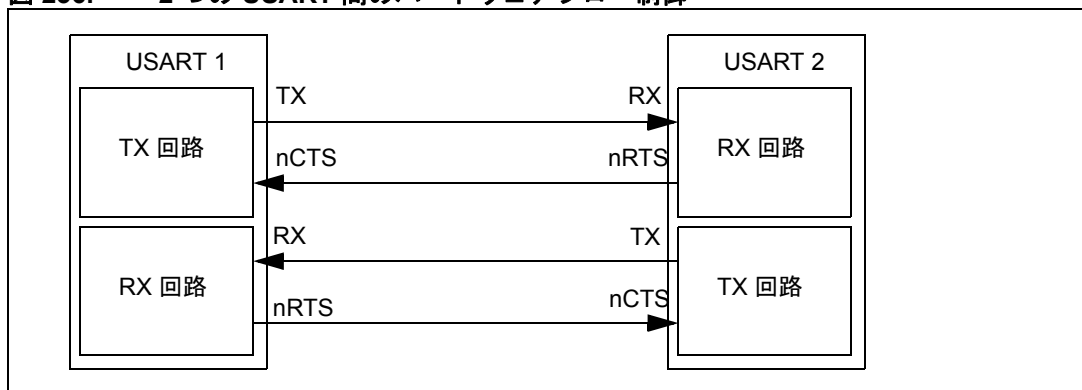
マルチバッファ通信における割り込み生成とエラーフラグ

マルチバッファ通信で、トランザクションの処理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。割り込みイネーブル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、別個のエラーフラグ割り込みイネーブルビット (USART_CR3 レジスタの EIE ビット) が用意されていて、セットされている場合、いずれかのエラーを持つ現在のバイトの後で割り込みを発行します。

27.3.14 ハードウェアフロー制御

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 236](#) 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 236. 2 つの USART 間のハードウェアフロー制御

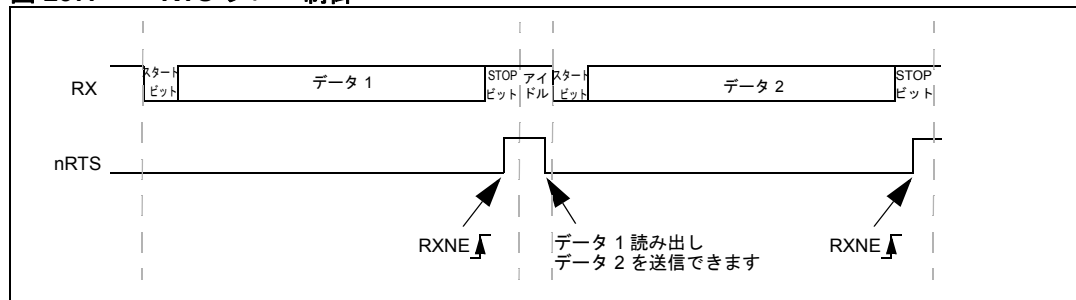


RTS と CTS のフロー制御は、USART_CR3 レジスタのそれぞれ RTSE ビットと CTSE ビットに 1 を書き込むことによって、個別に有効にできます。

RTS フロー制御

RTS フロー制御が有効にされた場合 (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 237 に、RTS フロー制御を有効にした通信の例を示します。

図 237. RTS フロー制御

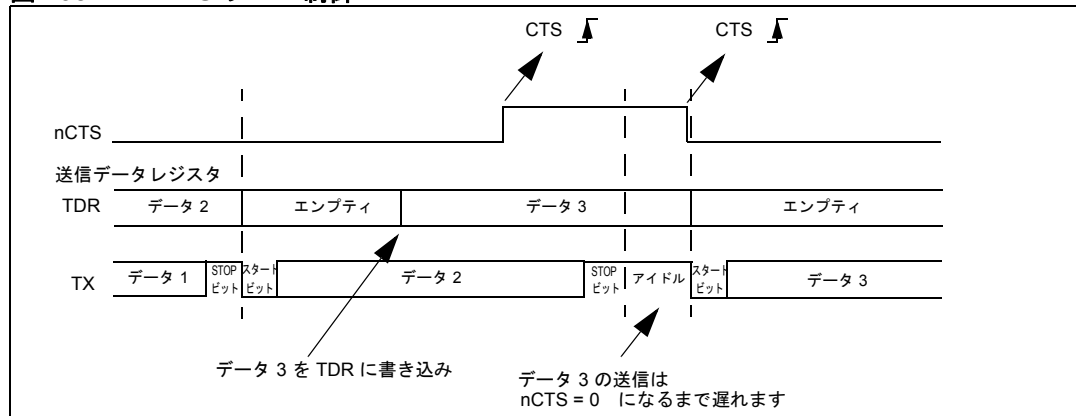


CTS フロー制御

CTS フロー制御が有効にされた場合 (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 がアサートされている場合 (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ものと想定、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 入力がトグルすると、CTSIF ステータスビットはただちに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ないかどうかを示します。USART_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。次の図に、CTS フロー制御を有効にしたときの通信例を示します。

図 238. CTS フロー制御



注： ブレークフレームの特殊な動作：CTS フローが有効になると、トランスミッタは、ブレークを送信するために nCTS 入力の状態をチェックしません。

27.4 USART 割り込み

表 155. USART 割り込みリクエスト

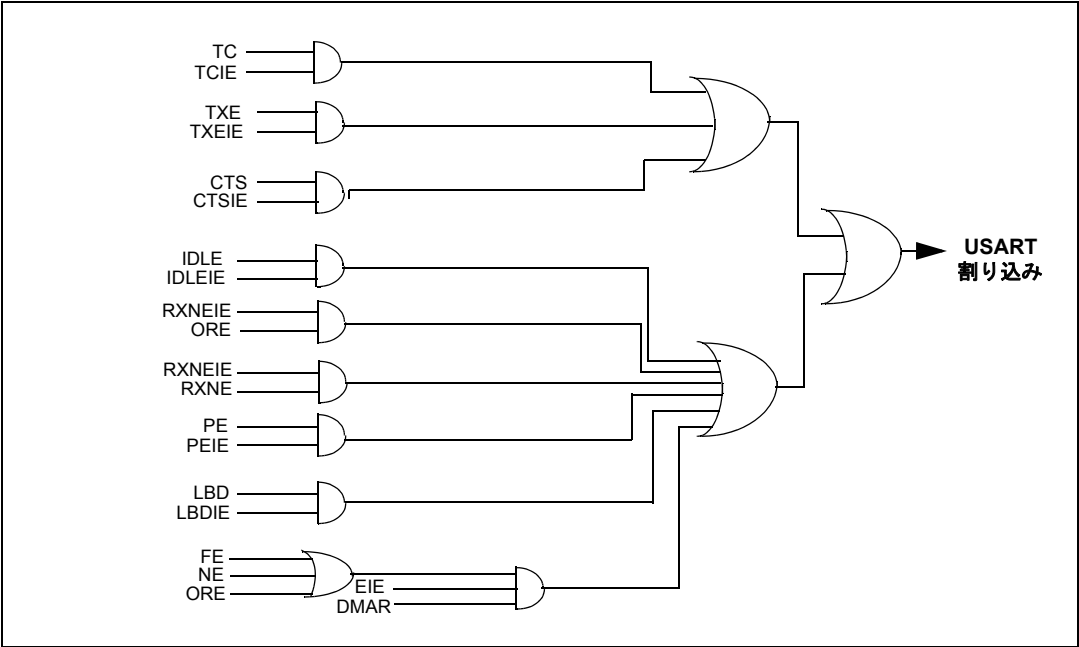
割り込みイベント	イベントフラグ	イネーブル制御ビット
送信データレジスタエンプティ	TXE	TXEIE
CTS フラグ	CTS	CTSIE
送信完了	TC	TCIE
受信データ読み出し可能	RXNE	RXNEIE
オーバーランエラー検出	ORE	
アイドルライン検出	IDLE	IDLEIE
パリティエラー	PE	PEIE
ブレークフラグ	LBD	LBDIE
マルチバッファ通信におけるノイズフラグ、オーバーランエラー、およびフレーミングエラー	NF または ORE または FE	EIE

USART 割り込みイベントは、同じ割り込みベクタに接続されます（図 239を参照）。

- 送信時：送信完了割り込み、CTS 割り込み、または送信データレジスタエンプティ割り込み
- 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（マルチバッファ通信時のみ）、およびフレーミングエラー（マルチバッファ通信時のみ）

これらのイベントは、対応するイネーブル制御ビットがセットされている場合、割り込みを生成します。

図 239. USART 割り込みマッピング図



27.5 USART モード設定

表 156. USART モード設定⁽¹⁾

USART モード	USART1	USART2	USART3	UART4	UART5
非同期モード	X	X	X	X	X
ハードウェアフロー制御	X	X	X	NA	NA
マルチバッファ通信 (DMA)	X	X	X	X	X
マルチプロセッサ通信	X	X	X	X	X
同期	X	X	X	NA	NA
スマートカード	X	X	X	NA	NA
半二重 (1線式モード)	X	X	X	X	X
IrDA	X	X	X	X	X
LIN	X	X	X	X	X

1. X = サポート、NA = 該当せず

27.6 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。

ペリフェラルレジスタは、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。

27.6.1 ステータスレジスタ (USART_SR)

アドレスオフセット : 0x00

リセット値 : 0x00C0 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						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
						rc_w0	rc_w0	r	rc_w0	rc_w0	r	r	r	r	r

ビット 31:10 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 **CTS** : CTS フラグ

CTSE ビットがセットされている場合、nCTS 入力がトグルすると、このビットはハードウェアによってセットされます。このビットは、ソフトウェアでクリアされます（このビットに 0 を書き込むことによって）。USART_CR3 レジスタの CTSIE=1 である場合、割り込みが生成されます。

0 : nCTS ステータスラインに変化はありませんでした。

1 : nCTS ステータスラインに変化がありました。

ビット 8 **LBD** : 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。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。USART_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： **LBDIE=1 の場合、LBD=1 になると割り込みが生成されます。**

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、TDR レジスタの内容がシフトレジスタに転送されたとき、ハードウェアによってセットされます。USART_CR1 レジスタの TXEIE=1 である場合、割り込みが生成されます。このビットは、USART_DR レジスタへの書き込みによってクリアされます。

0 : データはシフトレジスタに転送されていません。

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。USART_CR1 レジスタの TCIE=1 である場合、割り込みが生成されます。ソフトウェアシーケンスによってクリアされます（USART_SR レジスタの読み出しに続く USART_DR レジスタの書き込み）。TC ビットは、“0”を書き込むことによってもクリアできます。このクリアシーケンスは、マルチバッファ通信の場合にのみ推奨されます。

0 : 送信は完了していません。

1 : 送信は完了しています。

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 USART_DR レジスタに転送されたときに、ハードウェアによってセットされます。USART_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。このビットは、USART_DR レジスタへの読み出しによってクリアされます。RXNE フラグは、0 を書き込むことによってもクリアできます。このクリアシーケンスは、マルチバッファ通信の場合にのみ推奨されます。

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。USART_CR1 レジスタの IDLEIE=1 である場合、割り込みが生成されます。ソフトウェアシーケンスによってクリアされます (USART_SR レジスタへの読み出しと、それに続く USART_DR レジスタへの読み出し)。

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 (つまり、新しいアイドルラインが出現する) まで、IDLE ビットが再びセットされることはありません。***

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、シフトレジスタに受信中のワードを 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。USART_CR1 レジスタの RXNEIE=1 の場合、割り込みが生成されます。ソフトウェアシーケンスによってクリアされます (USART_SR レジスタへの読み出しと、それに続く USART_DR レジスタへの読み出し)。

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。マルチバッファ通信で EIE ビットがセットされた場合、ORE フラグに基づいて割り込みが生成されます。***

ビット 2 NF : 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。ソフトウェアシーケンスによってクリアされます (USART_SR レジスタへの読み出しと、それに続く USART_DR レジスタへの読み出し)。

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しません。なぜなら、割り込みを生成する RXNE ビットと同時に現れるからです。マルチバッファ通信で EIE ビットがセットされている場合、NF フラグに基づいて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 フラグを無効にするには、ONEBIT ビットに 1 をプログラミングして偏差に対する USART の許容誤差を増やします (セクション 27.3.5 : クロック偏差に対する USART レシーバの許容誤差 (739 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。ソフトウェアシーケンスによってクリアされます (USART_SR レジスタへの読み出しと、それに続く USART_DR レジスタへの読み出し)。

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。

注 : ***このビットは割り込みを生成しません。なぜなら、割り込みを生成する RXNE ビットと同時に現れるからです。現在転送中のワードによってフレームエラーとオーバーランエラーが発生した場合、転送は行われ、ORE ビットのみがセットされます。***

マルチバッファ通信で EIE ビットがセットされている場合、FE フラグに基づいて割り込みが生成されます。

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。ソフトウェアシーケンスによってクリアされます (ステータスレジスタへの読み出しと、それに続く USART_DR データレジスタへの読み出し)。ソフトウェアは、RXNE フラグがセットされるまで待たなければ、PE ビットをクリアできません。

USART_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。

27.6.2 データレジスタ (USART_DR)

アドレスオフセット : 0x04

リセット値 : 0xFFFF XXXX

ビット 31:9 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 **DR[8:0]** : データ値

読み出しが書き込みかに応じて、受信または送信のデータキャラクタを含みます。
データレジスタは、送信用 (TDR) と受信用 (RDR) の 2 個のレジスタから構成されるため、2 つの機能 (読み出しと書き込み) を実行します。
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 1 を参照)。
RDR レジスタは、入力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。
パリティを有効にして (USART_CR1 レジスタの PCE ビットに 1 を設定して) 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。
パリティを有効にして受信しているとき、MSB ビットに読み込まれる値は、受信したパリティビットです。

27.6.3 ボーレートレジスタ (USART_BRR)

注 : *TE または RE ビットがそれぞれ無効にされている場合、ボーカウンタはカウントを停止します。*

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DIV_Mantissa[11:0]												DIV_Fraction[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 **DIV_Mantissa[11:0]** : USARTDIV の仮数部

これらの 12 ビットは、USART ディバイダ (USARTDIV) の仮数部を定義します。

ビット 3:0 **DIV_Fraction[3:0]** : USARTDIV の小数部

これらの 4 ビットは、USART ディバイダ (USARTDIV) の小数部を定義します。OVER8=1 のとき、DIV_Fraction3 ビットは考慮されないため、クリアされた状態に保つ必要があります。

27.6.4 制御レジスタ 1 (USART_CR1)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVER8	予約済み	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK
rw	予約済み	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 **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

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。SCEN=1、IREN=1、または LINEN=1 であるとき、OVER8 はハードウェアによって"0"に固定されています。

ビット 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 **UE** : USART イネーブル

このビットがクリアされると、消費電力を減らすために、USART プリスケアラと出力は、現在のバイト転送の最後に停止します。このビットは、ソフトウェアによってセット/クリアします。

0 : USART プリスケアラと出力は無効です。

1 : USART は有効です。

ビット 12 **M** : ワード長

このビットは、ワード長を決定します。ソフトウェアによってセット/クリアされます。

0 : 1 スタートビット、8 データビット、n ストップビット

1 : 1 スタートビット、9 データビット、n ストップビット

注 : M ビットは、送受信のデータ転送中に変更することはできません。

ビット 11 **WAKE** : ウェイクアップ方法

このビットは、USART のウェイクアップ方法を決定し、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

ビット 10 **PCE** : パリティ制御イネーブル

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/クリアします。このビットがセットされると、送受信での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。

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た) 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

ビット 8 **PEIE** : PE 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/クリアします。

0 : 割り込みは禁止です。

1 : USART_SR レジスタで PE=1 になるたび、USART 割り込みが生成されます。



ビット 7 TXEIE : TXE 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/クリアします。

0 : 割り込みは禁止です。

1 : USART_SR 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 : 送信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/クリアします。

0 : 割り込みは禁止です。

1 : USART_SR 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/クリアします。

0 : 割り込みは禁止です。

1 : USART_SR レジスタで RXNE=1 または ORE=1 になるたび、USART 割り込みが生成されます。

ビット 4 IDLEIE : IDLE 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/クリアします。

0 : 割り込みは禁止です。

1 : USART_SR 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 : トランスミッタイネーブル

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 : 1 : スマートカードモードの場合を除いて、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プリアンブル (アイドルライン) が送信されます。

2 : 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 : レシーバイネーブル

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を開始します。

ビット 1 RWU : レシーバウェイクアップ

このビットは、USART がミュートモードかどうかを決定します。ソフトウェアによってセット/クリアされ、ウェイクアップ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 : 1 : ミュートモードを選択する (RWU ビットをセットする) 前に、まず、USART はデータバイトを受信しなければなりません。そうしないと、ミュートモードで、アイドルライン検出によるウェイクアップを伴う動作はできません。

2 : アドレスマーク検出のウェイクアップ設定 (WAKE=1) では、RXNE ビットがセットされている限り、ソフトウェアで RWU ビットを変更することはできません。

ビット 0 SBK : ブレーク送信

このビットは、ブレークキャラクタを送信するために使用されます。ソフトウェアによってセット/クリアできます。ソフトウェアによってセットされ、ブレークのストップビットの処理中にハードウェアによってリセットされます。

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。

27.6.5 制御レジスタ 2 (USART_CR2)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み	LINEN	STOP[1:0]		CLKEN	CPOL	CPHA	LBCL	予約済み	LBDIE	LBDL	予約済み	ADD[3:0]			
	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 **LINEN** : LIN モードイネーブル

このビットは、ソフトウェアによってセット/クリアします。

0 : LIN モードを無効化

1 : LIN モードを有効化

LIN モードでは、USART_CR1 レジスタの SBK ビットを使用して LIN 同期ブレーク (下位 13 ビット) を送信し、LIN 同期ブレークを検出することができます。

ビット 13:12 **STOP** : STOP ビット

これらのビットは、ストップビットのプログラミングに使用されます。

00 : 1 ストップビット

01 : 0.5 ストップビット

10 : 2 ストップビット

11 : 1.5 ストップビット

ビット 11 **CLKEN** : クロックイネーブル

このビットによって、SCLK ピンを有効にできます。

0 : SCLK ピンを無効化

1 : SCLK ピンを有効化

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 ビットと連携して機能し、希望するクロック/データ関係をもたらします。

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。

ビット 9 **CPHA** : クロック位相

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位相を選択できます。CPOL ビットと連携して機能し、希望するクロック/データ関係をもたらします (図 227 から 228 を参照)。

注 : 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。

ビット 8 **LBCL** : 最終ビットのクロックパルス

このビットによって、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します。

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。

最終ビットは、USART_CR1 レジスタの M ビットによって選択された 8 または 9 ビットフォーマットに応じて送信された 8 番目または 9 番目のデータビットです。

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 **LBDIE** : LIN ブレーク検出割り込みイネーブル

ブレーク割り込みマスク (ブレークデリミタを使用したブレーク検出)。

0 : 割り込みは禁止です。

1 : USART_SR レジスタで LBD=1 になると、割り込みが生成されます。



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

ビット 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 3:0 **ADD[3:0]** : USART ノードのアドレス

このビットフィールドは、USART ノードのアドレスを示します。

これは、ミュートモードでの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に使用されます。

注 : **CPOL**、**CPHA**、**LBCL** の 3 ビットには、トランスミッタが有効である間は書き込みを行わないでください。

27.6.6 制御レジスタ 3 (USART_CR3)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み				ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE
				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 **ONEBIT** : 1 サンプルビット方式イネーブル

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

ビット 10 **CTSIE** : CTS 割り込みイネーブル

0 : 割り込みは禁止です。

1 : USART_SR レジスタで CTS=1 になるたび、割り込みが生成されます。

ビット 9 **CTSE** : CTS イネーブル

0 : CTS ハードウェアフロー制御は無効です。

1 : CTS モードは有効であり、データは、nCTS 入力のアサートされている (0 に接続) ときにのみ送信されます。データの送信中に nCTS 入力にネゲートされた場合、送信が完了してから停止します。nCTS がアサートされていないときにデータレジスタにデータが書き込まれた場合、送信は nCTS がアサートされるまで延期されます。

ビット 8 **RTSE** : RTS イネーブル

0 : RTS ハードウェアフロー制御は無効です。

1 : RTS 割り込みは有効であり、データは、受信バッファに空きがあるときにのみリクエストされます。データの送信は、現在のキャラクタが送信された後で終わるものと期待されます。nRTS 出力は、データ受信が可能になるときにアサート (0 に接続) されます。

ビット 7 **DMAT** : DMA 対応のトランスミッタ

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。

1 : DMA モードの送信は有効です。

0 : DMA モードの送信は無効です。

ビット 6 DMAR : DMA 対応のレシーバ

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。

1 : DMA モードの受信は有効です。

0 : DMA モードの受信は無効です。

ビット 5 SCEN : スマートカードモードイネーブル

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されます。

0 : スマートカードモードは無効です。

1 : スマートカードモードは有効です。

ビット 4 NACK : スマートカード NACK イネーブル

0 : パリティエラーの場合の NACK 送信は無効です。

1 : パリティエラーの場合の NACK 送信は有効です。

ビット 3 HDSEL : 半二重選択

単線半二重モードの選択。

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。

ビット 2 IRLP : IrDA 低消費電力

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

ビット 1 IREN : IrDA モードイネーブル

このビットは、ソフトウェアによってセット/クリアします。

0 : IrDA を無効化

1 : IrDA を有効化

ビット 0 EIE : エラー割り込みイネーブル

エラー割り込みイネーブルビットは、マルチバッファ通信 (USART_CR3 レジスタの DMAR=1) において、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 (USART_SR レジスタの FE=1、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 : 割り込みは禁止です。

1 : USART_CR3 レジスタで DMAR=1 になるたび、および USART_SR レジスタで FE=1、ORE=1、または NF=1 になるたび、割り込みが生成されます。

27.6.7 ガード時間/プリスケアラレジスタ (USART_GTPR)

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
GT[7:0]								PSC[7:0]							
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 15:8 **GT[7:0]** : ガード時間値

このビットフィールドは、ガード時間値をバウンス数で与えます。
これはスマートカードモードで使用されます。このガード時間値が経過すると、送信完了フラグがセットされます。

ビット 7:0 **PSC[7:0]** : プリスケアラ値

– **IrDA 低電力モード** :

PSC[7:0] = IrDA 低消費電力ボーレート
システムクロック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されます。
ソースクロックは、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 - この値をプログラミングしないでください。
00000001 : ソースクロックは 1 分周されます。
00000010 : ソースクロックは 2 分周されます。
...

– **IrDA 通常モード** : PSC には 00000001 を設定する必要があります。

– **スマートカードモード** :

PSC[4:0] : プリスケアラ値
システムクロックを分周してスマートカードのクロックを提供するためのプリスケアラのプログラミングに使用されます。
ソースクロック周波数の分周係数を与えるために、このレジスタに与えられた値（上位 5 ビット）は 2 倍されます。
00000 : 予約済み - この値をプログラミングしないでください。
00001 : ソースクロックは 2 分周されます。
00010 : ソースクロックは 4 分周されます。
00011 : ソースクロックは 6 分周されます。
...

注 : 1 : スマートカードモードが使用される場合、ビット [7:5] は無効です。



27.6.8 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 157. 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	USART_SR	予約済み																						CTS	LBD	TXE	TC	RXNE	IDLE	ORE	NF	FE	PE
	リセット値																							0	0	1	1	0	0	0	0	0	0
0x04	USART_DR	予約済み																						DR[8:0]									
	リセット値																							0	0	0	0	0	0	0	0	0	0
0x08	USART_BRR	予約済み														DIV_Mantissa[15:4]										DIV_Fraction [3:0]							
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x0C	USART_CR1	予約済み														OVER8	予約済み	UE	M	WAKE	PCE	PS	PEIE	TXEIE	TCIE	RXNEIE	IDLEIE	TE	RE	RWU	SBK		
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
0x10	USART_CR2	予約済み														LINEN	STOP [1:0]		CLKEN	CPOL	CPHA	LBCL	予約済み	LBDIE	LBDL	予約済み	ADD[3:0]						
	リセット値															0	0	0	0	0	0	0	0	予約済み	0	0	0	0	0				
0x14	USART_CR3	予約済み														ONEBIT	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	NACK	HDSEL	IRLP	IREN	EIE						
	リセット値															0	0	0	0	0	0	0	0	0	0	0	0	0					
0x18	USART_GTPR	予約済み														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				

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

28 シリアルペリフェラルインタフェース (SPI)

28.1 SPI の概要

SPI インタフェースは、SPI プロトコルまたは I²S オーディオプロトコルをサポートする 2 つの主要機能を提供します。デフォルトで選択されているのは、SPI 機能です。ソフトウェアから、インタフェースを SPI から I²S へ切り替えることができます。

Cat.1 と Cat.2 デバイスでは I²S プロトコルは利用できません。

SPI (シリアルペリフェラルインタフェース) では、外部デバイスとの間で、半二重/全二重、および同期シリアル通信が可能です。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。

2 本のライン (双方向データラインを使用可能) での半二重同期転送や、CRC チェックを使用した信頼性の高い通信など、さまざまな目的に使用することができます。

I²S は、同期シリアル通信インタフェースでもあります。I²S フィリップス標準、MSB/LSB 詰め標準、PCM 標準など、4 つのオーディオ標準に対応できます。全二重モード (4 ピン使用) または半二重モード (6 ピン使用) でマスタデバイスまたはスレーブデバイスとして動作することができます。I²S が通信マスタとして設定されると、インタフェースは外部スレーブ部品にマスタクロックを供給できます。

警告 : SPI1 ピンには、JTAG インタフェースと共用されているピンがあるため (SPI1_NSS は JTDI と、SPI1_SCK は JTDO と、SPI1_MISO は NJTRST と、それぞれ共用)、次のいずれかが可能です :

- SPI1 を他のピンへ割り当てる
- JTAG を無効にして、SPI IO としてリストされたピンを設定する前に SWD インタフェースを使用するようにするか (アプリケーションのデバッグ時)、あるいは、
- JTAG/SWD の両インタフェースを無効にします (スタンドアロンアプリケーションの場合)。

JTAG/SWD インタフェースピンの設定の詳細については、[セクション 7.3.2 : I/O ピンマルチプレクサおよびマッピング](#)を参照してください。

28.2 SPI と I²S の主な機能

28.2.1 SPI 機能

- 3本のラインで全二重同期転送
- 2本のラインで単方向同期転送（双方向データラインの有無にかかわらず）
- 8または16ビットの転送フレームフォーマット選択
- マスタまたはスレーブ動作
- 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 $f_{PCLK}/2$ ）
- スレーブモード周波数（最大 $f_{PCLK}/2$ ）
- マスタとスレーブの両方で高速通信
- 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ミング可能
- データ順序をプログラミング可能（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 TI モード（Cat.3、Cat.4、Cat.5 デバイスの場合）
- ハードウェア CRC 機能による信頼性の高い通信：
 - － 送信モードでは 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、および CRC エラーフラグ（割り込み機能付き）
- 1バイトの送受信バッファ（DMA 機能付き：送受信リクエスト）

28.2.2 I²S機能

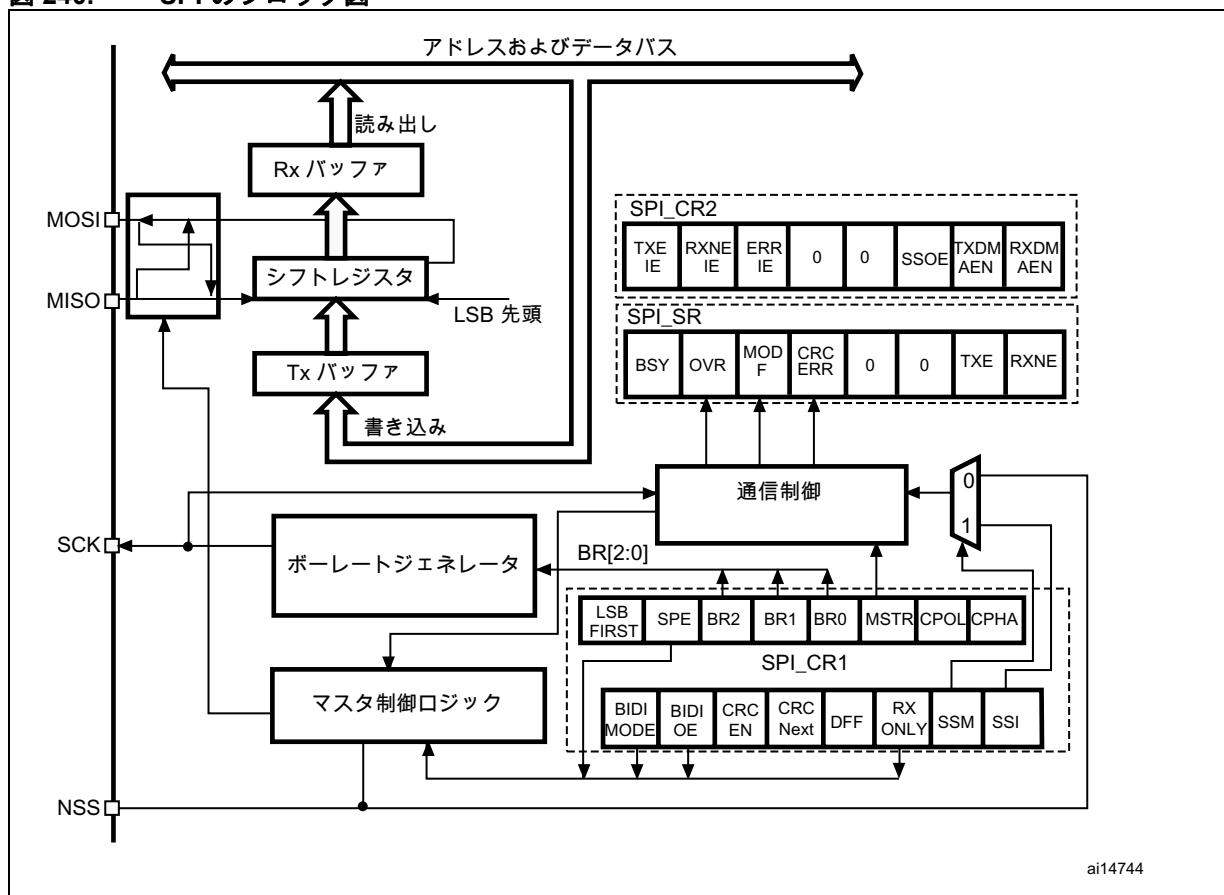
- 半二重 通信 (トランスミッタまたはレシーバのみ)
- マスタ動作またはスレーブ動作
- 正確なプロトコルサンプル周波数 (8~192 kHz) を実現するプログラミング可能な 8 ビットのリニアプリスケアラ
- 16 ビット、24 ビット、または 32 ビットのデータフォーマット
- パケットフレームはプロトコルチャネルによって 16 ビット (16 ビットデータフレーム) または 32 ビット (16 ビット、24 ビット、32 ビットデータフレーム) に固定
- プログラミング可能なクロック極性 (定常状態)
- スレーブ送信モードでのアンダーランフラグと受信モード (マスタとスレーブ) でのオーバーランフラグ
- 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)
- 次のI²S プロトコルをサポート:
 - I²S フィリップス標準
 - MSB 詰め標準 (左詰め)
 - LSB 詰め標準 (右詰め)
 - PCM 標準 (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
- 外部プロトコル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。周波数レートは $256 \times F_S$ に固定 (F_S はオーディオサンプリング周波数)
- 外部マスタデバイスと I²S スレーブデバイスの間のフレーム再同期を検出するフレームエラーフラグ。

28.3 SPI 機能詳細

28.3.1 概要

SPI のブロック図をに示します。図 240

図 240. SPI のブロック図



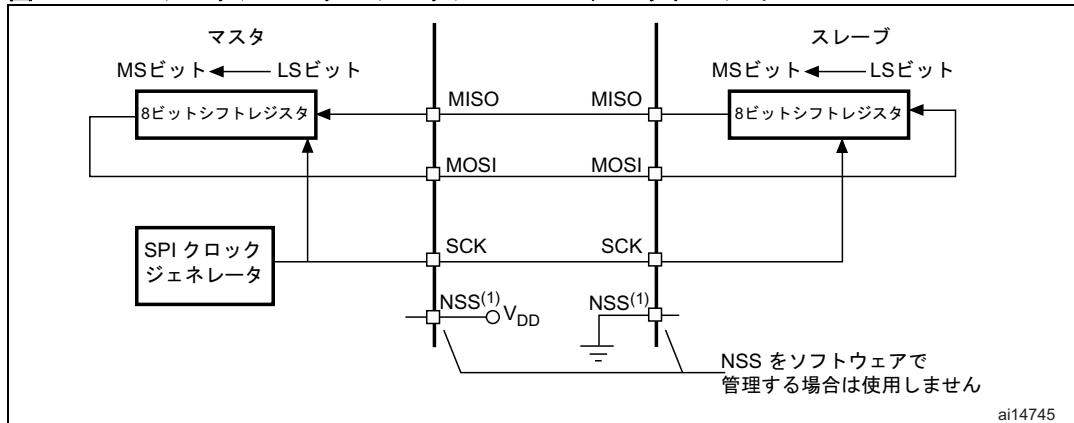
通常、SPI は 4 本のピンで外部デバイスに接続されます。

- MISO : Master In / Slave Out データ。このピンは、マスタモードではデータの受信に、スレーブモードではデータの送信に使用できます。
- MOSI : Master Out / Slave In データ。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用できます。
- SCK (シリアルクロック) : SPI マスタでは出力に、SPI スレーブでは入力に使用されます。
- NSS : スレーブ選択。これはスレーブデバイスを選択するためのオプションピンです。このピンが「チップセレクト」として機能することで、SPI マスタは複数のスレーブと個別に通信を行うことができ、データラインの競合を避けることができます。スレーブの NSS 入力は、マスタデバイスの標準入出力ポートによって駆動できます。NSS ピンは、SSOE ビットによって有効にされている場合は出力として使用することもでき、SPI がマスタ設定の場合はローレベルに駆動できます。このように、マスタの NSS ピンに接続されているデバイスからのすべての NSS ピンは、NSS ハードウェアモードに設定されている場合、ローレベルを受けてスレーブになります。NSS が入力として設定された状態で SPI がマスタモードに設定され (MSTR=1 および SSOE=0)、NSS がローレベルに引かれた場合、SPI はマスタモードフォールト状態に入ります。

MSTR ビットは自動的にクリアされ、デバイスはスレーブモードに設定されます（[セクション 28.3.10 : エラーフラグ \(794 ページ\)](#) を参照）。

1 個のマスタと 1 個のスレーブの間の基本的な相互接続例を [図 241](#) に示します。

図 241. シングルマスタ/シングルスレーブアプリケーション



1. この例では、NSS ピンは入力として設定されています。

MOSI ピンと MISO ピンは、それぞれに相互接続されています。このようにして、データはマスタとスレーブの間を（MSB ファーストで）シリアルに転送されます。

通信は常にマスタによって開始されます。マスタデバイスが MOSI ピンを介してスレーブデバイスにデータを送信すると、スレーブデバイスは MISO ピンを介して応答します。このことは、データ出力とデータ入力マスタデバイスから SCK ピンを介して供給される同じクロック信号に同期した全二重通信を意味します。

スレーブ選択 (NSS) ピンの管理

ハードウェアまたはソフトウェアのスレーブ選択管理の設定は、SPI_CR1 レジスタの SSM ビットを使用して行うことができます。

- ソフトウェア NSS 管理 (SSM = 1)

スレーブ 選択情報は、内部で SPI_CR1 レジスタの SSI ビット値から駆動されます。外部NSS ピンは他のアプリケーションで自由に使えます。
- ハードウェア NSS 管理 (SSM = 0)

NSS 出力設定 (SPI_CR2 レジスタの SSOE ビット) に応じて 2 つの設定が可能です。

 - NSS 出力を有効化 (SSM = 0、SSOE = 1)

この設定は、デバイスがマスタモードで動作する場合にのみ使用されます。NSS 信号は、マスタが通信を開始したときにロー・レベルになり、SPI が無効化されるまでローレベルを維持します。
 - NSS 出力を無効化 (SSM = 0、SSOE = 0)

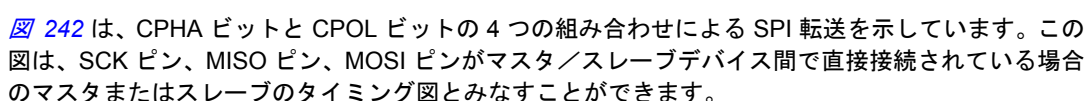
この設定により、マスタモードで動作するデバイスでマルチマスタ機能が可能になります。スレーブに設定されたデバイスでは、NSS ピンは従来型の NSS 入力として機能します (NSS がローレベルのときにスレーブが選択され、NSS がハイレベルのときに非選択)。

クロックの位相と極性

SPI_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、4 つのタイミング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定常値を制御します。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 がリセットされた場合、SCK ピンはローレベルのアイドル状態になります。CPOL がセットされた場合、SCK ピンはハイレベルのアイドル状態になります。

CPHA (クロック位相) ビットがセットされている場合、SCK ピンの 2 番目のエッジ (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) が最上位ビットのキャプチャストロブです。データは、2 番目のクロック遷移でラッチされます。CPHA ビットがリセットされている場合、SCK ピンでの最初のエッジ (CPOL ビットがリセットされていれば立ち上がりエッジ、セットされていれば立ち下がりエッジ) が最上位ビットのキャプチャストロブです。データは、最初のクロック遷移でラッチされます。

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。

 図 242 は、CPHA ビットと CPOL ビットの 4 つの組み合わせによる SPI 転送を示しています。この図は、SCK ピン、MISO ピン、MOSI ピンがマスタ/スレーブデバイス間で直接接続されている場合のマスタまたはスレーブのタイミング図とみなすことができます。

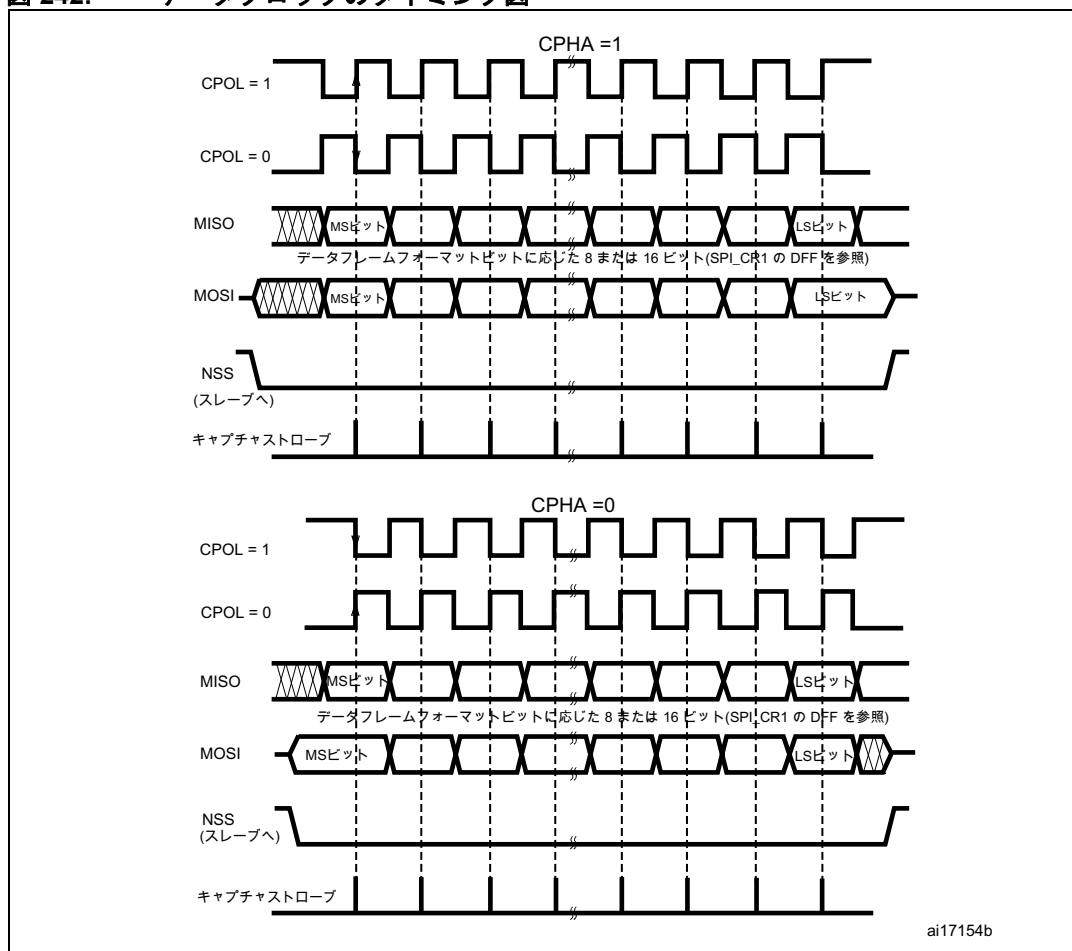
注： CPOL/CPHA ビットを変更する前に、SPE ビットをリセットすることによって、SPI を無効にする必要があります。

マスタとスレーブは、同じタイミングモードでプログラミングする必要があります。

SCK のアイドル状態は、SPI_CR1 レジスタで (CPOL=1 の場合は SCK のプルアップ、CPOL=0 の場合は 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。

データフレームフォーマット (8 ビットまたは 16 ビット) は、SPI_CR1 レジスタの DFF ビットによって選択され、送受信時のデータ長を決定します。

図 242. データクロックのタイミング図



1. これらのタイミングは、SPI_CR1 レジスタの LSBFIRST ビットをリセットした状態で示されています。

データフレームフォーマット

データは、SPI_CR1 レジスタの LSBFIRST ビットの値に応じて、MSB ファーストまたは LSB ファーストでシフトアウトできます。

各データフレームの長さは、SPI_CR1 レジスタの DFF ビットを使用してプログラミングされたデータのサイズに応じて、8 または 16 ビットです。選択されたデータフレームフォーマットは、送受信に適用できます。

28.3.2 SPI のスレーブモード設定

スレーブ設定では、マスタデバイスからのシリアルクロックは SCK ピンで受信されます。SPI_CR1 レジスタの BR[2:0] ビットに設定された値は、データ転送速度に影響を与えません。

注： 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。さもなければ、望ましくないデータ送信が発生することがあります。スレーブのデータレジスタは、通信クロックの最初のエッジまたは現在の通信の終了より前に、準備ができていなければならない必要があります。スレーブとマスタを有効にする前に、通信クロックの極性を定常値に設定することが必要です。

SPI をスレーブモードに設定するには、次の手順に従ってください。

手順

1. DFF ビットをセットして、8 または 16 ビットのデータフレームフォーマットを定義します。
2. CPOL ビットと CPHA ビットを選択して、データ転送とシリアルクロックの間の 4 つの関係のうちの 1 つを定義します (図 242 を参照)。データを正しく転送するには、スレーブデバイスとマスタデバイスで CPOL ビットと CPHA ビットを同じに設定する必要があります。SPI_CR2 レジスタの FRF ビットにより TI モードを選択している場合、このステップは不要です。
3. フレームフォーマット (SPI_CR1 レジスタの LSBFIRST ビットの値に応じて、MSB ファーストまたは LSB ファースト) は、マスタデバイスと同じでなければなりません。TI モードを選択している場合、このステップは不要です。
4. ハードウェアモード (スレーブ選択 (NSS) ピンの管理 (773 ページ) を参照) では、バイト送信シーケンス全体を通じて、NSS ピンをローレベル信号に接続しておく必要があります。NSS ソフトウェアモードでは、SPI_CR1 レジスタの SSM ビットをセットし、SSI ビットをクリアします。TI モードを選択している場合、このステップは不要です。
5. SPI_CR2 レジスタの FRF ビットをセットして、シリアル通信の TI モードプロトコルを選択します。
6. ピンをオルタネート機能に割り当てるには、SPI_CR1 レジスタの MSTR ビットをクリアし、SPE ビットをセットします。

この設定では、MOSI ピンはデータ入力に、MISO ピンはデータ出力になります。

送信シーケンス

データバイトは、書き込みサイクル中に送信バッファにパラレルロードされます。

スレーブデバイスが MOSI ピンでクロック信号と、データの最上位ビットを受信すると、送信シーケンスが開始されます。残りのビット (8 ビットデータフレームフォーマットでは 7 ビット、16 ビットデータフレームフォーマットでは 15 ビット) はシフトレジスタにロードされます。送信バッファからシフトレジスタにデータが送信されると、SPI_SR レジスタの TXE フラグがセットされ、SPI_CR2 レジスタの TXEIE ビットがセットされている場合は、割り込みが発生します。

受信シーケンス

レシーバの場合、データ転送が完了すると、

- シフトレジスタ内のデータが受信バッファに転送され、SPI_SR レジスタの RXNE フラグがセットされます。
- SPI_CR2 レジスタの RXNEIE ビットがセットされている場合、割り込みが発生します。

最後のサンプリングクロックエッジの後、RXNE ビットがセットされ、シフトレジスタに受信されたデータバイトのコピーが受信バッファに移されます。SPI_DR レジスタが読み出されると、SPI ペリフェラルはこのバッファリングされた値を返します。

RXNE ビットのクリアは、SPI_DR レジスタを読み出すことによって行われます。

スレーブモードでの SPI TI プロトコル

スレーブモードでは、SPI インタフェースは TI プロトコルと互換性を持ちます。SPI_CR2 レジスタの FRF ビットを使って、スレーブ SPI シリアル 通信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。

SPI_CR1 レジスタに設定される値によらず、クロックの極性と位相は TI プロトコル条件に強制的に一致させられます。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、SPI_CR1 レジスタと SPI_CR2 レジスタによる NSS 管理の設定 (SSM, SSI, SSOE など) がユーザーに対してトランスペアレントになります。

スレーブモード (図 243 : TI モード - スレーブモード、シングル送信および図 244 : TI モード - スレーブモード、連続送信) では、SPI ボーレートプリスケラを使って MISO ピン 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。任意のボーレートを使用できるため、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、ボーレートは一般に外部マスタクロックボーレートに設定されます。MISO 信号が HI-Z になるために要する時間 (t_{release}) は、内部再同期と SPI_CR1 レジスタの BR[2:0] で設定されたボーレート値に依存します。これは次式で与えられます：

$$\frac{t_{\text{baud\_rate}}}{2} + 4 \times t_{\text{pclk}} < t_{\text{release}} < \frac{t_{\text{baud\_rate}}}{2} + 6 \times t_{\text{pclk}}$$

注： 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)。

エラー割り込み (ERRIE = 1) を使ってスレーブトランスミッタ専用モードで TI フレームエラーを検出するときは、SPI_CR1 レジスタで BIDIMODE と BIDIOE に 1 を設定して SPI を 2 ライン単方向モードに設定する必要があります。BIDIMODE に 0 を設定すると、OVR が 1 に設定されます。これは、データレジスタが読み出されることがなく、エラー割り込みが常に発生するためです。これに対して BIDIMODE に 1 を設定すると、データが受信されないため、OVR は設定されません。

図 243. TI モード - スレーブモード、シングル送信

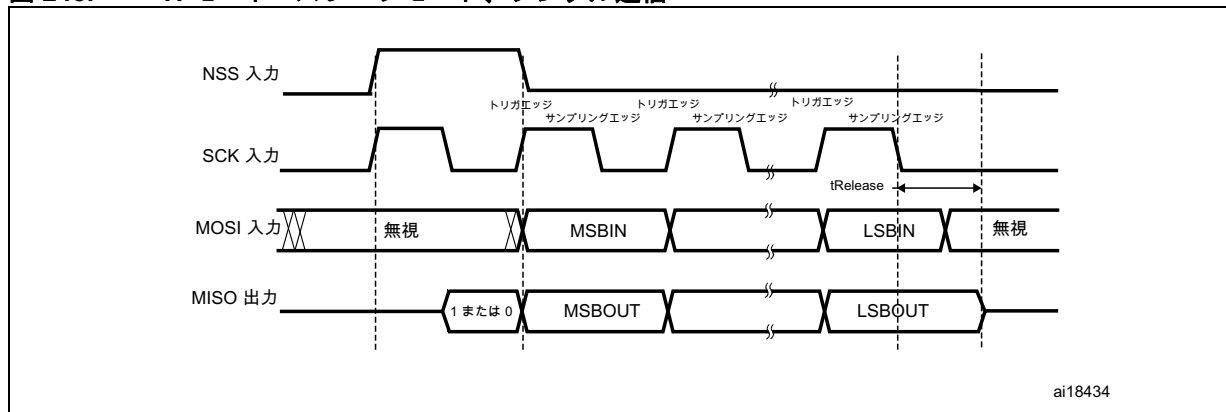
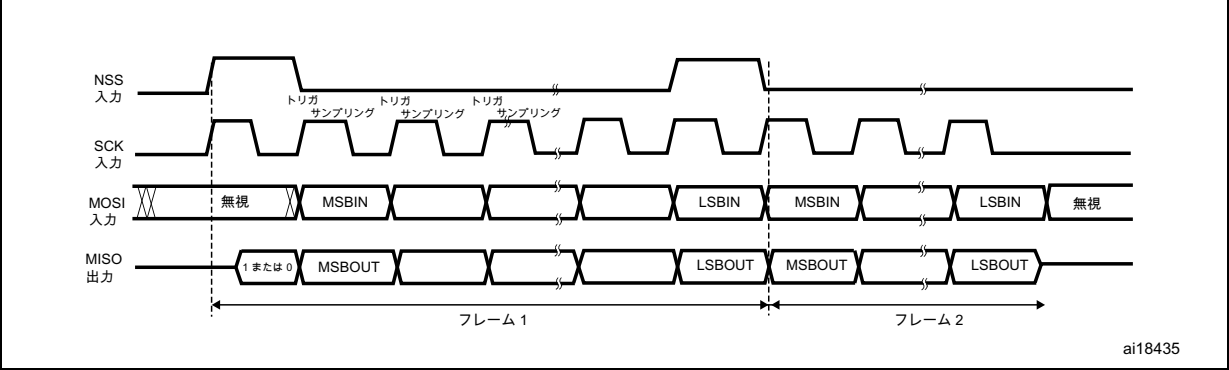


図 244. TI モード - スレーブモード、連続送信



28.3.3 SPI のマスタモード設定

マスタ設定では、シリアルクロックは SCK ピンで生成されます。

手順

1. BR[2:0] ビットを選択して、シリアルクロックのボーレートを定義します (SPI_CR1 レジスタを参照)。
2. CPOL ビットと CPHA ビットを選択して、データ転送とシリアルクロックの間の 4 つの関係のうちの 1 つを定義します (図 242 を参照)。TI モードを選択している場合、このステップは不要です。
3. DFF ビットをセットして、8 または 16 ビットのデータフレームフォーマットを定義します。
4. SPI_CR1 レジスタの 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。TI モードを選択している場合、このステップは不要です。
5. 入力モードの NSS ピンが必要な場合、ハードウェアモードで、完全なバイト送信シーケンス中に NSS ピンをハイレベル信号に接続します。NSS ソフトウェアモードで、SPI_CR1 レジスタの SSM ビットと SSI ビットをセットします。出力モードの NSS ピンが必要な場合、SSOE ビットのみをセットします。TI モードを選択している場合、このステップは不要です。
6. SPI_CR2 の FRF ビットをセットして、シリアル通信の TI プロトコルを選択します。
7. MSTR ビットと SPE ビットをセットする必要があります (これらのビットは、NSS ピンがハイレベル信号に接続されている場合にのみ、セット状態を維持します)。

この設定では、MOSI ピンはデータ出力、MISO ピンはデータ入力になります。

送信シーケンス

送信シーケンスは、送信バッファにバイトが書き込まれたときに開始されます。

データバイトは、最初のビット送信中に内部バスからシフトレジスタにパラレルロードされ、SPI_CR1 レジスタの LSBFIRST ビットに応じて MSB ファーストまたは LSB ファーストで、MOSI ピンにシリアルにシフトアウトされます。データが送信バッファからシフトレジスタに転送されると TXE フラグがセットされ、SPI_CR2 レジスタの TXEIE ビットがセットされている場合、割り込みが発生します。

受信シーケンス

レシーバの場合、データ転送が完了すると、

- シフトレジスタのデータが受信バッファに転送され、RXNE フラグがセットされます。
- SPI_CR2 レジスタの RXNEIE ビットがセットされている場合、割り込みが発生します。

最後のサンプリングクロックエッジで RXNE ビットがセットされ、シフトレジスタに受信されたデータバイトのコピーが受信バッファに移されます。SPI_DR レジスタが読み出されると、SPI ペリフェラルはこのバッファリングされた値を返します。

RXNE ビットのクリアは、SPI_DR レジスタを読み出すことで行われます。

送信が開始されると、次に送信されるデータが送信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維持できます。なお、送信バッファにデータを書き込むには、TXE フラグが 1 であることが必要です。

注： マスタが、送信と受信の間に選択解除する必要のある SPI スレーブと通信している場合、NSS ピンは GPIO として設定する必要があります。あるいは、別の GPIO を使ってソフトウェアからトグルする必要があります。



マスタモードでの SPI TI プロトコル

マスタモードでは、SPI インタフェースは TI プロトコルと互換性を持ちます。SPI_CR2 レジスタの FRF ビットを使って、マスタ SPI シリアル 通信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。

SPI_CR1 レジスタに設定される値によらず、クロックの極性と位相は TI プロトコル条件に強制的に一致させられます。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、SPI_CR1 レジスタと SPI_CR2 レジスタによる NSS 管理の設定 (SSM、SSI、SSOE など) がユーザーに対してトランスペアレントになります。

図 245 : TI モード - マスタモード、シングル転送 および 図 246 : TI モード - マスタモード、連続転送) に、マスタモードで TI モードを選択したときの SPI マスタ 通信波形を示します。

図 245. TI モード - マスタモード、シングル転送

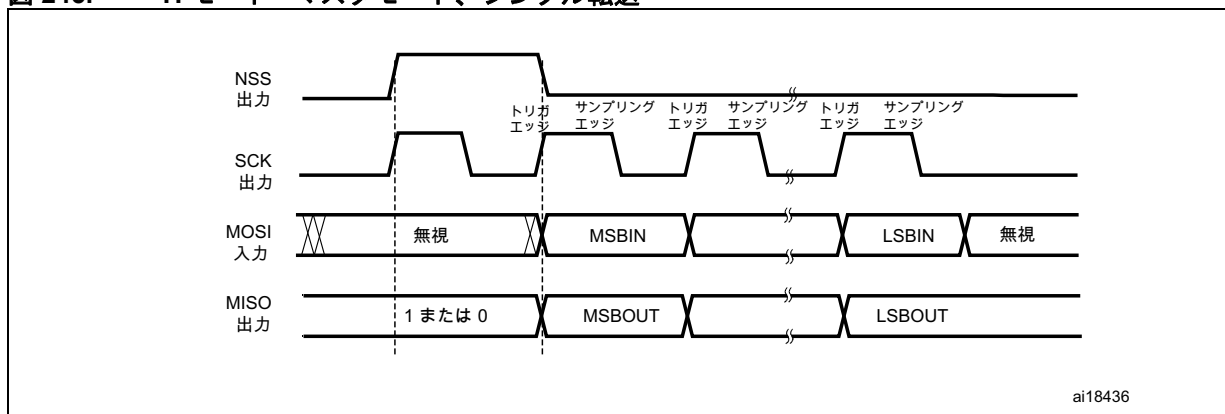
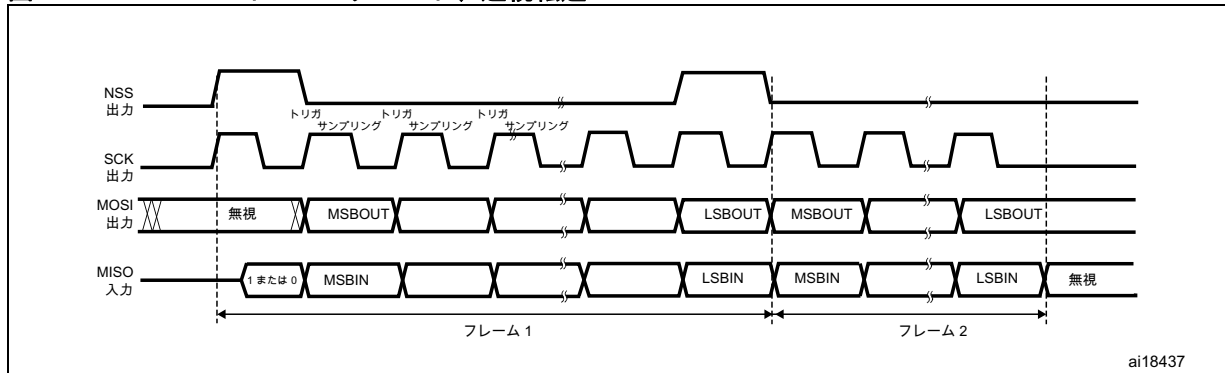


図 246. TI モード - マスタモード、連続転送



28.3.4 SPI の半二重通信への設定

SPI は、次の 2 つの設定において半二重モードで動作できます。

- 1 本のクロック線と 1 本の双方向データ線
- 1 本のクロック線と 1 本のデータ線（受信専用または送信専用）

1 本のクロック線と 1 本の双方向データ線 (BIDIMODE=1)

このモードを有効にするには、SPI_CR1 レジスタの BIDIMODE ビットをセットします。このモードでは、SCK ピンがクロックに使用され、マスタモードの MOSI ピンまたはスレーブモードの MISO ピンがデータ通信に使用されます。転送方向（入力/出力）は、SPI_CR1 レジスタの BIDIOE ビットによって選択されます。このビットが 1 の場合、データラインは出力です。そうでない場合、データラインは入力です。

1 本のクロック線と 1 本の単方向データ線 (BIDIMODE=0)

このモードでは、アプリケーションは、SPI を送信専用モードまたは受信専用モードで使用できます。

- 送信専用モードは全二重モード (BIDIMODE=0、RXONLY=0) と似ています。データは送信ピン（マスタモードの MOSI ピンまたはスレーブモードの MISO ピン）で送信され、受信ピン（マスタモードの MISO ピンまたはスレーブモードの MOSI ピン）は汎用入出力として使用できます。この場合、アプリケーションは受信バッファを無視する必要があります（データレジスタに受信したデータ値は含まれていません）。
- 受信専用モードでは、アプリケーションは、SPI_CR2 レジスタの 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。この場合、送信 I/O ピン（マスタモードの MOSI ピンまたはスレーブモードの MISO ピン）が解放されるので、他の目的に使用できます。

受信専用モードで通信を開始するには、SPI を設定して有効にします。

- マスタモードでは、通信はただちに開始され、SPE ビットがクリアされて現在の受信が停止すると、通信は停止します。このモードでは、BSY フラグを読み出す必要はありません。SPI 通信が行われているときは、常にセットされています。
- スレーブモードでは、NSS がプルダウンされていて（または NSS ソフトウェアモードで SSI ビットがクリアされていて）、SCK が動作している限り、SPI は受信を続行します。

28.3.5 データの送受信手順

受信バッファと送信バッファ

受信の場合、受信されたデータは内部の受信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送信バッファに格納されてから、送信されます。

SPI_DR レジスタの読み出しアクセスでは、受信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送信バッファに格納されます。

マスタモードでの開始シーケンス

- 全二重の場合 (BIDIMODE=0 および RXONLY=0)
 - データが SPI_DR レジスタ (送信バッファ) に書き込まれると、シーケンスが開始されます。
 - その後、データは、最初のビット送信時に送信バッファから 8 ビットシフトレジスタに同時にロードされてから、MOSI ピンに連続的にシフトアウトされます。
 - 同時に、MISO ピンで受信されたデータは、8 ビットシフトレジスタに連続的にシフトインされてから、SPI_DR レジスタ (受信バッファ) に同時にロードされます。
- 単方向受信専用モードの場合 (BIDIMODE=0 および RXONLY=1)
 - SPE=1 になると、このシーケンスはすぐに開始されます。
 - レシーバのみが有効になり、MISO ピンで受信されたデータは 8 ビットシフトレジスタに連続的にシフトインされてから、SPI_DR レジスタ (受信バッファ) に同時にロードされます。
- 双方向モードでの送信時 (BIDIMODE=1 および BIDIOE=1)
 - データが SPI_DR レジスタ (送信バッファ) に書き込まれると、シーケンスが開始されます。
 - その後、データは、最初のビット送信時に送信バッファから 8 ビットシフトレジスタに同時にロードされてから、MOSI ピンに連続的にシフトアウトされます。
 - データは受信されません。
- 双方向モードでの受信時 (BIDIMODE=1 および BIDIOE=0)
 - SPE=1 および BIDIOE=0 になると、このシーケンスはすぐに開始されます。
 - MOSI ピンで受信されたデータは、8 ビットシフトレジスタに連続的にシフトインされてから、SPI_DR レジスタ (受信バッファ) にパラレルロードされます。
 - トランスミッタは有効にされず、MOSI ピンに連続的にシフトアウトされるデータはありません。

スレーブモードでの開始シーケンス

- 全二重モードの場合 (BIDIMODE=0 および RXONLY=0)
 - スレーブデバイスが MOSI ピンでクロック信号と、データの最初のビットを受信すると、このシーケンスが開始されます。残りの 7 ビットは、シフトレジスタにロードされます。
 - 同時に、最初のビット送信時にデータは送信バッファから 8 ビットシフトレジスタに同時にロードされてから、MISO ピンに連続的にシフトアウトされます。SPI マスタデバイスが転送を開始する前に、ソフトウェアは送信すべきデータを書き込んでおく必要があります。
- 単方向受信専用モードの場合 (BIDIMODE=0 および RXONLY=1)
 - スレーブデバイスが MOSI ピンでクロック信号と、データの最初のビットを受信すると、このシーケンスが開始されます。残りの 7 ビットは、シフトレジスタにロードされます。
 - トランスミッタは有効にされず、MISO ピンに連続的にシフトアウトされるデータはありません。

- 双方向モードでの送信時 (BIDIMODE=1 および BIDIOE=1)
 - スレーブデバイスがクロック信号を受信し、送信バッファの最初のビットが MISO ピンで送信されると、このシーケンスが開始されます。
 - その後、データは、最初のビット送信時に送信バッファから 8 ビットシフトレジスタにパラレルロードされてから、MISO ピンに連続的にシフトアウトされます。SPI マスタデバイスが転送を開始する前に、ソフトウェアは送信すべきデータを書き込んでおく必要があります。
 - データは受信されません。
- 双方向モードでの受信時 (BIDIMODE=1 および BIDIOE=0)
 - スレーブデバイスが MISO ピンでクロック信号と、データの最初のビットを受信すると、このシーケンスが開始されます。
 - MISO ピンで受信されたデータは、8 ビットシフトレジスタに連続的にシフトインされてから、SPI_DR レジスタ (受信バッファ) にパラレルロードされます。
 - トランスミッタは有効にされず、MISO ピンに連続的にシフトアウトされるデータはありません。

データの送受信処理

データが送信バッファからシフトレジスタに転送されると、TXE フラグ (送信バッファエンプティ) がセットされます。これは、内部の送信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 ビットのクリアは、SPI_DR レジスタへの書き込みによって行われます。

注： **ソフトウェアは、送信バッファへの書き込みを行う前に、TXE フラグを 1 にセットする必要があります。さもなければ、以前に送信バッファに書き込まれたデータが上書きされます。**

データがシフトレジスタから受信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 (受信バッファノットエンプティ) がセットされます。これは、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 ビットのクリアは、SPI_DR レジスタを読み出すことで行われます。

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。

マスタまたはスレーブモードでの全二重送受信手順 (BIDIMODE=0 および RXONLY=0)

データを送受信するには、ソフトウェアは次の手順に従う必要があります ([図 247](#) および [図 248](#) を参照) :

1. SPE ビットを 1 にセットして、SPI を有効にします。
2. 送信する最初のデータ項目を SPI_DR レジスタに書き込みます (これによって TXE フラグがクリアされます)。
3. TXE=1 になるまで待ち、送信する 2 番目のデータ項目を書き込みます。その後、RXNE=1 になるまで待ち、SPI_DR を読み出して、最初の受信データ項目を取得します (これによって RXNE ビットがクリアされます)。n-1 個のデータを受信するまで、送受信するデータ項目ごとにこの操作を繰り返します。
4. RXNE=1 になるまで待ち、最後の受信データを読み出します。
5. TXE=1 になるまで待ち、BSY=0 になるまで待つてから、SPI を無効にします。

この手順は、RXNE または TXE フラグの立ち上がりエッジのたびに起動される専用の割り込みサブルーチンを使用しても実行できます。

図 247. 連続転送の場合におけるマスタ/全二重モード (BIDIMODE=0 および RXONLY=0) での TXE/RXNE/BSY の動作

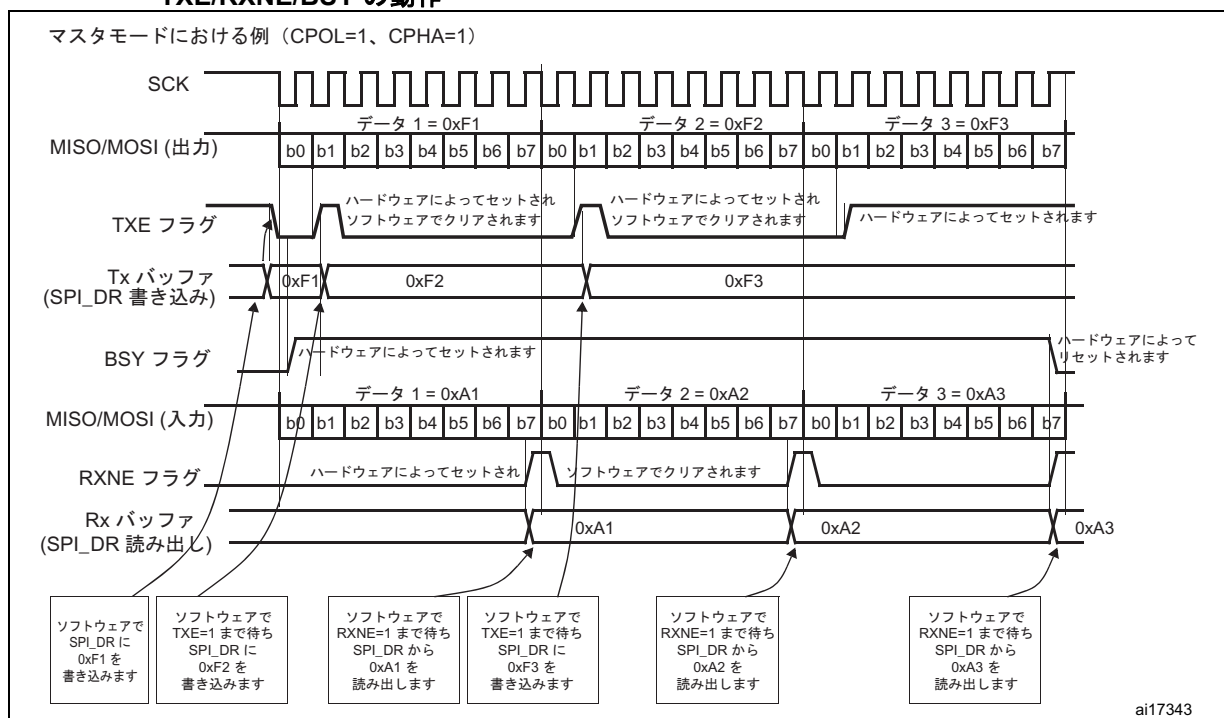
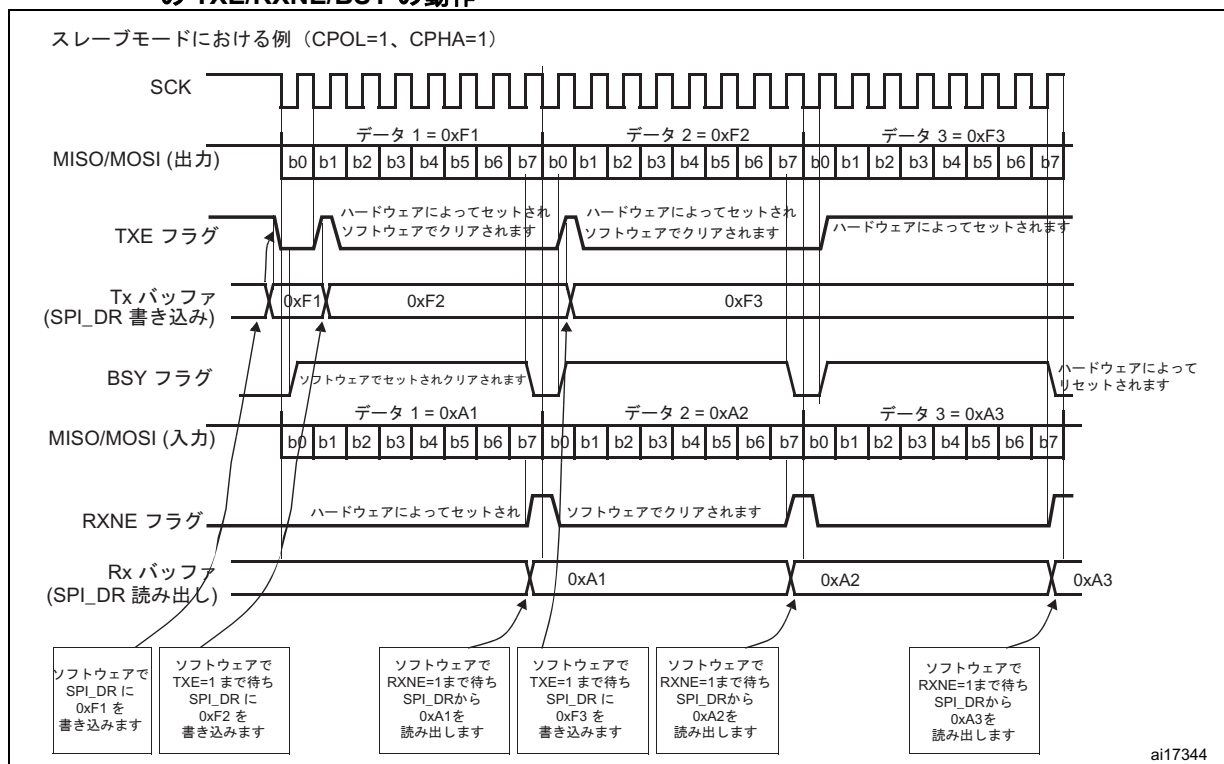


図 248. 連続転送の場合におけるスレーブ/全二重モード (BIDIMODE=0 および RXONLY=0) での TXE/RXNE/BSY の動作



送信専用手順 (BIDIMODE=0、RXONLY=0)

このモードでは、手順は下記のように短縮でき、BSY ビットを使用して送信の完了まで待つことができます (図 249 と図 250 を参照)。

1. SPE ビットを 1 にセットして、SPI を有効にします。
2. 送信する最初のデータ項目を SPI_DR 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)。
3. TXE=1 になるまで待ち、送信する次のデータ項目を書き込みます。送信するデータ項目ごとに、このステップを繰り返します。
4. SPI_DR レジスタに最後のデータ項目を書き込んだ後、TXE=1 になるまで待ち、さらに BSY=0 になるまで待ちます。これは最後のデータの送信が完了したことを示します。

この手順は、TXE フラグの立ち上がりエッジのたびに起動される専用の割り込みサブルーチンを使用しても実行できます。

注： 不連続通信時には、SPI_DR への書き込み動作と BSY ビットの設定の間に、APB の 2 クロック周期分の遅延があります。その結果、送信専用モードでは、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。

2 つのデータ項目を送信専用モードで送信した後、受信データが読み出されることはないので、SPI_SR レジスタの OVR フラグがセットされます。

図 249. 連続転送の場合におけるマスタ送信専用モード (BIDIMODE=0 および RXONLY=0) での TXE/BSY の動作

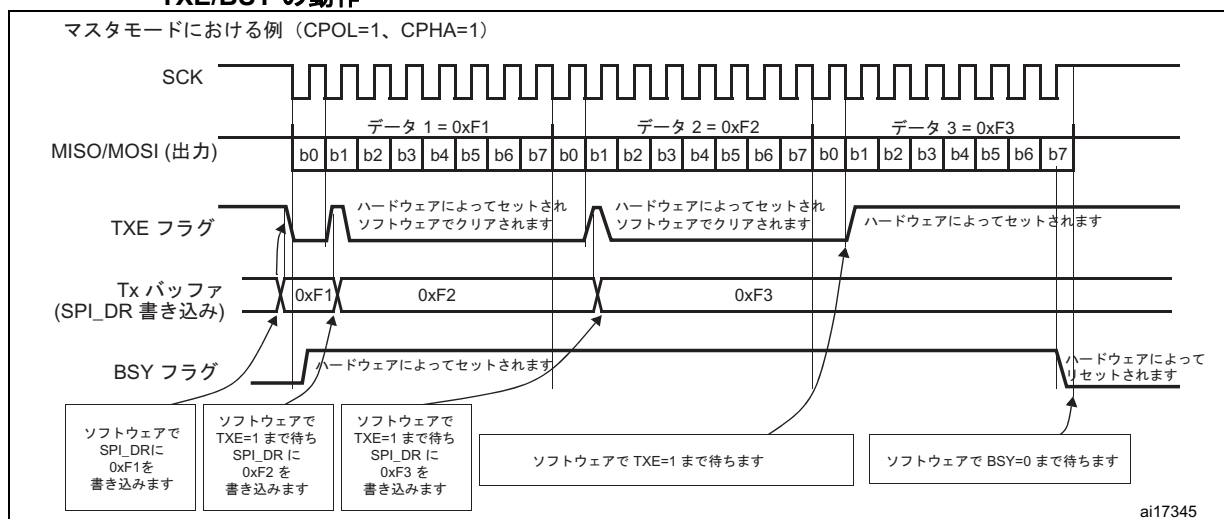
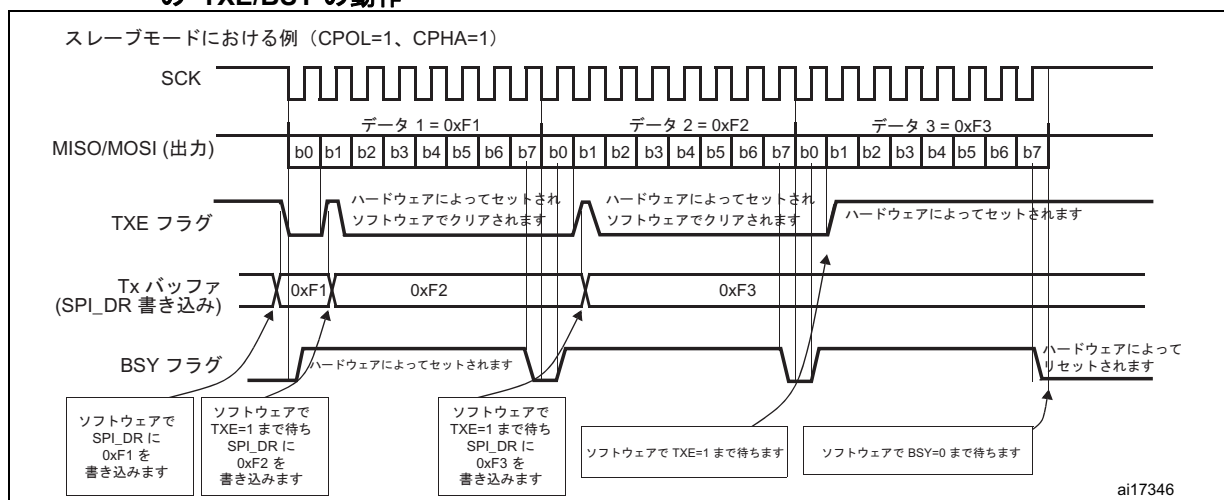


図 250. 連続転送の場合におけるスレーブ送信専用モード (BIDIMODE=0 および RXONLY=0) での TXE/BSY の動作



双方向送信手順 (BIDIMODE=1 および BIDIOE=1)

このモードでは、手順は送信専用モードの手順と似ていますが、SPI を有効にする前に、SPI_CR2 レジスタの BIDIMODE ビットと BIDIOE ビットをセットする必要があります。

単方向受信専用手順 (BIDIMODE=0 および RXONLY=1)

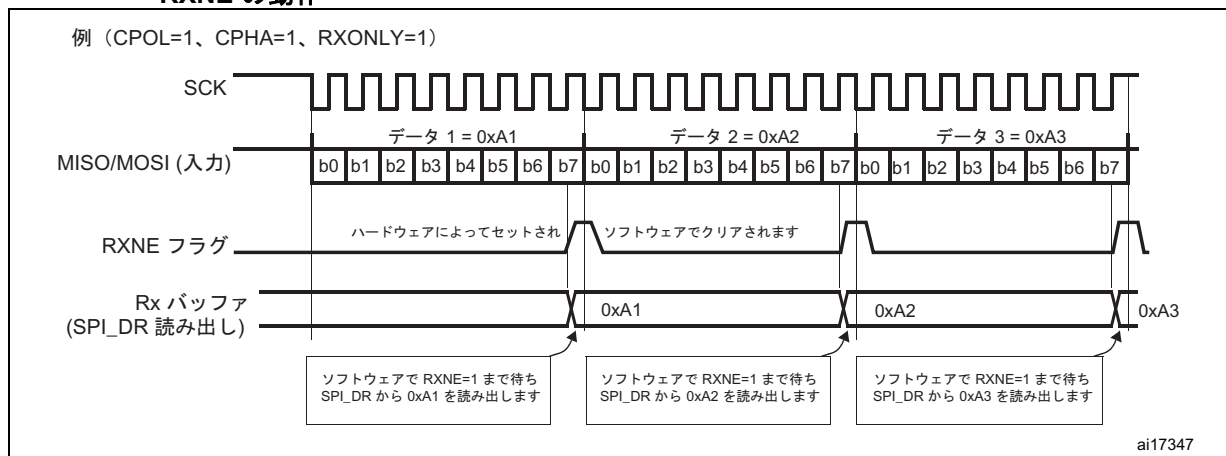
このモードでは、手順は下記のように短縮できます (図 251 を参照)：

- SPI_CR2 レジスタの RXONLY ビットをセットします。
- SPE ビットを 1 にセットすることによって、SPI を有効にします：
 - マスタモードでは、これによって SCK クロックの生成がすぐに有効になり、データは、SPI が無効にされる (SPE=0) まで連続的に受信されます。
 - スレーブモードでは、SPI マスタデバイスが NSS をローレベルに駆動して SCK クロックを生成すると、データが受信されます。
- RXNE=1 になるまで待ち、SPI_DR レジスタから受信データを読み出します (これによって RXNE ビットがクリアされます)。受信されるデータ項目ごとにこの操作を繰り返します。

この手順は、RXNE フラグの立ち上がりエッジのたびに起動される専用の割り込みサブルーチンを使用しても実行できます。

注： 最後の転送の後に SPI を無効にする必要がある場合は、[セクション 28.3.8 : SPI の無効化 \(791 ページ\)](#) に述べる推奨事項に従ってください。

図 251. 連続転送の場合における受信専用モード (BIDIRMODE=0 および RXONLY=1) での RXNE の動作



双方向受信手順 (BIDIMODE=1 および BIDIOE=0)

このモードでは、手順は受信専用モードの手順と似ていますが、SPI を有効にする前に、SPI_CR2 レジスタの BIDIMODE ビットをセットし、BIDIOE ビットをクリアする必要があります。

連続転送と不連続転送

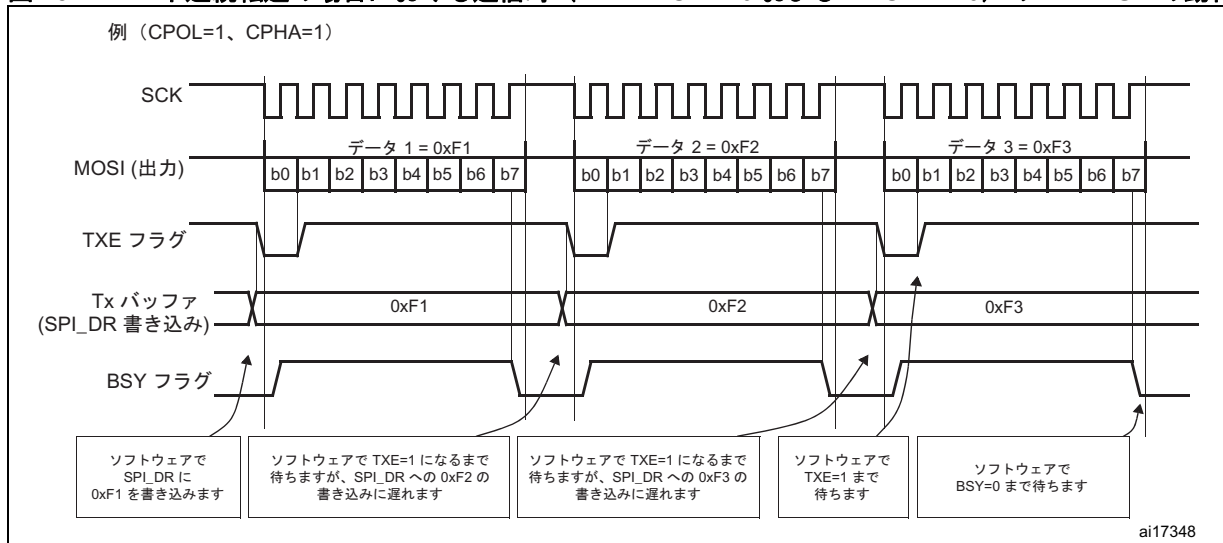
マスタモードでデータを送信するとき、ソフトウェアが十分に高速であり、TXE (または TXE 割り込み) の各立ち上がりエッジを検出し、現在のデータ転送が完了する前に SPI_DR レジスタにすぐに書き込みできる場合、通信は連続的であると言えます。この場合、各データ項目間で SPI クロックを生成する際に不連続性はなく、各データ転送間で BSY ビットがクリアされることはありません。

逆に、ソフトウェアが十分に高速でない場合、通信に若干の不連続性が生じることがあります。この場合、各データ送信間で BSY ビットがクリアされます (図 252 を参照)。

マスタ受信専用モード (RXONLY=1) では、通信は常に連続的であり、BSY フラグは常に 1 で読み出されます。

スレーブモードでは、通信の連続性は SPI マスタデバイスによって決定されます。いずれにしても、通信が連続的であっても、BSY フラグは、各転送間で SPI の 1 クロックサイクルという最小期間だけローレベルになります (図 250 を参照)。

図 252. 不連続転送の場合における送信時 (BIDIRMODE=0 および RXONLY=0) の TXE/BSY の動作



28.3.6 CRC の計算

CRC 計算機は、信頼性の高い通信を実現するために実装されました。送信データと受信データに対して、別個の CRC 計算機が実装されています。CRC は、各ビットに対してプログラミング可能な多項式を連続的に使用して計算されます。CRC は、SPI_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計算されます。

注： *SPI が提供する 2 種類の CRC 計算基準は、送受信用に選択されたデータフレームフォーマットに直接依存します (8 ビットデータでは CR8、16 ビットデータでは CRC16)。*

CRC 計算は、SPI_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。これによって CRC レジスタ (SPI_RXCRCR および SPI_TXCRCR) がリセットされます。全二重モードまたは送信専用モードでは、転送をソフトウェアから管理する場合 (CPU モード)、SPI_DR に送信される最後のデータが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。この最後のデータ転送の終わりで、SPI_TXCRCR の値が送信されます。

受信専用モードで転送がソフトウェアから管理される場合 (CPU モード)、最後から 2 番目のデータが受信された後に CRCNEXT ビットを書き込む必要があります。最終データの受信後に CRC が受信された後、CRC チェックが実行されます。

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 SPI_SR レジスタの CRCERR フラグがセットされます。

送信バッファにデータが存在する場合、CRC 値はデータバイトの送信後にのみ送信されます。CRC の送信時に、CRC 計算機はスイッチオフされ、レジスタの値は変化しません。

CRC を使用した SPI 通信は、次の手順によって可能になります。

1. CPOL、CPHA、LSBFirst、BR、SSM、SSI、および MSTR の値をプログラミングします。
2. SPI_CR1 レジスタに多項式をプログラミングします。
3. SPI_CR1 レジスタの CRCEN ビットをセットして、CRC 計算を有効にします。これによって、SPI_RXCRCR レジスタと SPI_TXCRCR レジスタがクリアされます。
4. SPI_CR1 レジスタの SPE ビットをセットして、SPI を有効にします。
5. 通信を開始し、1 つを除いてすべてのバイトまたはハーフワードが送受信されるまで、通信を維持します。
 - 全二重モードまたは送信専用モードで、転送がソフトウェアで管理される場合、最後のバイトまたはハーフワードを Tx バッファに書き込んだときに、SPI_CR1 レジスタの CRCNEXT ビットをセットし、最終バイトの送信後に CRC が送信されることを示します。
 - 受信専用モードでは、最後から 2 番目のデータを受信した直後に CRCNEXT ビットをセットして、最終データの受信終了時に、SPI が CRC フェーズに入る準備を行います。CRC 送信時には、CRC 計算は凍結されます。
6. 最終バイトまたはハーフワードの転送後、SPI は CRC 転送およびチェックのフェーズに入ります。全二重モードまたは受信専用モードでは、受信した CRC は SPI_RXCRCR 値と比較されます。値が一致しない場合、SPI_SR レジスタの CRCERR フラグがセットされ、SPI_CR2 レジスタの ERRIE ビットがセットされていれば、割り込みを生成できます。

注： SPI がスレーブモードのとき、CRC 計算を有効にするのは、クロックが安定している（つまり、クロックが定常状態にある）ときのみにしてください。さもないと、CRC 計算を間違えることがあります。実際、SPE ビットの値に関係なく、CRCEN がセットされると、CRC はすぐに SCK スレーブ入カクロックの影響を受けます。

高ビットレート周波数では、CRC の送信時に注意が必要です。CRC 転送フェーズで使用する CPU サイクル数は、できるだけ少なくする必要があります。したがって、最後のデータと CRC 受信におけるエラーを避けるために、CRC 送信シーケンスでソフトウェア関数を呼び出すことは禁止されています。つまり、CRCNEXT ビットには、最後のデータの送受信が終わるまでに書き込む必要があります。

ビットレートの高い周波数では、CPU アクセスが SPI 帯域幅に影響を与えることによる SPI 速度性能の低下を避けるため、DMA モードの使用をお勧めします。

デバイスがスレーブとして設定され、NSS ハードウェアモードが使用される場合、データフェーズと CRC フェーズの間で NSS ピンをローレベルに保持する必要があります。

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、マスタとスレーブをそれぞれの CRC 計算用に再同期するために、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（SPE = 0）。
2. CRCEN ビットをクリアします。
3. CRCEN ビットをセットします。
4. SPI を有効にします（SPE = 1）。

28.3.7 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、4 つのステータスフラグが用意されています。

送信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、送信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時にクリアされます。

受信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、受信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 の読み出し時にクリアされます。

ビジーフラグ

この 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です)。BSY フラグは、SPI の通信の状態を示します。

BSY がセットされると、SPI が通信中でビジーであることを示します。マスタモード/双方向受信モード (MSTR=1、BDM=1、および BDOE=0) 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 SPI を無効にして停止モードに入りたい (またはペリフェラルクロックを無効にしたい) 場合、BSY フラグは転送の終わりを検出するのに役立ちます。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、下記の手順を厳しく順守する必要があります。

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 フラグは転送が開始されるとセットされます。ただし、マスタモード/双方向受信モード (MSTR=1、BDM=1、および BDOE=0) の場合を除きます。

BSY フラグは次の場合にクリアされます。

- 転送が終わったとき (通信が連続的である場合はマスタモードを除きます)
- SPI が無効にされたとき
- マスタモード障害が発生したとき (MODF=1)

通信が連続的でないとき、BSY フラグは各通信間でローレベルになります。

通信が連続的なとき：

- マスタモードでは、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 SPI 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。

28.3.8 SPI の無効化

転送が終了すると、アプリケーションは SPI ペリフェラルを無効にすることによって、通信を停止することができます。これには SPE ビットをクリアします。

設定によっては、転送が行われているときに SPI を無効にして停止モードに入ると、現在の転送内容が破壊されたり、BSY フラグが信頼できなくなることがあります。

このような影響を回避するには、SPI を無効にするとき、以下の手順を順守することをお勧めします。

マスタまたはスレーブの全二重モード (BIDIMODE=0、RXONLY=0)

1. RXNE=1 になるまで待つから、最後のデータを受信します。
2. TXE=1 になるまで待ちます。
3. 次に、BSY=0 になるまで待ちます。
4. SPI を無効にし (SPE=0)、結果的に停止モードに入ります (あるいは、ペリフェラルクロックを無効にします)。

マスタまたはスレーブの単方向送信専用モード (BIDIMODE=0、RXONLY=0) または双方向送信モード (BIDIMODE=1、BIDIOE=1)

SPI_DR レジスタに最後のデータが書き込まれた後：

1. TXE=1 になるまで待ちます。
2. 次に、BSY=0 になるまで待ちます。
3. SPI を無効にし (SPE=0)、結果的に停止モードに入ります (あるいは、ペリフェラルクロックを無効にします)。

マスタの単方向受信専用モード (MSTR=1、BIDIMODE=0、RXONLY=1) または双方向受信モード (MSTR=1、BIDIMODE=1、BIDIOE=0)

このケースは、SPI が新しい転送を開始しないように、特別の方法で管理する必要があります。次のシーケンスは SPI モトローラ設定に対してのみ有効です (FRF ビットに 0 を設定)：

1. RXNE=1 (n-1) の最後から 2 番目の出現を待ちます。
2. ソフトウェアループを使用して SPI の 1 クロックサイクルの間待つから、SPI を無効にします (SPE=0)。
3. 次に、最後の RXNE=1 になるまで待つから、停止モードに入ります (あるいは、ペリフェラルクロックを無効にします)。

SPI が TI モードに設定されている場合 (FRF ビットを 1 に設定)、次の手順を守り、SPI を無効化する際に NSS 上に不要なパルスが発生するのを防止する必要があります。

1. RXNE=1 (n-1) の最後から 2 番目の出現を待ちます。
2. ソフトウェアループを使って、次のウインドウフレームで SPI を無効化します (SPE = 0)：
 - 少なくとも 1 SPI クロックサイクル経過後で、
 - LSB データ転送の始まる前。

注： マスタの双方向受信モード (MSTR=1、BDM=1、および BDOE=0) では、BSY フラグは転送時にローレベルに保持されます。



スレーブの受信専用モード (MSTR=0、BIDIMODE=0、RXONLY=1) または双方向受信モード (MSTR=0、BIDIMODE=1、BIDOE=0)

1. SPI はいつでも無効にできます (SPE=1 の書き込み)。現在の転送が完了してから、SPI は実質的に無効になります。
2. 次に、停止モードに入りたい場合は、まず BSY = 0 になるまで待ってから、停止モードに入ります (あるいは、ペリフェラルクロックを無効にします)。

28.3.9 DMA (Direct Memory Addressing) を使用する SPI 通信

SPI が最大速度で動作するには、SPI に送信用のデータを供給し、受信バッファに受信したデータを読み出してオーバーランを回避する必要があります。転送を容易にするため、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。

SPI_CR2 レジスタのイネーブルビットを有効にすると、DMA アクセスがリクエストされます。送信バッファと受信バッファには、別個のリクエストを発行する必要があります (図 253 と 図 254 を参照)。

- 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 SPI_DR レジスタに書き込みます (これによって TXE フラグがクリアされます)。
- 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 SPI_DR レジスタを読み出します (これによって RXNE フラグがクリアされます)。

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、受信され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 チャンネルだけを有効にすることができます。

送信モードで、DMA が送信されるすべてのデータを書き込んだとき (DMA_ISR レジスタのフラグ TCIF 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。この操作は、SPI を無効にしたり STOP モードに入る前に、最後の送信内容の破壊を避けるために必要です。ソフトウェアは、まず TXE=1 になるまで待ってから、BSY=0 になるまで待つ必要があります。

注： *不連続通信時には、SPI_DR への書き込み動作と BSY ビットの設定の間に、APB の 2 クロック周期分の遅延があります。結果として、最後のデータを書き込んだ後、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。*

図 253. DMA を使用した送信

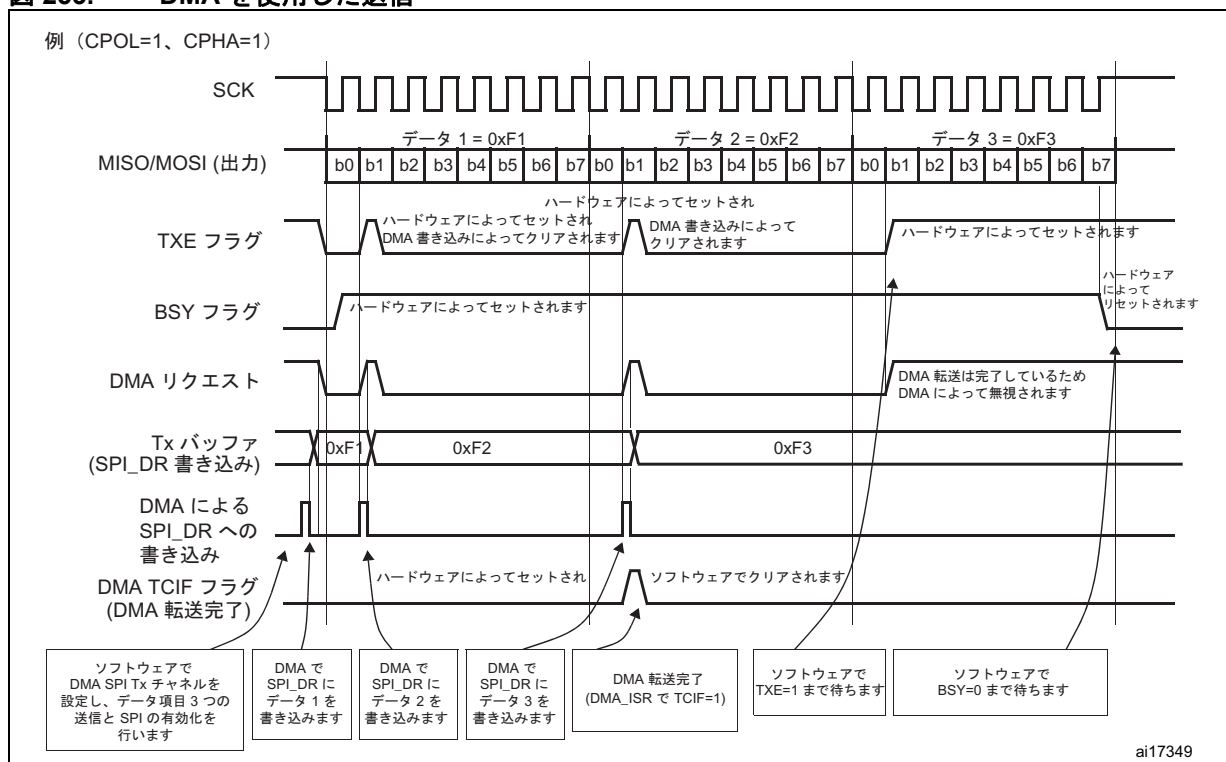
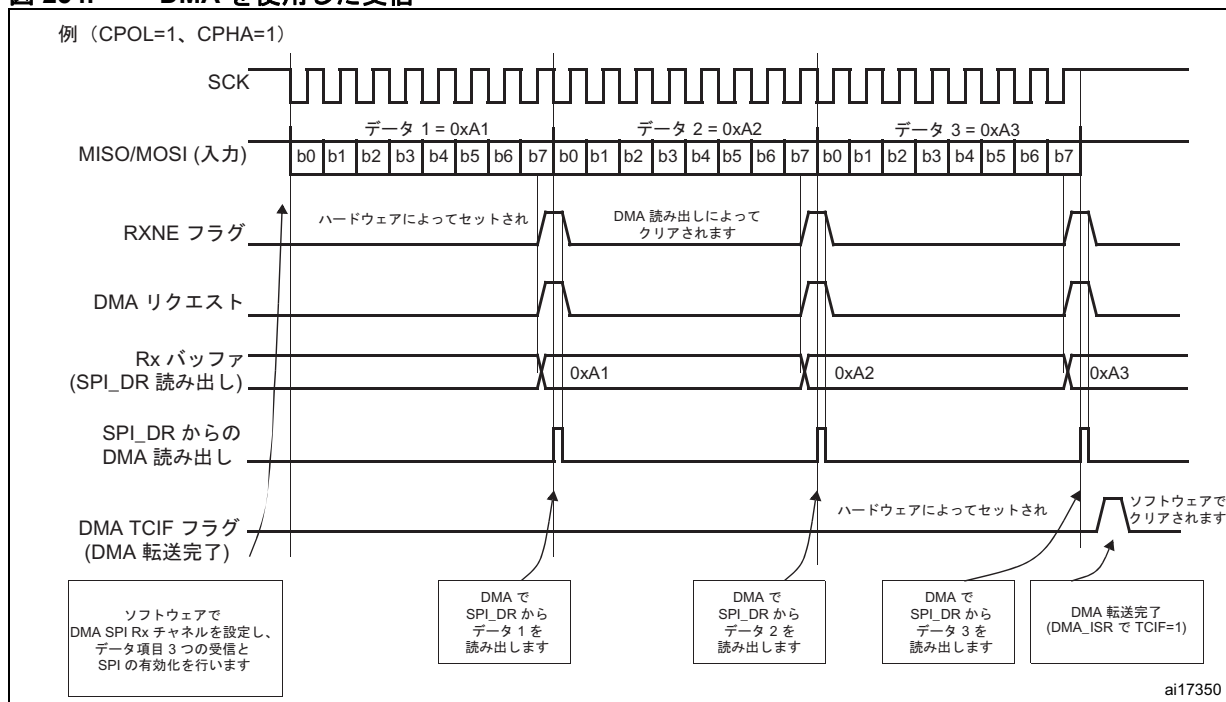


図 254. DMA を使用した受信



CRC 付きの DMA 機能

CRC 通信付きの DMA モードで SPI 通信が有効にされると、CRCNEXT ビットを使用することなく、通信の最後で CRC の送受信が自動的に行われます。CRC 受信後、SPI_DR レジスタから CRC を読み出し、RXNE フラグをクリアする必要があります。

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 SPI_SR レジスタの CRCERR フラグがセットされます。

28.3.10 エラーフラグ

マスタモードフォールト (MODF)

マスタモードフォールトは、マスタデバイスが NSS ピンをローにしたとき (NSS ハードウェアモード)、または SSI ビットをローにしたとき (NSS ソフトウェアモード) に発生し、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 ペリフェラル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 割り込みが発生します。
- 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に、SPI_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、SPI_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 ビットをセットできません。ただし、マルチマスタ設定では、デバイスはこの MODF ビットをセットした状態でスレーブモードになることができます。この場合、MODF ビットは、システム制御に関してマルチマスタ競合が生じた可能性を示します。割り込みルーチンを使用して、リセットを行ったりデフォルト状態に復帰することによって、この状態からクリーンに回復できます。

オーバーラン条件

オーバーラン条件が発生するのは、マスタデバイスがデータバイトを送信し、スレーブデバイスが以前に送信されたデータバイトに起因する RXNE ビットをクリアしなかった場合です。オーバーラン条件が発生すると、

- 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発生します。

この場合、レシーババッファは、マスタデバイスから新しく受信したデータによって更新されません。このバイトは、SPI_DR レジスタの読み出しによって返されます。その後送信されたすべてのバイトは失われます。

OVR ビットのクリアは、SPI_DR レジスタを読み出した後に SPI_SR レジスタを読み出すことによって行われます。

CRC エラー

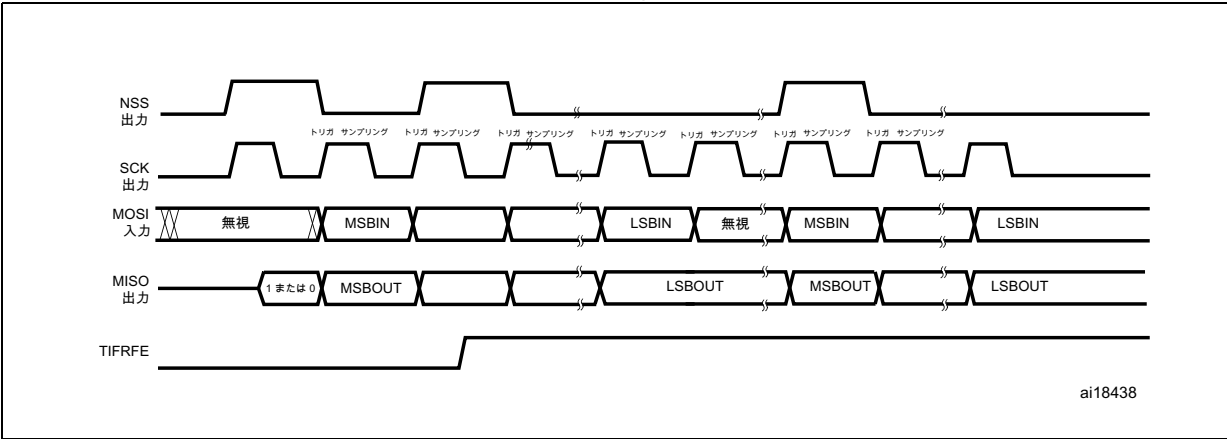
このフラグを使用して、SPI_CR1 レジスタの CRCEN ビットがセットされているときに受信された値の有効性を検証します。シフトレジスタに受信された値が SPI_RXCRCR の値と一致しなかった場合は、SPI_SR レジスタの CRCERR フラグがセットされます。

TI モードフレームフォーマットエラー

SPI がスレーブモードとして機能し、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、SPI_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 は無効にされずにこの NSS パルスは無視され、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、データは破損した可能性があります。

SPI_SR レジスタを読み出すと、FRE がクリアされます。ビット ERRIE がセットされた場合、NSS エラーが検出されると、割り込みが発生します。この場合データデータのー貫性が保証されなくなり、スレーブ SPI が再度有効化されたとき、マスタから通信が再起動されるため、SPI を無効にする必要があります。

図 255. TI モードフレームフォーマットエラーの検出



28.3.11 SPI 割り込み

表 158. 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
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
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 エラーフラグ	CRCERR	
TI frame format error	FRE	ERRIE

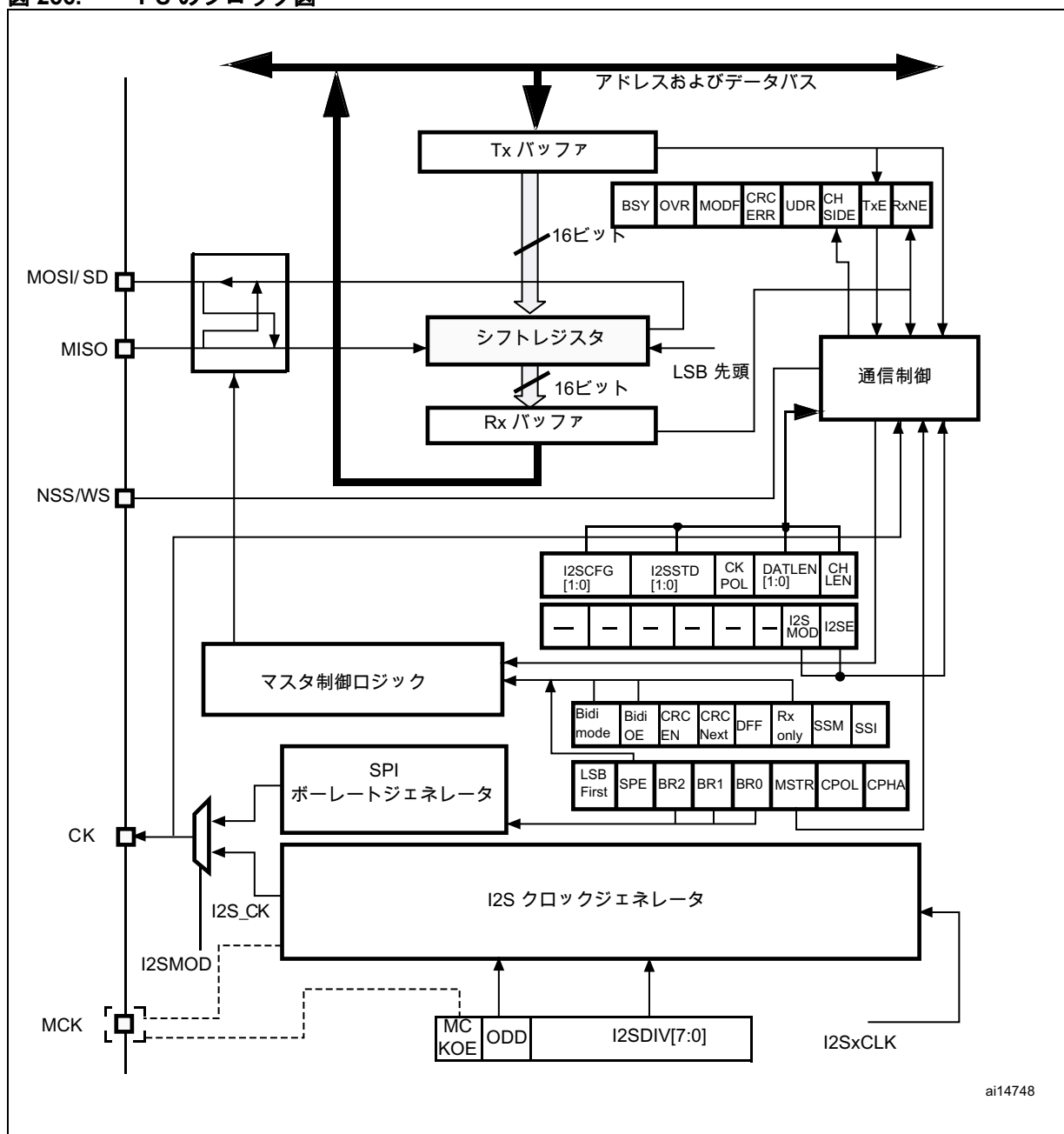
28.4 I²S 機能の詳細

Cat.1とCat.2 デバイスではI²S オーディオプロトコルは使用できません。

28.4.1 I^2S の概要

I^2S のブロック図を [図 256](#) に示します。

図 256. I^2S のブロック図



SPI_I2SCFGR レジスタの I2SMOD ビットをセットして I²S 機能を有効にすると、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。このインタフェースは、SPI とほぼ同じピン、フラグ、および割り込みを使用します。

I²S は SPI と 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 つの時間マルチプレクスデータチャネルを送受信します (半二重モードの場合)。
- 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) : 別個に配置され、I²S がマスタモードに設定され (かつ、SPI_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発生させます。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 SPI_I2SPR にリンクされ、もう 1 つは汎用 I²S 設定レジスタ SPI_I2SCFGR (オーディオ標準、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。

SPI_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、SPI_CR2 レジスタの SSOE ビットと SPI_SR 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 (SPI_DR) を使用します。

28.4.2 サポートされるオーディオプロトコル

3 線式バスでは、一般に 2 つのチャネル (右チャネルと左チャネル) で時間マルチプレクス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。ただし、送受信には 1 つの 16 ビットレジスタしかありません。したがって、注目する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 SPI_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 ビットパケットで拡張された 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 回の読み出し/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、SPI_DR との間で 2 つの 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 つの DMA 動作) を必要とします。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 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標準に対して、最上位ビットは常に最初に送信されます (MSB ファースト)。

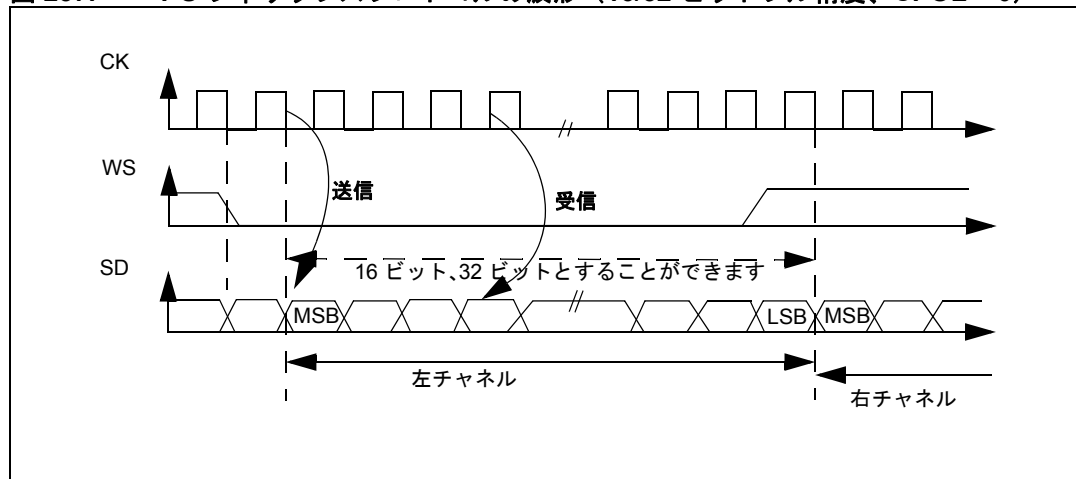


I²S インタフェースは、SPI_I2SCFGR レジスタの I2SSTD[1:0] と PCMSYNC ビットを使用して設定可能な 4 つのオーディオ標準をサポートします。

I²S フィリップス標準

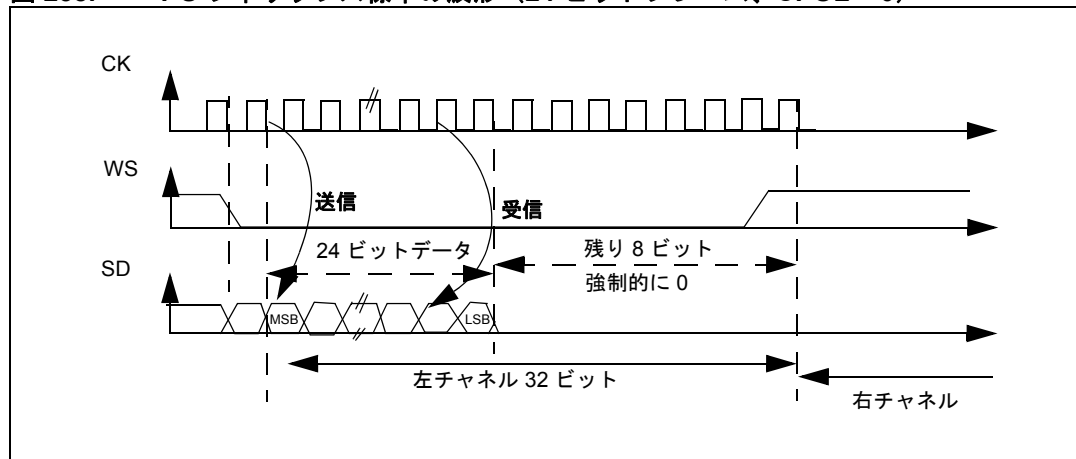
この標準では、どのチャンネルが送信されているかを示すために WS 信号を使用します。この信号が有効になってから CK の 1 クロックサイクル後に最初のビット (MSB) が使用可能になります。

図 257. I²S フィリップスプロトコルの波形 (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。

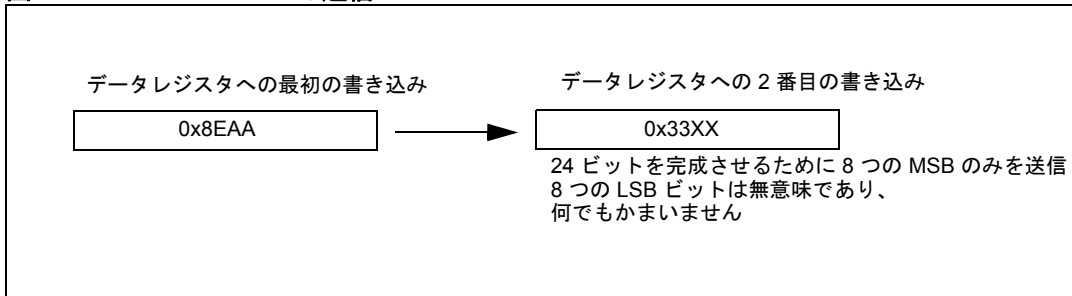
図 258. I²S フィリップス標準の波形 (24 ビットフレーム、CPOL = 0)



このモードでは、SPI_DR に対して 2 つの書き込みまたは読み出し動作が必要です。

- 送信モードでは :
0x8EAA33 を送信する必要がある場合 (24 ビット) :

図 259. 0x8EAA33 の送信



- 受信モードでは :
データ 0x8EAA33 が受信される場合 :

図 260. 0x8EAA33 の受信

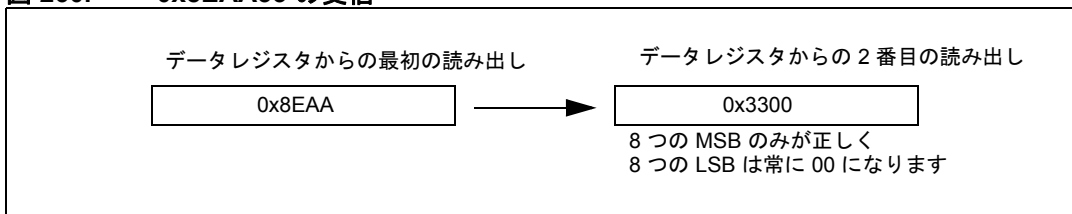
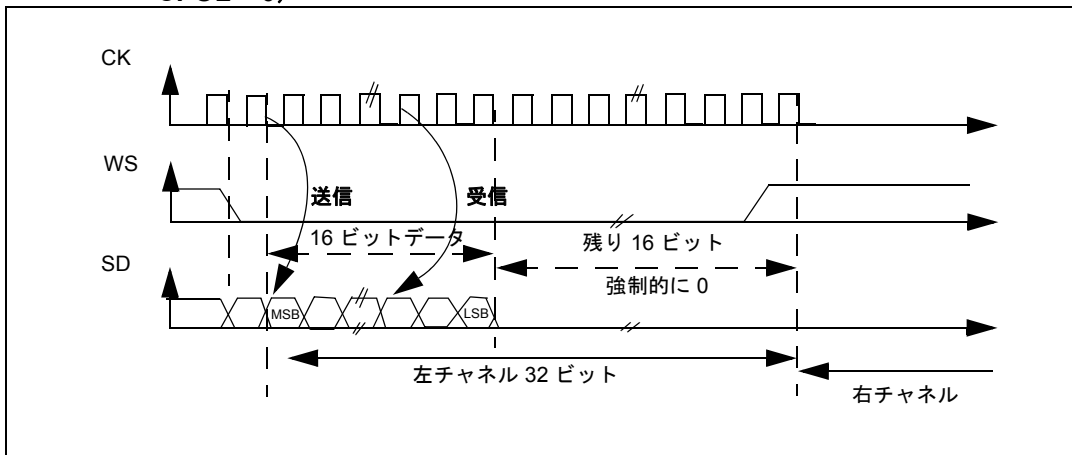


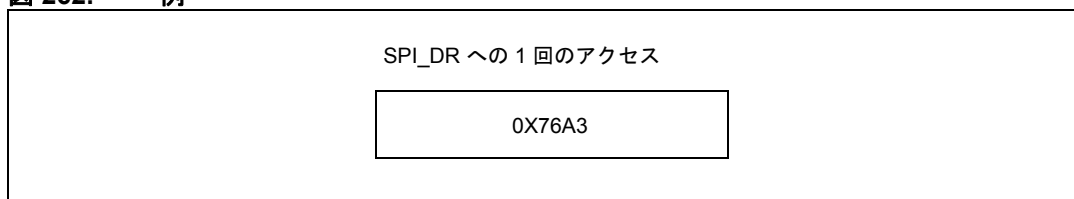
図 261. I²S フィリップス標準 (32 ビットパケットフレームへ 16 ビット拡張、CPOL = 0)



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、SPI_DR 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 262](#) に示す動作が要求されます。

図 262. 例



送信では、SPI_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、SPI_DR に新しい送信値をロードするために割り込みが発生します。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としても行われます。

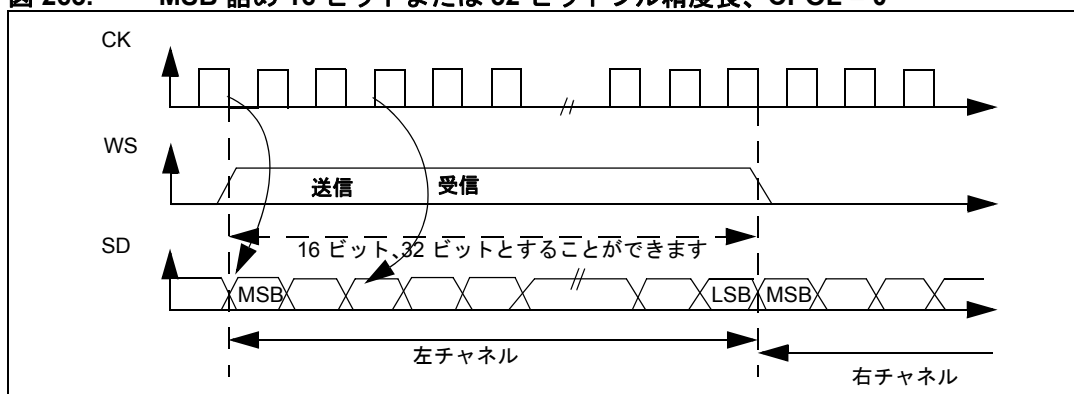
受信では、RXNE フラグがセットされ、可能ならば、最初の 16 MSB ハーフワードの受信時にその割り込みが発生します。

このように、2 つ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 詰め標準

この標準では、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 263. 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 264. MSB 詰め 24 ビットフレーム長、CPOL = 0

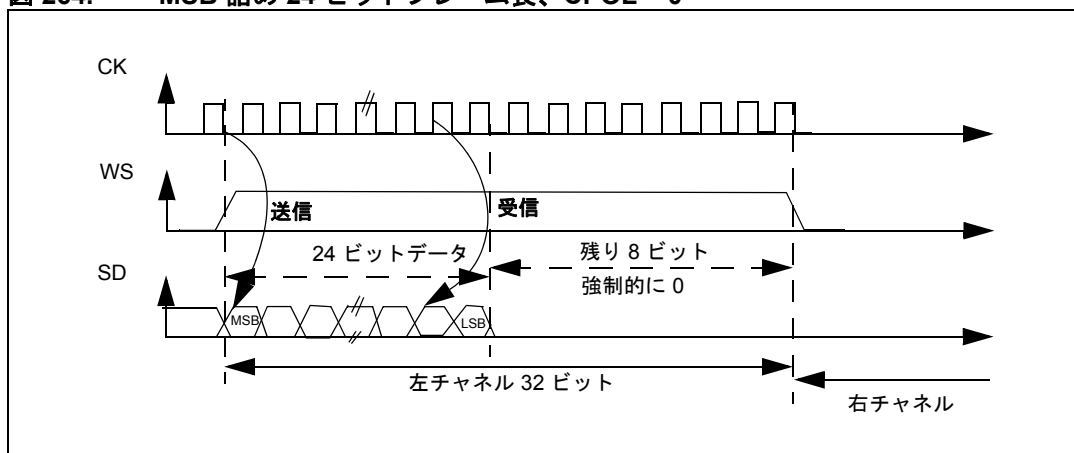
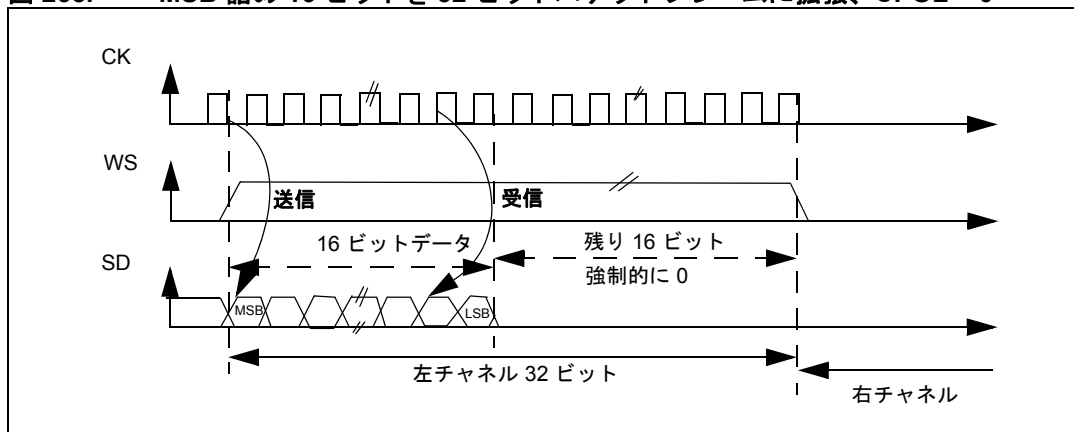


図 265. MSB 詰め 16 ビットを 32 ビットパケットフレームに拡張、CPOL = 0



LSB 詰め標準

この標準は、MSB 詰め標準と似ています (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 266. LSB 詰め 16 ビットまたは 32 ビットのフル精度、CPOL = 0

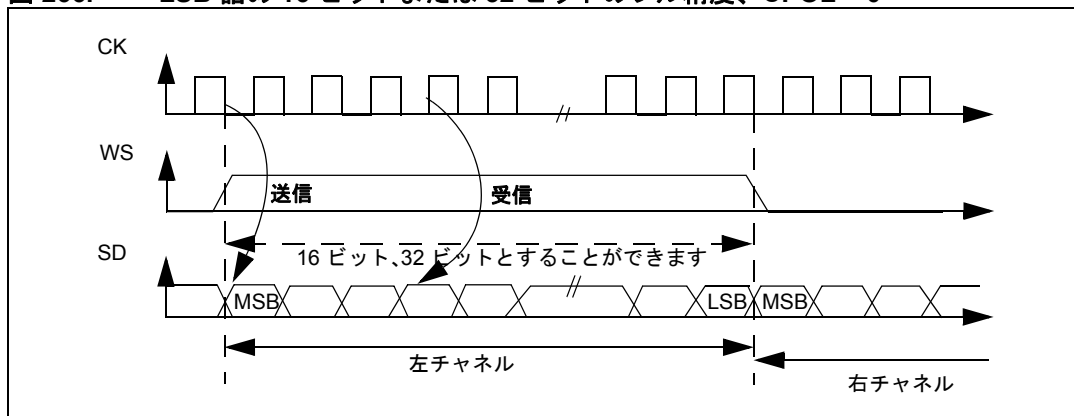
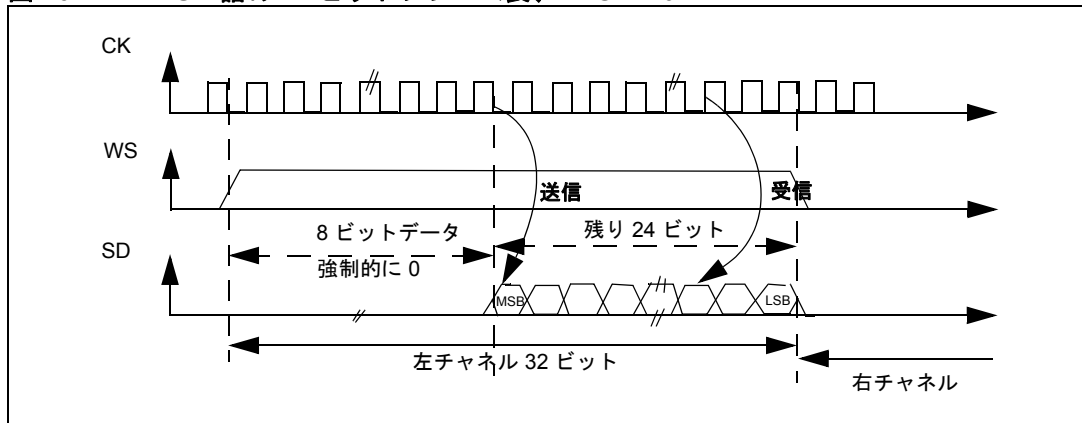


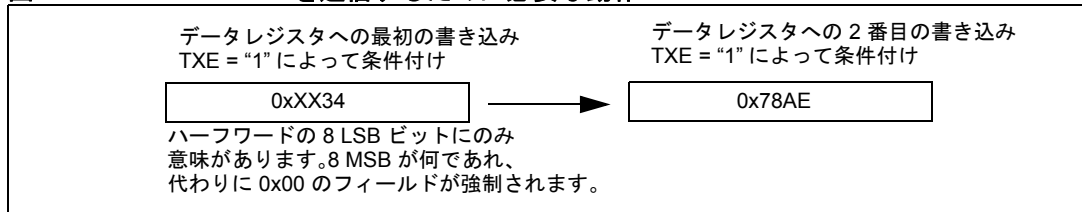
図 267. LSB 詰め 24 ビットフレーム長、CPOL = 0



- 送信モードでは：

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 SPI_DR レジスタへの 2 つの書き込み動作が必要です。この動作を次に示します。

図 268. 0x3478AE を送信するために必要な動作



- 受信モードでは：

データ 0x3478AE が受信される場合、RXNE イベントごとに SPI_DR から連続する 2 つの読み出し動作が必要です。

図 269. 0x3478AE の受信に必要な動作

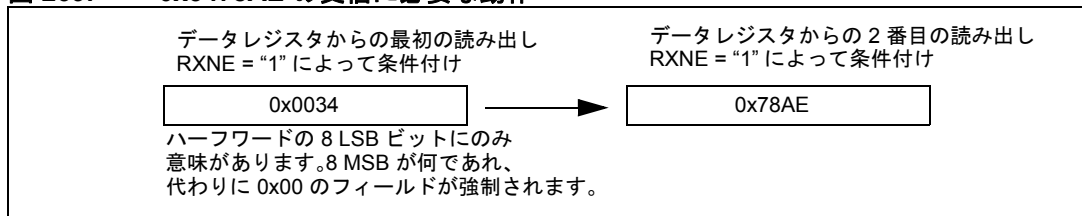
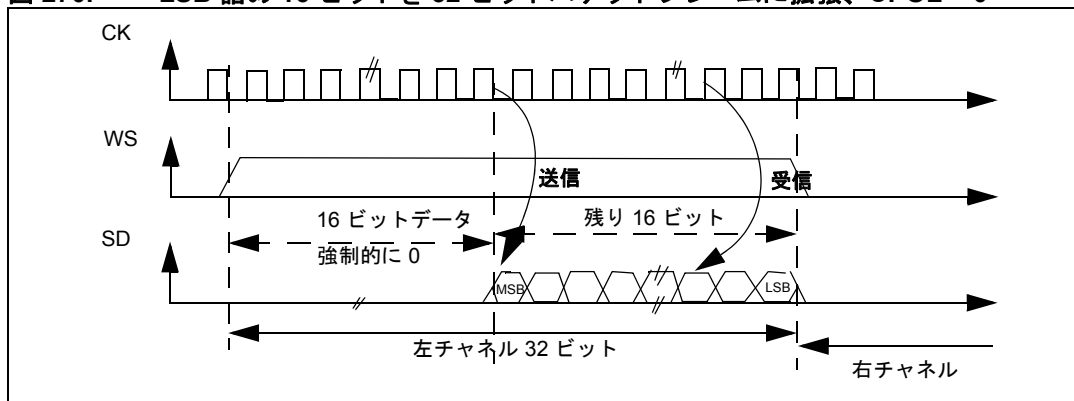


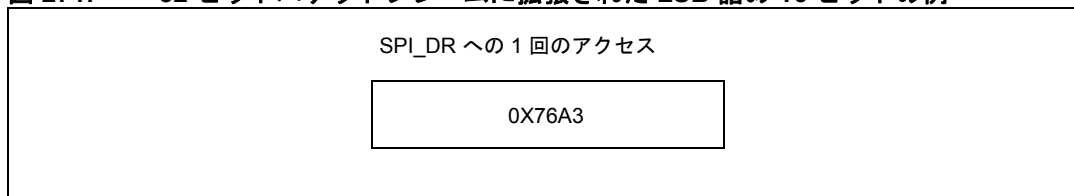
図 270. LSB 詰め 16 ビットを 32 ビットパケットフレームに拡張、CPOL = 0



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、SPI_DR 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データが 0x76A3 (32 ビットに拡張された 0x0000 76A3) である場合、[図 271](#)に示す動作が要求されます。

図 271. 32 ビットパケットフレームに拡張された LSB 詰め 16 ビットの例



送信モードでは、TXE がアサートされると、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 が再びアサートされます。

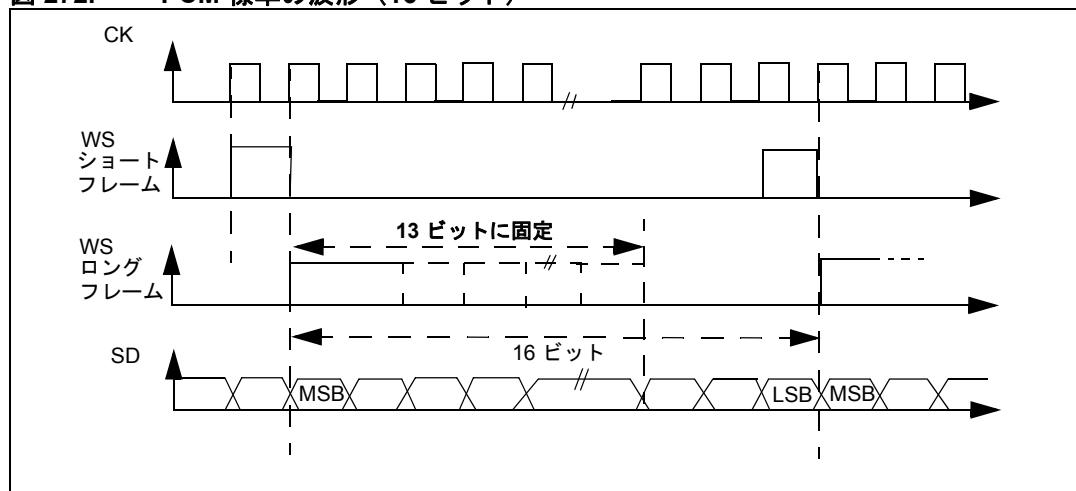
受信モードでは、0x0000 フィールドではない有効なハーフワードが受信されると、すぐに RXNE がアサートされます。

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 つ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 標準

PCM 標準では、チャンネルサイド情報を使用する必要はありません。SPI_I2SCFGR 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）を使用し、設定することができます。

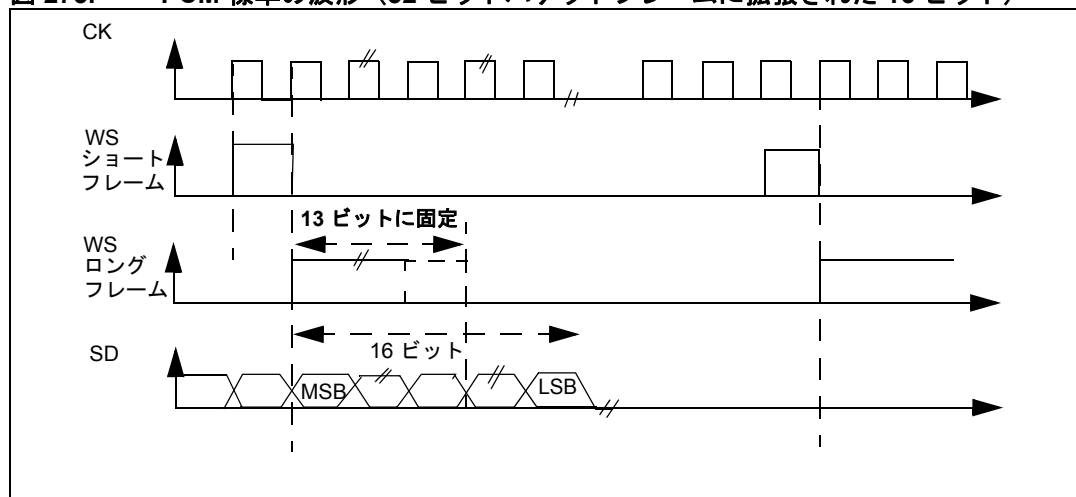
図 272. PCM 標準の波形（16 ビット）



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 273. PCM 標準の波形（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（SPI_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。

28.4.3 クロックジェネレータ

I²S ビットレートにより、I²S データラインと I²S クロック信号周波数が決まります。

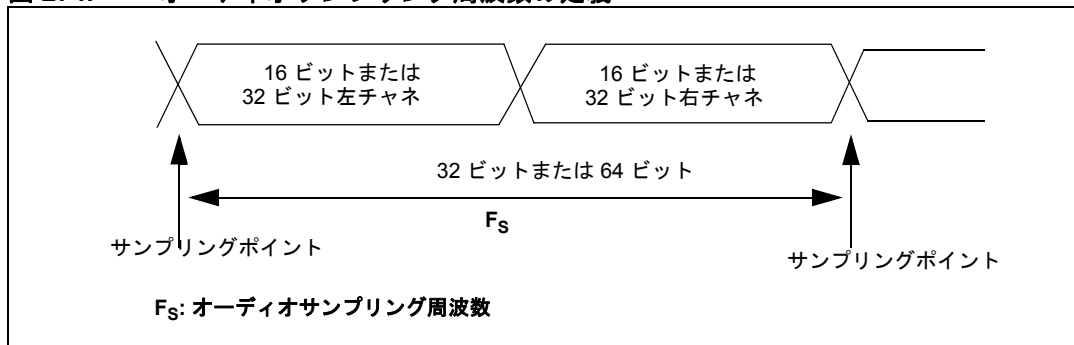
I²S ビットレート = チャンネルごとのビット数 × チャンネル数 × オーディオサンプリング周波数

16 ビットオーディオ、左チャンネル、および右チャンネルでは、I²S ビットレートは次のように計算されます。

$$I^2S \text{ ビットレート} = 16 \times 2 \times F_S$$

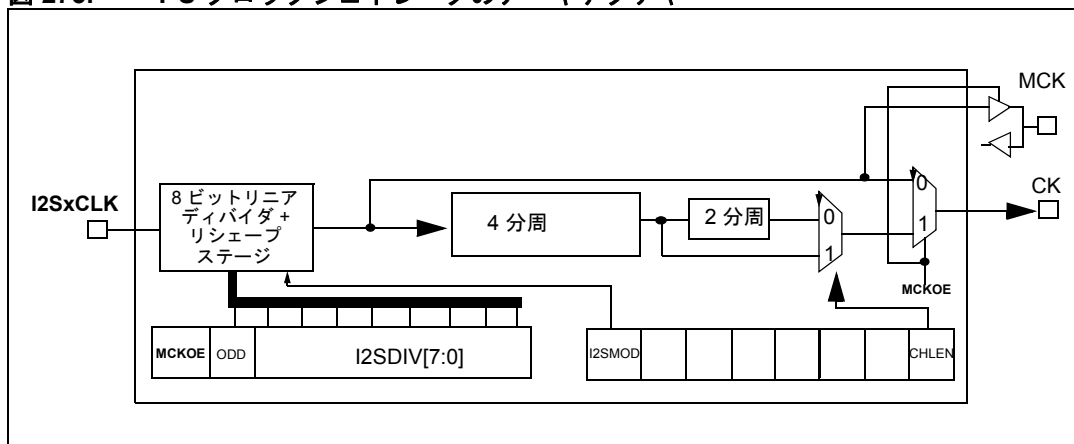
パケット長が 32 ビットの場合、I²S ビットレート = 32 × 2 × F_S となります。

図 274. 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、希望するプロトコル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ミングする必要があります。

図 275. I²S クロックジェネレータのアーキテクチャ



1. x は 2 または 3 です。

図 274 に、通信クロックのアーキテクチャを示します。I2SxCLK ソースはシステムクロックです (HSI、HSE、または PLL によって提供され、AHB クロックを供給します)。

オーディオサンプリング周波数は、96 kHz、48 kHz、44.1 kHz、32 kHz、22.05 kHz、16 kHz、11.025 kHz、または 8 kHz (あるいは、この範囲内の他の任意の値) とすることができます。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のプログラミングを行う必要があります。

マスタクロックが生成される場合 (SPI_I2SPR レジスタの MCKOE ビットをセット) :

$$F_S = I2SxCLK / [(16 \times 2) \times ((2 \times I2SDIV) + ODD) \times 8] \dots \text{チャンネルフレームが 16 ビット幅の場合}$$

$$F_S = I2SxCLK / [(32 \times 2) \times ((2 \times I2SDIV) + ODD) \times 4] \dots \text{チャンネルフレームが 32 ビット幅の場合}$$

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 159 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 : これ以外の設定でも、最適なクロック精度を実現することは可能です。

表 159. 標準の 8 MHz HSE を使用したオーディオ周波数精度 (Cat.3、Cat.4、Cat.5 デバイスの場合)

データ長	I2SDIV	I2SODD	MCLK	目標 fs (Hz)	実際の fs (kHz)	エラー
16	5	0	なし	96000	100	4.1667%
32	2	0	なし	96000	100	4.1667%
16	10	1	なし	48000	47.619	0.7937%
32	5	0	なし	48000	50	4.1667%
16	11	1	なし	44100	43.478	1.4098%
32	5	1	なし	44100	45.454	3.0715%
16	15	1	なし	32000	32.258	0.8065%
32	8	0	なし	32000	31.25	2.3430%
16	22	1	なし	22050	22.222	0.7811%
32	11	1	なし	22050	21.739	1.4098%
16	31	1	なし	16000	15.873	0.7937%
32	15	1	なし	16000	16.129	0.8065%
16	45	1	なし	11025	10.989	0.3264%
32	22	1	なし	11025	11.111	0.7811%
16	62	1	なし	8000	8	0.0000%
32	31	1	なし	8000	7.936	0.7937%
16	2	0	あり	32000	31.25	2.3430%
32	2	0	あり	32000	31.25	2.3430%
16	3	0	あり	22050	20.833	5.5170%
32	3	0	あり	22050	20.833	5.5170%
16	4	0	あり	16000	15.625	2.3428%
32	4	0	あり	16000	15.625	2.3428%
16	5	1	あり	11025	11.363	3.0715%
32	5	1	あり	11025	11.363	3.0715%
16	8	0	あり	8000	7.812	2.3428%
32	8	0	あり	8000	7.812	2.3428%

注 : この表は、さまざまなクロック設定に対する値の例を示すに過ぎません。専用 HSE クロック値によるその他の設定では、最適クロック精度が可能です。

0 エラー精度を得るためには、I²S サンプリングレートは、目標 fs ではなく実際の fs に基づく必要があります。

28.4.4 I²S マスタモード

マスタモードで、I²S を送信および受信に設定することができます。これはシリアルクロックは CK ピンまたはワードセレクト信号 WS 上で生成されることを意味します。マスタクロック (MCK) は、SPI_I2SPR レジスタの MCKOE ビットのおかげで、出力にすることも、出力にしないこともできます。

手順

1. 適切なプロトコルサンプル周波数に到達するシリアルクロックボーレートを定義するため、SPI_I2SPR レジスタの I2SDIV[7:0] ビットを選択します。SPI_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 オーディオ部品にマスタクロック MCK を供給する必要がある場合、SPI_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 28.4.3 : クロックジェネレータ](#)を参照)。
3. SPI_I2SCFGR の I2SMOD ビットをセットして I²S 機能を有効にし、I2SSTD[1:0] と PCMSYNC ビットを通じて I²S 標準を、DATLEN[1:0] ビットを通じてデータ長を、CHLEN ビットを設定してチャンネルごとのビット数を、それぞれ選択します。SPI_I2SCFGR レジスタの I2SCFG[1:0] ビットを通じて 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. 必要ならば、SPI_CR2 レジスタへの書き込みによって、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. SPI_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。SPI_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、送信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。

おそらく、送信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します。データが送信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送信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。

フルフレームとは、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パーシャル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、送信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、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が生成されます。

選択された I²S 標準モードに依存する書き込み動作の詳細については、[セクション 28.4.2 : 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 SPI_DR 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。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した ([セクション 28.4.4 : I²S マスタモード](#)に示す手順を参照)。

データやチャネルの長さに関係なく、プロトコルデータは 16 ビットの packets によって受信されます。つまり、受信バッファが満杯になるたびに RXNE フラグがセットされ、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が発生します。データ長とチャネル長の設定にもよりますが、受信バッファへの 1 回または 2 回の受信によって、プロトコル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 ビットのクリアは、SPI_DR レジスタを読み出すことで行われます。

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 信号に敏感です。

選択された I²S 標準モードに依存する読み出し動作の詳細については、[セクション 28.4.2 : 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。SPI_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、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。

28.4.5 I²S スレーブモード

スレーブモードで、I²S を送信または受信モードに設定することができます。動作モードは主に、I²S マスタ設定での説明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. SPI_I2SCFGR レジスタの I2SMOD ビットをセットして I²S 機能にアクセスし、I2SSTD[1:0] ビットを通じて I²S 標準を選択し、DATLEN[1:0] ビットを通じてデータ長を選択し、CHLEN ビットを設定してフレームのチャンネルごとのビット数を選択します。SPI_I2SCFGR レジスタの I2SCFG[1:0] ビットを通じて、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、SPI_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. SPI_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と、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 データレジスタにロードしなければ、マスタは通信を開始しません。

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項目は、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、データは送信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 信号に敏感です。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 PCLK の 2 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パラレルロードされてから、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、送信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、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が生成されます。

なお、送信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 標準モードに依存する書き込み動作の詳細については、[セクション 28.4.2: 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 SPI_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 SPI_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。SPI_CR2 レジスタの ERRIE ビットがセットされた場合、SPI_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。ポイント 1 では、SPI_I2SCFGR レジスタの I2SCFG[1:0] ビットを使用してマスタ受信モードを設定する必要がありました ([セクション 28.4.5: I²S スレーブモード](#)) に示す手順を参照)。

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、受信バッファが満杯になるたびに SPI_SR レジスタの RXNE フラグがセットされ、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が発生します。データ長とチャネル長の設定にもよりますが、受信バッファへの 1 回または 2 回の受信によって、プロトコル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。

CHSIDE フラグは、SPI_DR 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、外部マスタ部品によって管理される外部 WS ラインに反応します。

RXNE ビットのクリアは、SPI_DR レジスタを読み出すことで行われます。

選択された I²S 標準モードに依存する読み出し動作の詳細については、[セクション 28.4.2: 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。SPI_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発生します。

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、プロトコルチャネルを介して 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。

28.4.6 ステータスフラグ

I²S バスの状態を完全に監視するため、アプリケーションには 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です)。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、下記の手順を厳しく順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 がマスタレシーバモードにある場合を除きます。

BSY フラグのクリア：

- 転送が完了したとき (ただし、マスタ送信モードの場合は通信は連続的であると考えられます)
- I²S が無効化された場合

通信が連続的なとき：

- マスタ送信モードでは、BSY フラグは転送時に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。

送信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、送信バッファはエンプティ（空）であり、次に送信するデータを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送信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。

受信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、受信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 SPI_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します)。

このフラグは、PCM 標準では意味を持ちません (ショートおよびロングフレームモード)。

SPI_SR の OVR または UDR フラグがセットされ、SPI_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、SPI_SR ステータスレジスタを読み出します。

28.4.7 エラーフラグ

I²S 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 SPI_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、SPI_I2SCFGR の I2SMOD ビットがセットされると使用できます。SPI_CR2 の ERRIE ビットがセットされた場合、割り込みが生成されることがあります。

UDR ビットは、SPI_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 SPI_DR 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。SPI_CR2 の ERRIE ビットがセットされた場合、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、トランスミッタ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。SPI_DR レジスタへの読み出し動作によって、以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、SPI_DR レジスタを読み出し、続けて SPI_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からセットすることができます。このフラグは、スレーブとして WS ラインの変化が想定され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するためには、次のステップに従います。

1. I2S を無効化します。
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 詰めモードあるいは PCM モードではローレベル)、再度有効化します。

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環境に起因する可能性があります。ERRIE ビットをセットして、エラー割り込みを発生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。

28.4.8 I²S 割り込み

表 160に I²S 割り込みのリストを示します。

表 160. 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	FRE

28.5 SPI レジスタと I²S 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、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。

28.5.1 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 モード	BIDI OE	CRC EN	CRC NEXT	DFF	RX ONLY	SSM	SSI	LSB FIRST	SPE	BR[2:0]			MSTR	CPOL	CPHA
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 **BIDIMODE** : 双方向データモードイネーブル

0 : 2 線単方向データモードを選択

1 : 1 線双方向データモードを選択

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 : 出力無効（受信専用モード）

1 : 出力有効（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。

I²S モードでは使用しません。

ビット 12 **CRCNEXT** : CRC 転送ネクスト

0 : データフェーズです (CRC フェーズはありません)。

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 ビットを書き込む必要があります。

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。

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。

I²S モードでは使用しません。

ビット 10 **RXONLY** : 受信専用

2 線単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。このビットはマルチスレーブシステムでも役立ちます。つまり、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 : 出力無効 (受信専用モード)

注 : このビットは I^2S 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 ピンの入力値は SSI ビットからの値によって置き換えられます。

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 I^2S 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 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 I^2S 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。
 I^2S 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 I^2S モードでは使用しません。

SPI を無効化する場合は、[セクション 28.3.8 : SPI の無効化に記載された手順に従います](#)。

ビット 5:3 **BR[2:0]** : ボーレート制御

000 : $f_{PCLK}/2$

001 : $f_{PCLK}/4$

010 : $f_{PCLK}/8$

011 : $f_{PCLK}/16$

100 : $f_{PCLK}/32$

101 : $f_{PCLK}/64$

110 : $f_{PCLK}/128$

111 : $f_{PCLK}/256$

注 : これらのビットは、通信中には変更しないでください。

これらは I^2S モードでは使用しません。

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。

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。

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。

I²S モードおよび **SPI TI** モードでは使用しません。

28.5.2 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
予約済み								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** : 送信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** : 受信バッファ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

エラー状態が発生したとき (SPI モードでは、CRCERR、OVR、MODF、FRE および **I²S** モードでは、UDR、OVR、FRE)、このビットは割り込みの発生を制御します。

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 (Cat.3、Cat.4、Cat.5 デバイスの場合)

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。

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 : マスタモードでセルが有効であるとき、SS 出力は有効です。セルはマルチマスタ環境では機能しません。

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 : 送信バッファ 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 : 送信バッファ DMA 無効
- 1 : 送信バッファ DMA 有効

ビット 0 RXDMAEN : 受信バッファ 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 : 受信バッファ DMA 無効
- 1 : 受信バッファ DMA 有効

28.5.3 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
予約済み							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 : フレームエラーなし

1 : フレームエラー発生

このビットは、ハードウェアによってセットされ、SPI_SR レジスタを読み出すことによってソフトウェアによってクリアされます。

選択するオーディオプロトコルによらず、このビットは SPI TI モードまたは I2S モードで使用されます。スリープモードにおいて予期しないタイミングで発生する NSS ラインまたは WS ラインの変化を検出し、外部マスタデバイスとスリープとの間の同期外れを通知します。

ビット 7 **BSY** : ビジーフラグ

0 : SPI (または I2S) 非ビジー状態

1 : SPI (または I2S) が通信ビジー状態であるか、または送信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 フラグの使用には注意が必要です (セクション 28.3.7 : ステータスフラグ およびセクション 28.3.8 : SPI の無効化 を参照)。**

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 28.4.7 \(811 ページ\)](#) 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。フラグクリアの操作手順は [セクション 28.3.10 \(794 ページ\)](#) 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 : アンダーランが発生しませんでした。

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。フラグクリアの操作手順は [セクション 28.4.7 \(811 ページ\)](#) 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 モードでは無意味です。

ビット 1 **TXE** : 送信バッファエンプティ

0 : 送信バッファは空ではありません。

1 : 送信バッファは空です。

ビット 0 **RXNE** : 受信バッファ非エンプティ

0 : 受信バッファは空です。

1 : 受信バッファは空ではありません。

28.5.4 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

受信されたデータ、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、送信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、受信バッファに保持されている値が返されます。

注 : 次の注は 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 0 に強制されます。

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。

28.5.5 SPI CRC 多項式レジスタ (SPI_CRCPR) (I²S モードでは不使用)

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]															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 15:0 **CRCPOLY[15:0]** : CRC 多項式レジスタ

このレジスタは、CRC 計算用の多項式を内蔵します。

CRC 多項式 (0007h) は、このレジスタのリセット値です。他の多項式は、必要に応じて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。

28.5.6 SPI RX CRC レジスタ (SPI_RXCRCR) (I²S モードでは不使用)

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ミング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)、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 標準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)、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 標準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 モードに使用されません。

28.5.7 SPI TX CRC レジスタ (SPI_TXCRCR) (I²S モードでは不使用)

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ミング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)、8 つの LSB ビットのみが考慮されます。CRC 計算は、任意の CRC8 標準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)、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 標準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 モードに使用されません。

28.5.8 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
予約済み				I2SMOD	I2SE	I2SCFG		PCMSYN C	予約済み	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 ペリフェラルを無効化

1 : I²S ペリフェラルを有効化

注 : このビットは SPI モードで不使用。

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、SPI または I²S が無効なときに設定してください。

このビットは SPI モードで不使用。

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 標準使用) の場合にのみ意味を持ちます。

このビットは SPI モードで不使用。

ビット 6 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 5:4 I2SSTD : I2S 標準選択

00 : I²S フィリップス標準

01 : MSB 詰め標準 (左詰め)

10 : LSB 詰め標準 (右詰め)

11 : PCM 標準

I²S 標準の詳細については、を参照してください。[セクション 28.4.2 \(797 ページ\)](#) SPI モードでは未使用。

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 モードで不使用。

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 : 許されません。

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。

このビットは SPI モードで不使用。

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 モードでは使用されません。

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。

28.5.9 SPI_I²S プリスケアラレジスタ (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
予約済み						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、SPI または 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されます。

このビットは SPI モードで不使用。

ビット 8 ODD : プリスケアラの奇除数

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 28.4.3 \(805 ページ\)](#) を参照してください。SPI モードでは未使用。

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。このビットは、I²S がマスタモードのときにのみ使用されます。

ビット 7:0 I2SDIV : I2S リニアプリスケアラ

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 28.4.3 \(805 ページ\)](#) を参照してください。SPI モードでは未使用。

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。このビットは、I²S がマスタモードのときにのみ使用されます。

28.5.10 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 161. 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	予約済み																BIDIMODE	BIDIOE	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	0			
0x04	SPI_CR2	予約済み																								TXEIE	RXNEIE	ERRIE	FRF	予約済み	SSOE	TXDMAEN	RXDMAEN				
	リセット値																									0	0	0	0		0	0	0	0	0	0	0
0x08	SPI_SR	予約済み																								FRE	BSY	OVR	MODF	CRCERR	UDR	CHSIDE	TXE	RXNE			
	リセット値																									0	0	0	0	0	0	0	0	1	0		
0x0C	SPI_DR	予約済み																DR[15:0]																			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x10	SPI_CRCPR	予約済み																CRCPOLY[15:0]																			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1				
0x14	SPI_RXCRCR	予約済み																RxCRC[15:0]																			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x18	SPI_TXCRCR	予約済み																TxCRC[15:0]																			
	リセット値																	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
0x1C	SPI_I2SCFGR	予約済み																			I2SMOD	I2SE	I2SCFG	PCMSYNC	予約済み				I2SSTD	CKPOL	DATLEN	CHLEN					
	リセット値																				0	0	0		0	0	0	0	0	0	0	0	0				
0x20	SPI_I2SPR	予約済み																							MCKOE	ODD	I2SDIV										
	リセット値																								0	0	0	0	0	0	0	0	0	0	0	0	1

レジスタ境界アドレスについては、表 5 (47 ページ) を参照してください。

29 セキュアデジタル入力/出力インタフェース (SDIO)

このセクションは、Cat.4 デバイスにのみ適用されます。SDIO を使用可能かどうかについては、デバイスデータシートを参照してください。

29.1 SDIO の主な機能

SD/SDIO MMC カードホストインタフェース (SDIO) は、APB2 ペリフェラルバスと MultiMediaCard (MMC)、SD メモリカード、SDIO カード、および CE-ATA デバイス間のインタフェースを提供します。

MMCA 技術委員会から発行された MultiMediaCard システム仕様書は、MultiMediaCard Association のウェブサイト <http://www.jedec.org/> から入手できます。

SD メモリカードと SD I/O カードシステム仕様書は、SD card Association のウェブサイト www.sdcard.org から入手できます。

CE-ATA システム仕様書は、CE-ATA ワークグループのウェブサイトから入手できます。

SDIO には以下の機能があります。

- MultiMediaCard System Specification Version 4.2 に完全準拠。3 つの異なるデータバスモードに対応したカードサポート：1ビット（デフォルト）、4ビット、8ビット
- 以前のバージョンの MultiMediaCards との完全互換（上位互換性）
- SD Memory Card Specifications Version 2.0 に完全準拠
- SD I/O Card Specification Version 2.0 に完全準拠：2 つの異なるデータバスモードに対応したカードサポート：1ビット（デフォルト）および 4ビット
- CE-ATA 機能の完全サポート（CE-ATA digital protocol Rev1.1 に完全準拠）
- 8ビットモードで最大 48 MHz のデータ転送
- 外部双方向ドライバを制御するデータおよびコマンド出力イネーブル信号。

注： SDIO は SPI 互換の通信モードを備えていません。

SD メモリカードプロトコルは、MultiMediaCard system specification V2.11 で定義されている MultiMediaCard プロトコルのスーパーセットです。SD メモリカードに必要ないくつかのコマンドは、SD I/O 専用カードやコンボカードの I/O 部分でサポートされません。消去コマンドなど、SD I/O デバイスで使用されないコマンドは、SDIO でサポートされません。また、いくつかのコマンドは、SD メモリカードと SD I/O カードで異なるため、SDIO でサポートされません。詳細については、SD I/O card Specification Version 1.0 を参照してください。CE-ATA は、既存の MMC アクセスプリミティブを利用するプロトコルを使用して、MMC 電気接続によりサポートされます。インタフェースの電気信号定義は、MMC 基準で定義されているとおりです。

MultiMediaCard/SD バスは、カードをコントローラに接続します。

現行バージョンの SDIO は、一度に 1 つの SD/SDIO/MMC 4.2 カードと、MMC4.1 以前のスタックだけをサポートします。

29.2 SDIO バストポロジ

バス上の通信は、コマンドとデータ転送に基づいて行われます。

MultiMediaCard/SD/SD I/O バス上の基本的なトランザクションは、コマンド/レスポンストランザクションです。これらの種類のバストランザクションでは、コマンドまたはレスポンス構造内で直接、情報を転送します。また、データトークンを持つ操作もあります。

SD/SDIO メモリカードに対するデータ転送は、データブロック単位で行われます。MMC に対するデータ転送は、データブロック単位またはストリームで行われます。CE-ATA デバイスに対するデータ転送は、データブロック単位で行われます。

図 276. SDIO "no response" と "no data" 操作

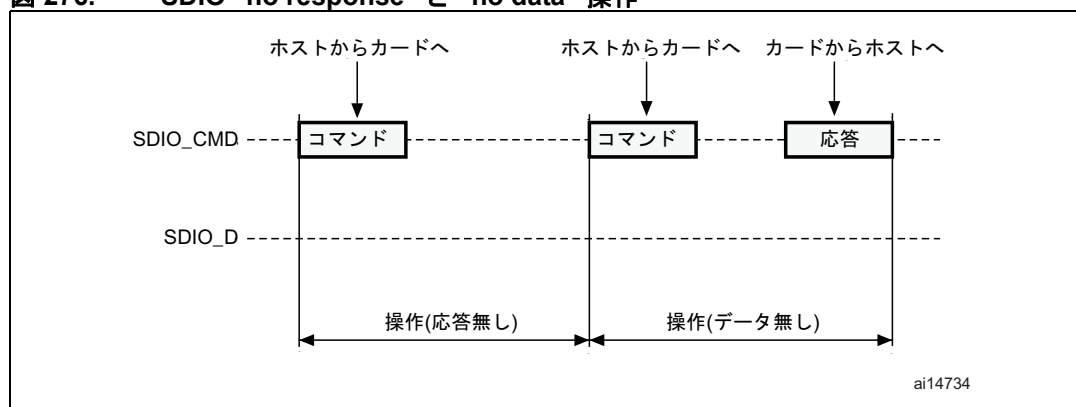


図 277. SDIO (複数) ブロック読み出し操作

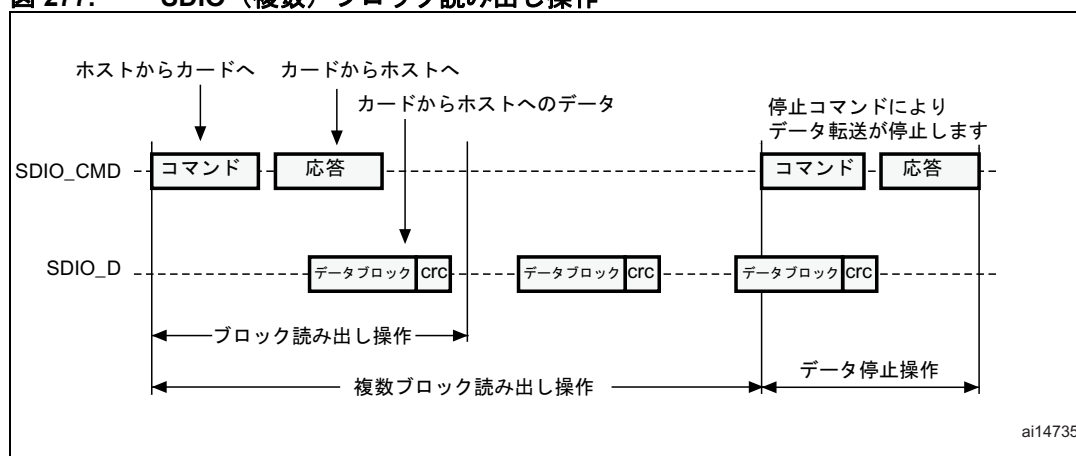
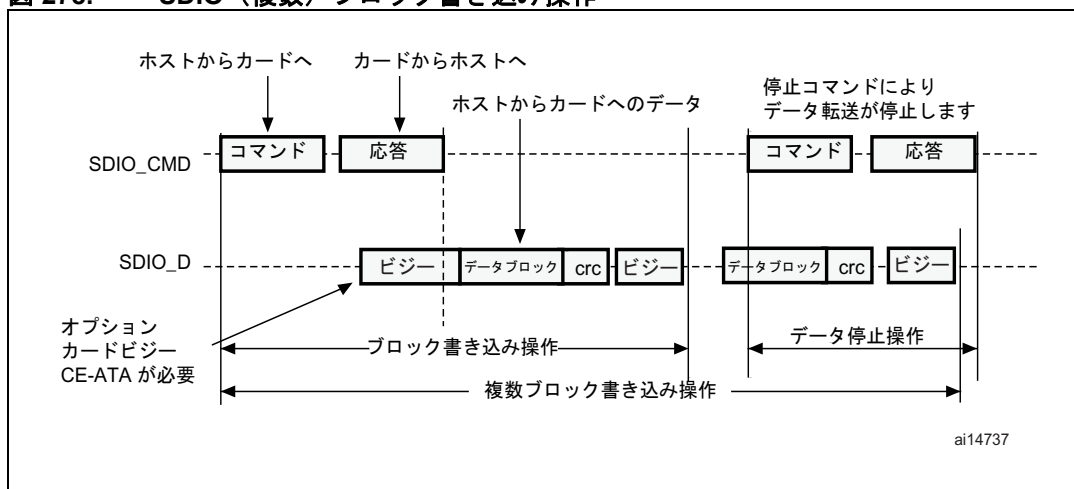


図 278. SDIO (複数) ブロック書き込み操作



注: SDIO は、ビジー信号がアサートされている限り (SDIO_D0 をローヘプル)、データを送信しません。

図 279. SDIO 順次読み出し操作

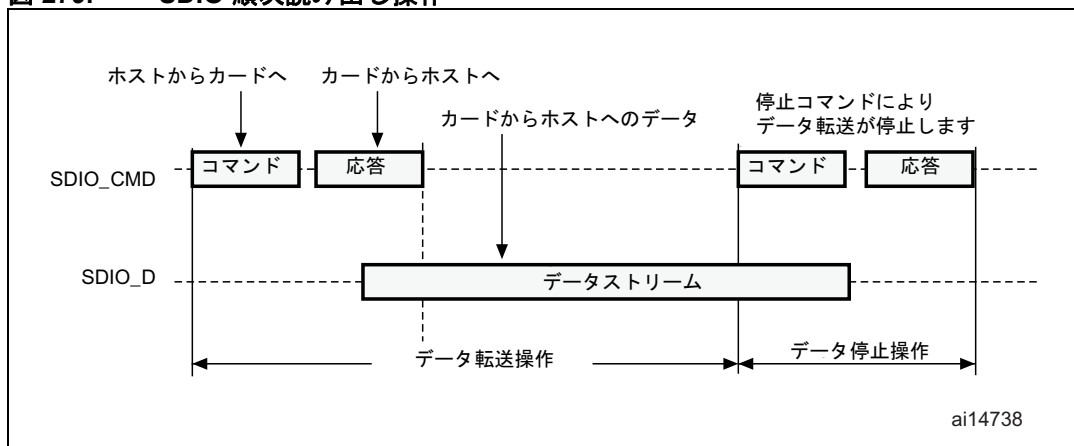
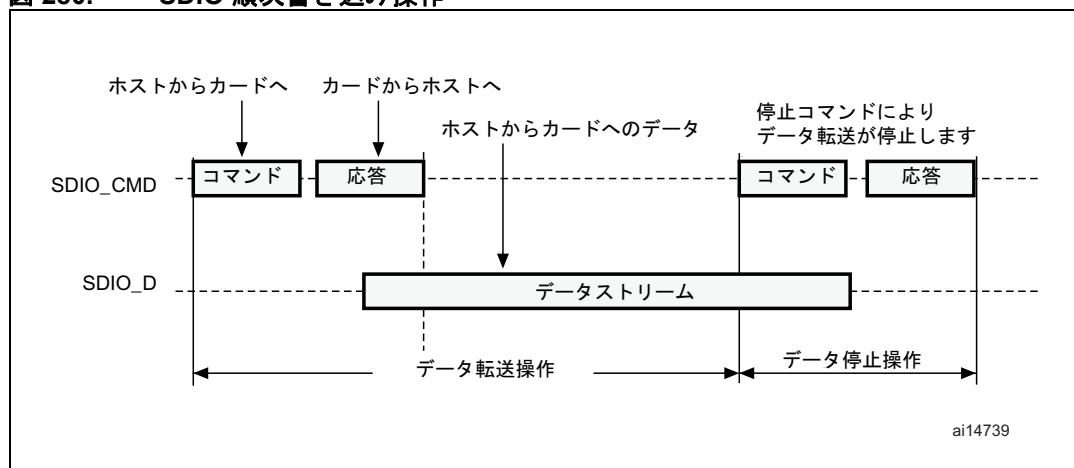


図 280. SDIO 順次書き込み操作

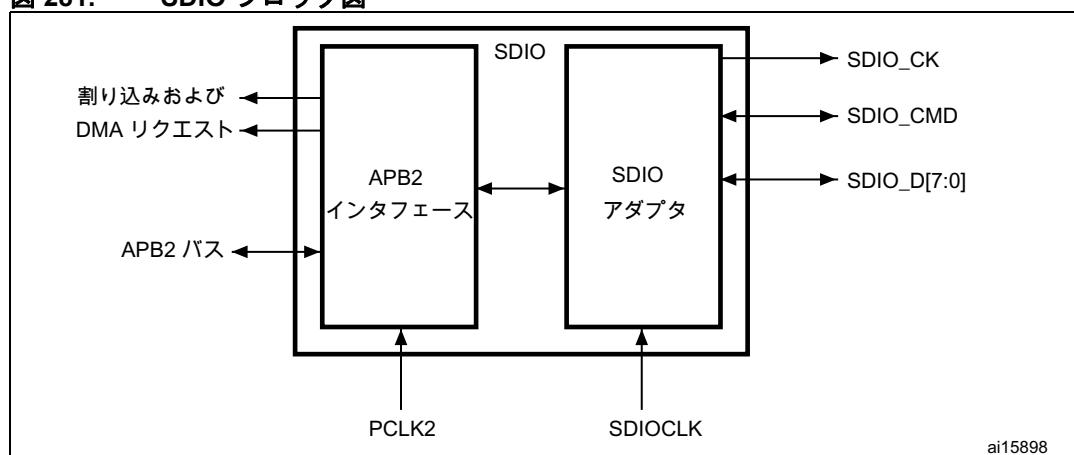


29.3 SDIO の機能詳細

SDIO は、2 つの部分から成ります。

- SDIO アダプタブロックは、クロック生成ユニット、コマンドおよびデータ転送など、MMC/SD/SD I/O カードに固有のすべての機能を提供します。
- APB2 インタフェースは、SDIO アダプタレジスタにアクセスして、割り込みおよび DMA リクエスト信号を生成します。

図 281. SDIO ブロック図



デフォルトでは、SDIO_D0 はデータ転送に使用されます。初期化後、ホストはデータバス幅を変更することができます。

MultiMediaCard がバスに接続されている場合、SDIO_D0、SDIO_D[3:0]、または SDIO_D[7:0] はデータ転送に使用できます。MMC V3.31 以前では、1ビットのデータのみがサポートされるので、使用できるのは SDIO_D0 だけです。

SD または SD I/O カードがバスに接続されている場合、ホストはデータ転送に SDIO_D0 または SDIO_D[3:0] を使用するように構成することができます。すべてのデータラインはプッシュプルモードで動作します。

SDIO_CMD には、2 つの動作モードがあります。

- 初期化の場合は、オープンドレイン（MMCV3.31 以前）
- コマンド転送の場合は、プッシュプル（SD/SD I/O カード MMC4.2 は、初期化もプッシュプルドライバを使用します）

SDIO_CK は、カードに供給されるクロックです。各クロックサイクルで、コマンドラインとデータラインの両方に 1 ビットが転送されます。クロック周波数は、MultiMediaCard V3.31 の場合は 0 MHz から 20 MHz、MultiMediaCard V4.0/4.2 の場合は 0 から 48 MHz、SD/SD I/O カードの場合は 0 から 25 MHz の間で変化します。

SDIO は 2 つのクロック信号を使用します。

- SDIO アダプタクロック（SDIOCLK= 48 MHz）
- APB2 バスクロック（PCLK2）

PCLK2 および SDIO_CK の周波数は、下記の条件に従う必要があります。

$$\text{Frequency(PCLK2)} \geq 3/8 \times \text{Frequency(SDIO\_CK)}$$

表 162 に示されている信号は、MultiMediaCard/SD/SD I/O カードバス上で使用されます。

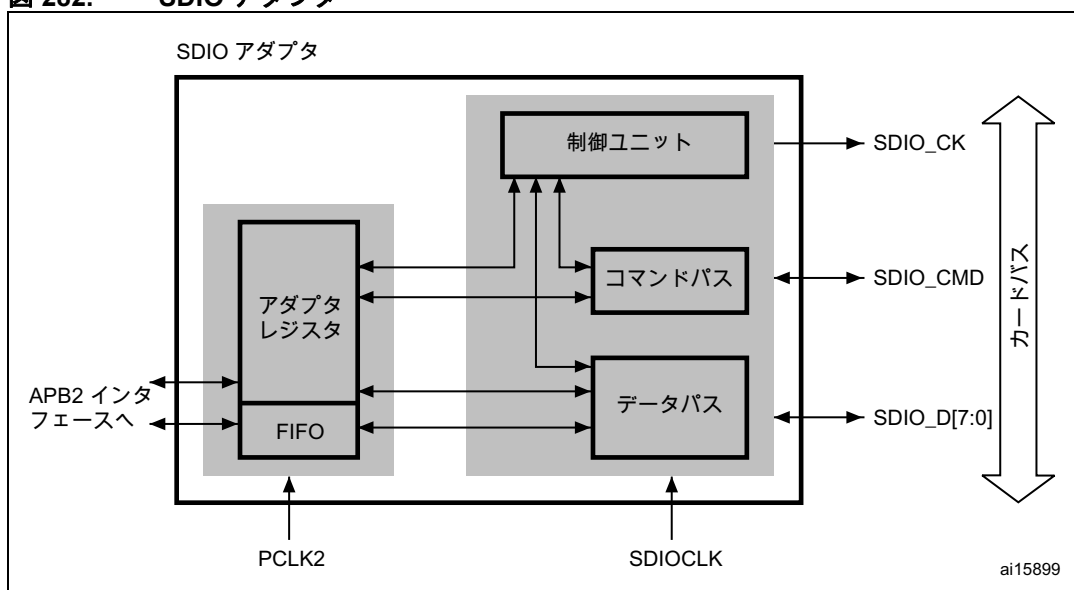
表 162. SDIO I/O 定義

ピン	方向	説明
SDIO_CK	出力	MultiMediaCard/SD/SDIO カードクロックこのピンは、ホストからカードへのクロックです。
SDIO_CMD	双方向	MultiMediaCard/SD/SDIO カードコマンドこのピンは、双方向のコマンド/レスポンス信号です。
SDIO_D[7:0]	双方向	MultiMediaCard/SD/SDIO カードデータ。これらのピンは、双方向データバスです。

29.3.1 SDIO アダプタ

図 282 は、SDIO アダプタの簡略化されたブロック図です。

図 282. SDIO アダプタ



SDIO アダプタは、マルチメディアカードスタックやセキュアデジタルメモリカードとのインタフェースを提供するマルチメディア/セキュアデジタルメモリカードバスマスタです。次の 5 つのサブユニットで構成されています。

- アダプタレジスタブロック
- 制御ユニット
- コマンドバス
- データバス
- データ FIFO

注： アダプタレジスタと FIFO は、APB2 バスクロックドメイン (PCLK2) を使用します。制御ユニット、コマンドバス、およびデータバスは、SDIO アダプタクロックドメイン (SDIOCLK) を使用します。

アダプタレジスタブロック

アダプタレジスタブロックは、すべてのシステムレジスタを含んでいます。このブロックは、マルチメディアカードのスタティックフラグをクリアする信号も生成します。クリア信号は、SDIO クリアレジスタの対応するビット位置に 1 が書き込まれると生成されます。

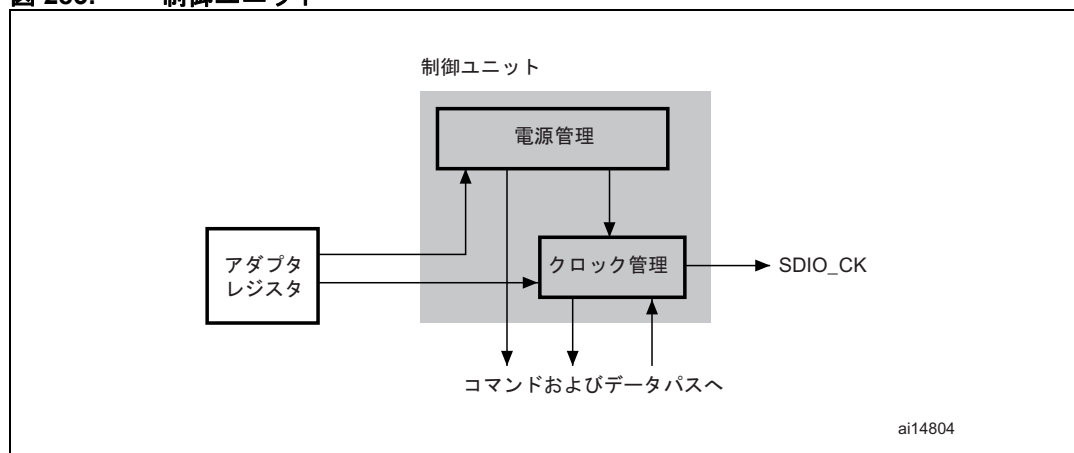
制御ユニット

制御ユニットは、電源管理機能とメモ리카ードクロックのクロック分周器を含みます。

3つの電源フェーズがあります。

- パワーオフ
- パワーアップ
- パワーオン

図 283. 制御ユニット



制御ユニットは、図 283 に示されています。電源管理サブユニットとクロック管理サブユニットで構成されています。

電源管理サブユニットは、パワーオフおよびパワーアップフェーズの間、カードバス出力信号を無効にします。

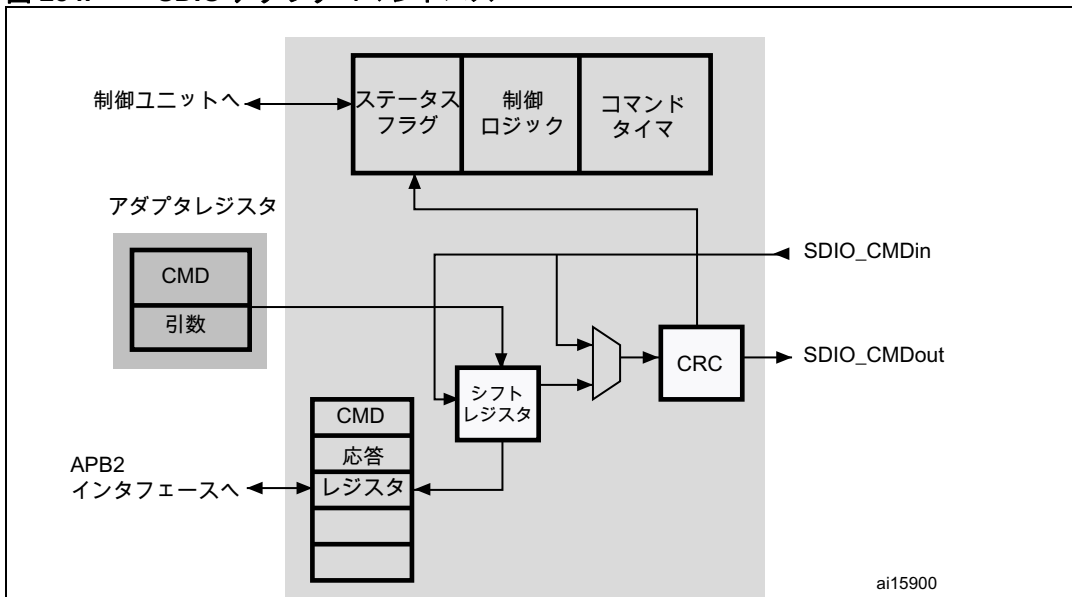
クロック管理サブユニットは、SDIO_OK 信号を生成し、制御します。SDIO_CK 出力は、クロック分周またはクロックバイパスモードを使用することができます。クロック出力は、次の場合にインアクティブです。

- リセット後
- パワーオフまたはパワーアップフェーズ中
- 節電モードが有効であり、カードバスがアイドル状態の場合（コマンドおよびデータバスサブユニットがアイドルフェーズに入ってから 8 クロック周期後）

コマンドパス

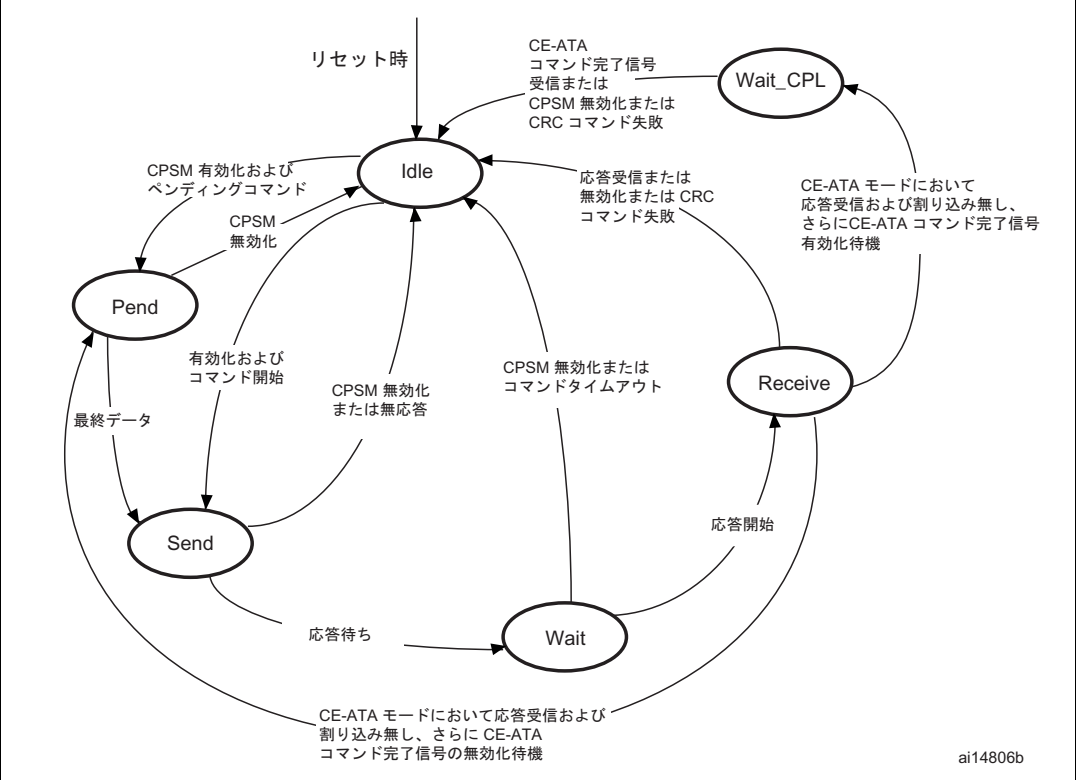
コマンドパスユニットは、カードにコマンドを送信し、カードからレスポンスを受信します。

図 284. SDIO アダプタコマンドパス



- コマンドパスステートマシン (CPSM)
 - － コマンドレジスタが書き込まれて、イネーブルビットがセットされると、コマンド転送が開始します。コマンドが送信されると、コマンドパスステートマシン (CPSM) は、ステータスフラグをセットし、レスポンスが不要な場合はアイドル状態になります。レスポンスが必要な場合は、レスポンスを待ちます (図 285 (832 ページ) を参照)。レスポンスを受信されると、受信された CRC コードと内部生成されたコードが比較され、適切なステータスフラグがセットされます。

図 285. コマンドパスステートマシン (CPSM)



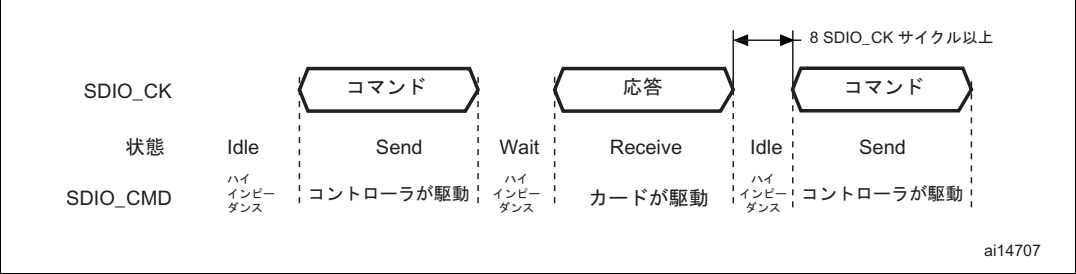
ウェイト状態に入ると、コマンドタイマが動き出します。CPSM が受信状態になる前にタイムアウトに達した場合には、タイムアウトフラグがセットされ、アイドル状態に入ります。

注： **コマンドタイムアウトは、64 SDIO_CK クロック周期の固定値です。**

コマンドレジスタの割り込みビットがセットされた場合、タイマは無効であり、CPSM はカードの 1 つからの割り込みリクエストを待ちます。コマンドレジスタのペンディングビットがセットされた場合、CPSM は保留状態に入り、データパスサブユニットからの CmdPend 信号を待ちます。CmdPend が検出されると、CPSM は送信状態に移行します。これによって、停止コマンドの送信をトリガするデータカウンタが有効になります。

注： **CPSM は、 N_{CC} および N_{RC} タイミング制約を満たすために、少なくとも 8 SDIO_CK 周期の間、アイドル状態を保ちます。 N_{CC} は、2 つのホストコマンド間の最小遅延であり、 N_{RC} は、ホストコマンドとカードレスポンスの間の最小遅延です。**

図 286. SDIO コマンド転送



- コマンドフォーマット

- コマンド: コマンドは、動作を開始するトークンです。コマンドは、ホストから 1 つのカード (アドレス付きコマンド) または接続されたすべてのカードに送信されます (ブロードキャストコマンドは MMCV3.31 以前で使用可能です)。コマンドは、CMD ライン上で連続的に転送されます。すべてのコマンドは、固定長 48 ビットです。MultiMediaCard、SD メモリカード、および SDIO カードのコマンドトークンの一般的なフォーマットを [表 163](#) に示します。CE-ATA コマンドは、MMC コマンド V4.2 の拡張であり、同じフォーマットです。コマンドパスは半二重モードで動作するので、コマンドとレスポンスは送受信のいずれかができます。CPSM が送信状態でない場合、SDIO_CMD は [図 286 \(832 ページ\)](#) に示されるようにハイインピーダンス状態になります。SDIO_CMD のデータは、SDIO_CK の立ち上がりエッジと同期します。[表 163](#) に、コマンドフォーマットを示します。

表 163. コマンドフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	1	送信ビット
[45:40]	6	-	コマンドインデックス
[39:8]	32	-	引数
[7:1]	7	-	CRC7
0	1	1	終了ビット

- レスポンス: レスポンスは、以前に受信されたコマンドへの応答として、アドレス指定されたカードから (または MMC V3.31 以前では、すべてのカードから同期して) 送信されるトークンです。レスポンスは、CMD ライン上で連続的に転送されます。

SDIO は、2 種類のレスポンスをサポートします。どちらも CRC エラーチェックを使用します。

- 48 ビットショートレスポンス
- 136 ビットロングレスポンス

注: レスポンスが CRC を含まない場合 (CMD1 レスポンス)、デバイスドライバは CRC 障害ステータスを無視する必要があります。

表 164. ショートレスポンスフォーマット

ビット位置	幅	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	-	コマンドインデックス
[39:8]	32	-	引数
[7:1]	7	-	CRC7 (または 1111111)
0	1	1	終了ビット

表 165. ロングレスポンスフォーマット

ビット位置	幅	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	111111	予約済み
[127:1]	127	-	CID または CSD (内部 CRC7 を含む)
0	1	1	終了ビット

コマンドレジスタは、コマンドインデックス（カードに送信される 6 ビット）とコマンドタイプを含みます。これらによって、コマンドがレスポンスを必要とするかどうかと、レスポンスが 48 ビット長か 136 ビット長かが決まります（[セクション 29.9.4 \(868 ページ\)](#) を参照）。コマンドパスは、[表 166](#) に示されているように、ステータスフラグを実装します。

表 166. コマンドパスステータスフラグ

フラグ	説明
CMDREND	レスポンス CRC が OK の場合にセットされます。
CCRCFAIL	レスポンス CRC が失敗した場合にセットされます。
CMDSENT	コマンド（レスポンスを必要としない）が送信されたときにセットされます。
CTIMEOUT	レスポンスタイムアウト。
CMDACT	コマンド転送中。

CRC 生成回路は、CRC コードの前のすべてのビットについて CRC チェックサムを計算します。これには、開始ビット、送信ビット、コマンドインデックス、およびコマンド引数（またはカードステータス）が含まれます。CRC チェックサムは、ロングレスポンスフォーマットの場合、CID または CSD の最初の 120 ビットについて計算されます。開始ビット、送信ビット、および 6 つの予約ビットは、CRC 計算に使用されないことに注意してください。

CRC チェックサムは 7 ビットの値です。

$$\text{CRC}[6:0] = \text{剰余} [(M(x) * x^7) / G(x)]$$

$$G(x) = x^7 + x^3 + 1$$

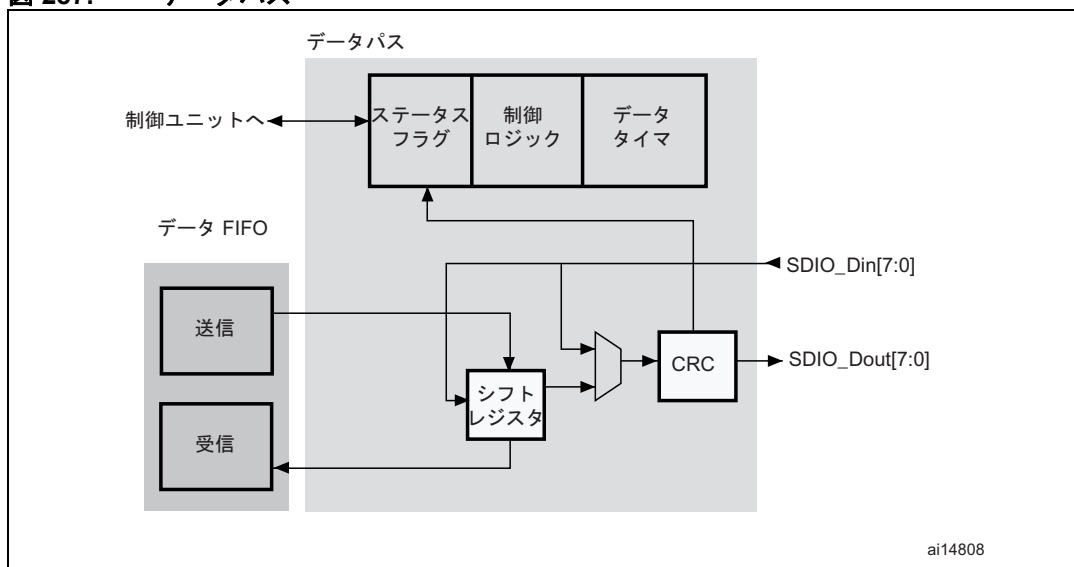
$$M(x) = (\text{start bit}) * x^{39} + \dots + (\text{CRC 前の最後のビット}) * x^0, \text{または}$$

$$M(x) = (\text{start bit}) * x^{119} + \dots + (\text{CRC 前の最後のビット}) * x^0$$

データパス

データパスサブユニットは、カードとの間でデータ転送を行います。[図 287](#) に、データパスのブロック図を示します。

図 287. データパス



カードデータバス幅は、クロック制御レジスタを使用してプログラムできます。4 ビット幅バスモードが有効な場合、データは 4 つすべてのデータ信号 (SDIO_D[3:0]) で、クロックサイクルごとに 4 ビットで転送されます。8 ビット幅バスモードが有効な場合、データは 8 つすべてのデータ信号 (SDIO_D[7:0]) で、クロックサイクルごとに 8 ビットで転送されます。ワイドバスモードが有効でない場合は、SDIO_D0 上でクロックサイクルごとに 1 ビットだけが転送されます。

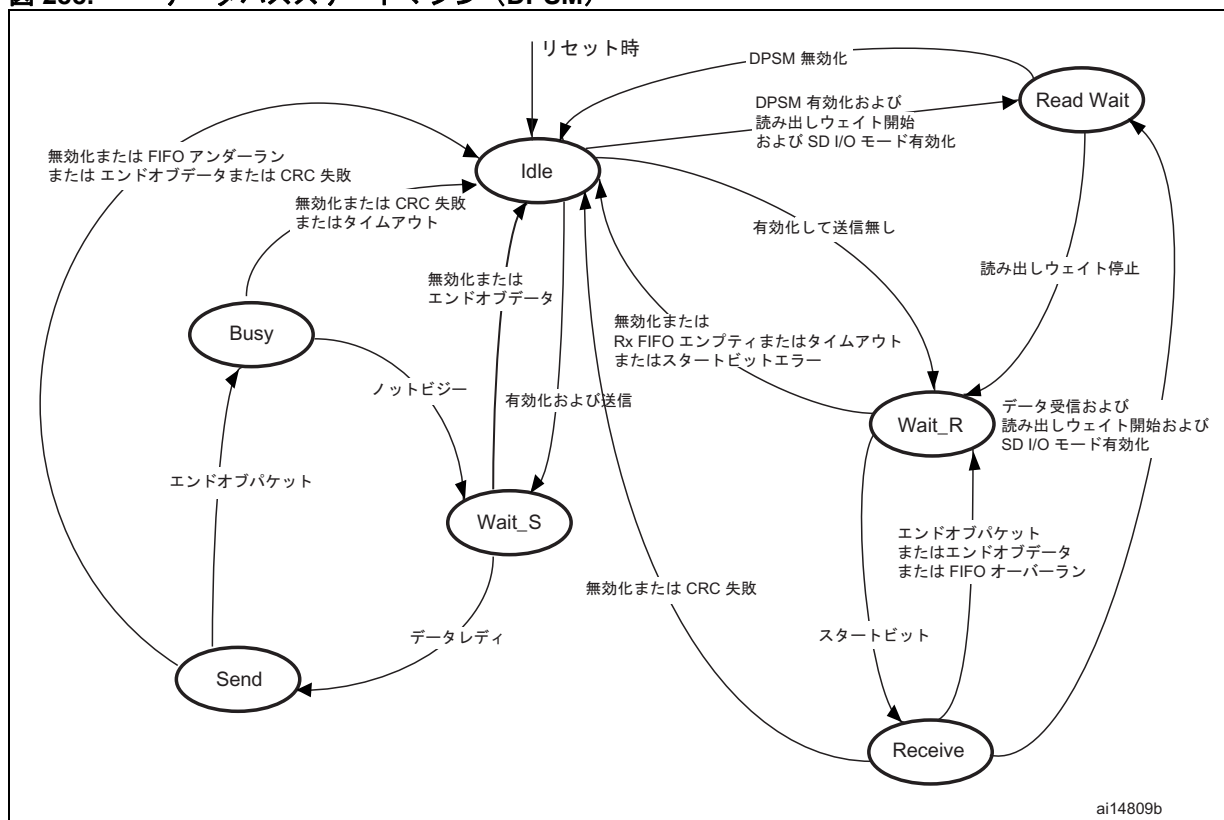
転送の方向 (送信または受信) によって、データパスステートマシン (DPSM) は、有効になったときに Wait_S または Wait_R 状態へ移行します。

- 送信 : DPSM は Wait_S 状態へ移行します。転送 FIFO にデータがある場合、DPSM は送信状態へ移行し、データパスサブユニットはカードへのデータ転送を開始します。
- 受信 : DPSM は Wait_R 状態へ移行して、開始ビットを待ちます。開始ビットを受信すると、DPSM は受信状態へ移行し、データパスサブユニットはカードからのデータ受信を開始します。

データパスステートマシン (DPSM)

DPSM は、SDIO_CK 周波数で動作します。カードバス信号上のデータは、SDIO_CK の立ち上がりエッジと同期しています。DPSM には、[図 288 : データパスステートマシン \(DPSM\)](#) に示されている 6 つの状態があります。

図 288. データパスステートマシン (DPSM)



- アイドル：データパスはインアクティブであり、SDIO_D[7:0] 出力はハイインピーダンスです。データ制御レジスタが書き込まれ、イネーブルビットがセットされると、DPSM はデータカウンタに新しい値をロードし、データ方向ビットに応じて Wait_S または Wait_R 状態へ移行します。
- Wait_R：データカウンタがゼロに等しい場合、DPSM は受信 FIFO が空になると、アイドル状態へ移行します。データカウンタがゼロでない場合、DPSM は SDIO_D 上の開始ビットを待ちます。タイムアウト前に開始ビットを受信した場合には、DPSM は受信状態に移行し、データブロックカウンタをロードします。開始ビットを検出する前にタイムアウトに達した場合、または開始ビットエラーが発生した場合には、アイドル状態へ移行して、タイムアウトステータスフラグをセットします。
- 受信：カードから受信されたシリアルデータは数バイトにまとめられ、データ FIFO に書き込まれます。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモードまたはストリームモードのいずれかです。
 - － ブロックモードでは、データブロックカウンタがゼロに達すると、DPSM は CRC コードを受信するまで待ちます。受信したコードが内部生成された CRC コードと一致する場合、DPSM は Wait_R 状態へ移行します。そうでない場合、CRC 失敗ステータスフラグをセットして、DPSM はアイドル状態へ移行します。
 - － ストリームモードでは、DPSM はデータカウンタがゼロでない間、データを受信します。カウンタがゼロになると、シフトレジスタ内の残りのデータがデータ FIFO に書き込まれ、DPSM は Wait_R 状態へ移行します。

FIFO オーバーランエラーが発生した場合、DPSM は FIFO エラーフラグをセットして、アイドル状態へ移行します。

- Wait_S : データカウンタがゼロの場合、DPSM はアイドル状態へ移行します。そうでない場合は、データ FIFO エンプティフラグがデアサートされるまで待つ、送信状態へ移行します。

注 : DPSM は、 N_{WR} タイミングの要件を満たすため、少なくとも 2 クロック周期の間、Wait_S 状態を保ちます。ここで、 N_{WR} は、カードレスポンスの受信とホストからのデータ転送の開始との間のクロックサイクル数です。

- 送信 : DPSM はカードへのデータの送信を開始します。データ制御レジスタの転送モードビットに応じて、データ転送はブロックモードまたはストリームモードのいずれかです。

- ブロックモードでは、データブロックカウンタがゼロに達すると、DPSM は内部生成された CRC コードと終了ビットを送信して、ビジー状態へ移行します。
- ストリームモードでは、DPSM は、イネーブルビットがハイであり、データカウンタがゼロでない間、カードへデータを送信します。その後、アイドル状態へ移行します。

FIFO アンダーランエラーが発生した場合、DPSM は FIFO エラーフラグをセットして、アイドル状態へ移行します。

- ビジー : DPSM は、CRC ステータスフラグを待ちます。
 - 良好な CRC ステータスを受信しなかった場合、DPSM はアイドル状態へ移行し、CRC 失敗ステータスフラグをセットします。
 - 良好な CRC ステータスを受信した場合、SDIO_D0 がローでない (カードがビジーでない) 場合は Wait_S 状態へ移行します。

DPSM がビジー状態の間にタイムアウトエラーが発生した場合には、DPSM はデータタイムアウトフラグをセットして、アイドル状態へ移行します。

データタイマは、DPSM が Wait_R またはビジー状態のときに有効になり、以下の場合にデータタイムアウトを生成します。

- データ送信中、DPSM がプログラムされたタイムアウト時間より長く、ビジー状態にとどまっていた場合。
- データ受信時、データの終わりが真でなく、DPSM がプログラムされたタイムアウト時間より長く Wait_R 状態にとどまっていた場合、タイムアウトが発生します。

- Data: データは、カードからホストへ、またはホストからカードへ転送されます。データは、データラインを介して転送されます。32ワードの FIFO に格納され、各ワードは 32ビット幅です。

表 167. データトークンフォーマット

説明	スタートビット	データ	CRC 16	終了ビット
ブロックデータ	0	-	あり	1
ストリームデータ	0	-	なし	1

データ FIFO

データ FIFO（ファーストインファーストアウト）サブユニットは、送受信ユニットを持つデータバッファです。

FIFO は、32ビット幅、32ワードの深さのデータバッファと、送受信ロジックを持っています。データ FIFO は APB2 クロックドメイン（PCLK2）で動作するので、SDIO クロックドメイン（SDIOCLK）のサブユニットからのすべての信号は、再同期されます。

TXACT および RXACT フラグに応じて、FIFO は、無効、送信有効、または受信有効にできます。TXACT と RXACT は、データパスサブユニットによって駆動され、相互排他的です。

- 送信 FIFO は、TXACT がアサートされると、送信ロジックとデータバッファを参照します。
 - 受信 FIFO は、RXACT がアサートされると、受信ロジックとデータバッファを参照します。
- 送信FIFO :
- データは、SDIO が送信有効なときに、APB2 インタフェースを通じて送信 FIFO に書き込むことができます。
- 送信 FIFO は、32 の連続アドレスを介してアクセス可能です。送信 FIFO は、読み出しポイントによって示されるデータワードを保持するデータ出力レジスタを含みます。データパスサブユニットにシフトレジスタがロードされると、読み出しポイントをインクリメントして、新しいデータを駆動します。
- 送信 FIFO が無効な場合、すべてのステータスフラグがデアサートされます。データパスサブユニットは、データを送信すると、TXACT をアサートします。

表 168. 送信 FIFO ステータスフラグ

フラグ	説明
TXFIFO	32 すべての送信 FIFO ワードが有効なデータを含んでいるとき、ハイにセットされます。
TXFIFOE	送信 FIFO が有効なデータを含まないとき、ハイにセットされます。
TXFIFOHE	8ワード以上の送信 FIFO ワードが空のとき、ハイにセットされます。このフラグは DMA リクエストとして使用できます。
TXDAVL	送信 FIFO が有効なデータを含むとき、ハイにセットされます。このフラグは、TXFIFOE フラグの反転です。
TXUNDERR	アンダーランエラーが発生すると、ハイにセットされます。このフラグは、SDIO クリアレジスタへの書き込みによってクリアされます。

- 受信 FIFO
- データパスサブユニットがワードデータを受信すると、書き込みデータバス上にデータを出力します。書き込み操作が完了すると、書き込みポイントがインクリメントされます。読み出し側では、読み出しポイントの現在値によって示される FIFO ワードの内容が、読み出しバス上に出力されます。受信 FIFO が無効な場合、すべてのステータスフラグがデアサートされ、読み出しおよび書き込みポイントがリセットされます。データパスサブユニットは、データを受信すると、RXACT をアサートします。表 169 に、受信 FIFO ステータスフラグを示します。受信 FIFO は、32 の連続アドレスを介してアクセス可能です。

表 169. 受信 FIFO ステータスフラグ

フラグ	説明
RXFIFO	32 すべての受信 FIFO ワードが有効なデータを含むとき、ハイにセットされます。
RXFIFOE	受信 FIFO が有効なデータを含まないとき、ハイにセットされます。
RXFIFOHF	8ワード以上の受信 FIFO ワードが空のとき、ハイにセットされます。このフラグは DMA リクエストとして使用できます。
RXDAVL	受信 FIFO が空でないとき、ハイにセットされます。このフラグは RXFIFOE フラグの反転です。
RXOVERR	オーバーランエラーが発生すると、ハイにセットされます。このフラグは、SDIO クリアレジスタへの書き込みによってクリアされます。

29.3.2 SDIO APB2 インタフェース

APB2 インタフェースは、割り込みおよび DMA リクエストを生成し、SDIO アダプタレジスタとデータ FIFO にアクセスします。データパス、レジスタデコーダ、および割り込み/DMA ロジックで構成されています。

SDIO 割り込み

割り込みロジックは、選択されたステータスフラグの少なくとも 1 つがハイのときにアサートされる割り込みリクエスト信号を生成します。マスクレジスタは、割り込みを生成する条件の選択を可能にします。ステータスフラグは、対応するマスクフラグがセットされている場合に、割り込みリクエストを生成します。

SDIO/DMA インタフェース : SDIO とメモリ間のデータ転送手順

次に示す例では、転送は SDIO ホストコントローラから MMC へ行われます (CMD24 (WRITE_BLOCK) を使用して 512 バイトを転送)。SDIO FIFO は、DMA コントローラを使用してメモリに格納されたデータで埋められます。

1. カード識別プロセスを行います。
2. SDIO_CK 周波数を上げます。
3. CMD7 を送信することによって、カードを選択します。
4. 次のように DMA2 を構成します。
 - a) DMA2 コントローラを有効にして、ペンディング中の割り込みをクリアします。
 - b) DMA2_Stream3 または DMA2_Stream6 Channel4 ソースアドレスレジスタをメモリ位置のベースアドレスでプログラムします。さらに、DMA2_Stream3 または DMA2_Stream6 Channel4 デスティネーションアドレスレジスタを SDIO FIFO レジスタアドレスでプログラムします。
 - c) DMA2_Stream3 または DMA2_Stream6 Channel4 制御レジスタをプログラムします (ペリフェラルでなくメモリインクリメント、ペリフェラルとソース幅はワードサイズ)。
 - d) DMA2_Stream3 または DMA2_Stream6 Channel4 をプログラムし、ペリフェラルをフローコントローラとして選択します (DMA_S3CR または DMA_S6CR 設定レジスタの PFCTRL ビットをセットします)。
 - e) DMA2_Stream3 または DMA2_Stream6 Channel4 において、インクリメンタルバースト転送を 4 ビートに設定します (少なくともペリフェラル側からの転送については)。
 - f) DMA2_Stream3 または DMA2_Stream6 Channel4 を有効化します。

5. 次のように、CMD24 (WRITE_BLOCK) を送信します。
 - a) SDIO データ長レジスタをプログラムします (SDIO データタイマレジスタは、カード識別プロセスの前にすでにプログラムされている必要があります)。
 - b) SDIO 引数レジスタに、データが転送されるカードのアドレス位置をプログラムします。
 - c) SDIO コマンドレジスタをプログラムします: CmdIndex を 24 (WRITE_BLOCK) に、WaitResp を“1”に (SDIO カードホストはレスポンスを待つ)、CPSMEN を“1”に (SDIO カードホストはコマンドの送信が可能)。他のフィールドはリセット値。
 - d) SDIO_STA[6] = CMDREND 割り込みを待ち、SDIO データ制御レジスタをプログラムします: DTEN を“1”に (SDIO カードホストはデータ送信が可能)、DTDIR を“0”に (コントローラからカードへ)、DTMODE を“0”に (ブロックデータ転送)、DMAEN を“1”に (DMA 有効)、DBLOCKSIZE を 0x9 に (512バイト)。他のフィールドは無視。
 - e) SDIO_STA[10] = DBCKEND を待ちます。
6. DMA イネーブルドチャネルステータスレジスタをポーリングして、チャネルがまだ有効にされていないことを確認します。

29.4 カード機能詳細

29.4.1 カード識別モード

カード識別モードでは、ホストはすべてのカードをリセットし、動作電圧範囲を確認し、カードを識別し、バス上の各カードの相対カードアドレス (RCA) をセットします。カード識別モードでは、すべての通信にコマンドライン (CMD) のみが使われます。

29.4.2 カードリセット

GO_IDLE_STATE コマンド (CMD0) は、ソフトウェアリセットコマンドであり、MultiMediaCard と SD メモリをアイドル状態にします。IO_RW_DIRECT コマンド (CMD52) は、SD I/O カードをリセットします。パワーアップまたは CMD0 の後、すべてのカード出力バスドライバはハイインピーダンス状態であり、カードはデフォルトの相対アドレス (RCA=0x0001) とデフォルトのドライバステージレジスタ設定 (最低速度、最高駆動電流) で初期化されます。

29.4.3 動作電圧範囲の検証

すべてのカードは、仕様の範囲内の任意の動作電圧で SDIO カードホストと通信できます。サポートされる最小および最大 V_{DD} 値は、カード上の動作条件レジスタ (OCR) で定義されます。

ペイロードメモリにカード識別番号 (CID) とカード固有データ (CSD) を格納するカードは、この情報をデータ転送 V_{DD} 条件下でのみ通信することができます。SDIO カードホストモジュールとカードの V_{DD} 範囲に互換性がないときには、カードは識別サイクルを完了できず、CSD データを送信できません。このため、SDIO カードホストに必要な V_{DD} 範囲に一致しないカードを識別して拒否するためのメカニズムとして、特殊なコマンド SEND_OP_COND (CMD1)、SD_APP_OP_COND (SD メモリ用 ACMD41)、および IO_SEND_OP_COND (SD I/O 用 CMD5) が用意されています。SDIO カードホストは、これらのコマンドのオペランドとして、必要な V_{DD} 電圧範囲を送信します。指定された範囲でデータ転送を行えないカードは、バスから切断され、インアクティブ状態になります。

電圧範囲をオペランドとして含めずに、これらのコマンドを使用することによって、SDIO カードホストは各カードへの問い合わせを行い、共通の電圧範囲を判断してから、範囲外のカードをインアクティブにすることができます。この問い合わせは、SDIO カードホストが共通の電圧範囲を選択できるときや、カードが使用不可能であることをユーザに通知する必要があるときに使用されます。

29.4.4 カード識別プロセス

カード識別プロセスは、MultiMediaCard と SD カードで異なります。MultiMediaCard では、識別プロセスは F_{od} クロックレートで開始します。SDIO_CMD ライン出力ドライバは、オープンドレインであり、このプロセス中の並行カード操作が可能です。登録プロセスは、次のように行われます。

1. バスがアクティベートされます。
2. SDIO カードホストは、SEND_OP_COND (CMD1) をブロードキャストして、動作条件を受信します。
3. レスポンスは、すべてのカードからの動作条件レジスタのワイヤード AND 操作です。
4. 互換性のないカードはインアクティブ状態へ移行されます。
5. SDIOカードホストは、すべてのアクティブカードに ALL_SEND_CID (CMD2) をブロードキャストします。
6. アクティブカードは、それぞれの CID 番号をシリアルに同時送信します。発信 CID ビットがコマンドラインのビットと一致しないカードは、転送を停止して、次の識別サイクルを待たなければなりません。1 つのカードが CID 全体を正常に SDIO カードホストに送信すると、識別状態へ移行します。
7. SDIO カードホストは、そのカードに SET_RELATIVE_ADDR (CMD3) を発行します。この新しいアドレスは、相対カードアドレス (RCA) と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードはスタンバイ状態へ変化し、その後の識別サイクルに反応せず、出力がオープンドレインからプッシュプルへ切り替わります。
8. SDIO カードホストは、タイムアウト条件を受信するまで、ステップ 5 ~ 7 を繰り返します。

SD カードの場合、識別プロセスは F_{od} クロックレートで開始し SDIO_CMD ライン出力ドライバはオープンドレインでなくプッシュプルドライバです。登録プロセスは、次のように行われます。

1. バスがアクティベートされます。
2. SDIO カードホストは SD_APP_OP_COND (ACMD41) をブロードキャストします。
3. カードは、それぞれの動作条件レジスタの内容で応答します。
4. 互換性のないカードは、インアクティブ状態へ移行されます。
5. SDIOカードホストは、すべてのアクティブカードに ALL_SEND_CID (CMD2) をブロードキャストします。
6. カードは、一意なカード識別番号 (CID) を送り返して、識別状態に入ります。
7. SDIO カードホストは、アクティブカードに SET_RELATIVE_ADDR (CMD3) をアドレスを指定して発行します。この新しいアドレスは、相対カードアドレス (RCA) と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードは、スタンバイ状態へ変化します。SDIO カードホストは、このコマンドを再発行して、RCA を変更することができます。カードの RCA は、最後に割り当てられた値です。
8. SDIO カードホストは、すべてのアクティブカードについて、ステップ 5 ~ 7 を繰り返します。

SD I/O カードの場合、登録プロセス次のように行われます。

1. バスがアクティベートされます。
2. SDIO カードホストは IO_SEND_OP_COND (CMD5) を送信します。
3. カードは、それぞれの動作条件レジスタの内容で応答します。
4. 互換性のないカードは、インアクティブ状態に設定されます。
5. SDIO カードホストは、アクティブカードに SET_RELATIVE_ADDR (CMD3) をアドレスを指定して発行します。この新しいアドレスは、相対カードアドレス (RCA) と呼ばれ、CID より短く、カードをアドレス指定します。これを割り当てられたカードは、スタンバイ状態へ変化します。SDIO カードホストは、このコマンドを再発行して、RCA を変更することができます。カードの RCA は、最後に割り当てられた値です。

29.4.5 ブロック書き込み

ブロック書き込み (CMD24-27) では、1 つ以上のブロックデータがホストからカードに転送され、ホストによって各ブロックの終わりに CRC が付加されます。ブロック書き込みをサポートするカードは、WRITE_BLK_LEN によって定義されたデータブロックを常に受け入れることができます。CRC が失敗した場合、カードは SDIO_D ライン上で失敗をしめし、転送されたデータは破棄されて書き込まれず、それ以降に転送されたブロック (複数ブロック書き込みモード時) はすべて無視されます。

ホストが、累積長がブロック長に合わない部分的なブロックを使用し、ブロックのミスアラインメントが許されない場合 (CSD パラメータ WRITE_BLK_MISALIGN がセットされていない)、カードはミスアラインされた最初のブロックでブロックミスアラインメントエラーを検出します。(ステータスレジスタの ADDRESS_ERROR ビットがセットされます)。ホストが書き込み保護されたエリアへの書き込みを試みた場合も、書き込み操作はアボートされます。ただし、この場合、カードは WP_VIOLATION ビットをセットします。

CID および CSD レジスタをプログラムする場合、以前のブロック長設定は不要です。転送データは、CRC 保護されます。CSD または CID レジスタの一部が ROM に格納される場合、この変更不能部分は、受信バッファの対応する部分と一致しなければなりません。これが一致しない場合、カードはエラーを報告し、レジスタの内容を変更しません。一部のカードでは、ブロックデータの書き込みに時間がかかり、その時間が予測不能ことがあります。ブロックデータの受信と CRC チェックの完了後、カードは書き込みを開始し、書き込みバッファが満杯で新しい WRITE_BLOCK コマンドから新しいデータを受け入れられない場合には、SDIO_D ラインをローに保ちます。ホストは、いつでも、SEND_STATUS コマンド (CMD13) でカードのステータスをポーリングでき、カードはそれぞれのステータスで応答します。READY_FOR_DATA ステータスビットは、カードが新しいデータを受け入れられるかどうか、または書き込み動作がまだ進行中かどうかを示します。ホストは、カードを切断状態にする CMD7 を発行することによってカードを選択解除でき (別のカードを選択するため)、書き込み操作を中断せずに SDIO_D ラインを解放できます。カードを再び選択したとき、プログラミングがまだ続行中であり、書き込みバッファを使用できない場合には、SDIO_D をローにプルすることによって、ビジー表示を再び有効にします。

29.4.6 ブロック読み出し

ブロック読み出しモードでは、データ転送の基本単位はブロックであり、その最大サイズは CSD (READ_BLK_LEN) で定義されます。READ_BLK_PARTIAL がセットされた場合、開始アドレスと終了アドレスが 1 つの物理ブロック (READ_BLK_LEN で定義) に完全に含まれる小さなブロックを送信することもできます。データの整合性を確保するために、各ブロックの終わりに CRC が付加されます。CMD17 (READ_SINGLE_BLOCK) は、ブロック読み出しを開始し、転送完了後、カードは転送状態に戻ります。

CMD18 (READ_MULTIPLE_BLOCK) は、複数の連続したブロックの転送を開始します。

ホストは、複数ブロック操作内のいつでも、タイプに関係なく、読み出しをアボートできます。トランザクションのアボートは、送信停止コマンドを送ることによって行われます。

カードが複数ブロック読み出し操作中にエラー (範囲外、アドレスのミスアラインメント、内部エラーなど) を検出した場合は、データ転送を停止して、データ状態に留まります。その場合、ホストは、送信停止コマンドを送ることによって操作をアボートする必要があります。読み出しエラーは、送信停止コマンドに対するレスポンスで報告されます。

カードが、事前定義されたブロック数の複数ブロック操作の最後のブロックを送信した後で、ホストが送信停止コマンドを送信した場合、カードはすでにデータ状態ではないので、無効なコマンドとして応答されます。ホストが、累積長さがブロックのアラインメントでない部分的なブロックを使用し、ブロックのミスアラインメントが許されない場合、カードは最初のミスアラインブロックの始めてブロックミスアラインメントエラー条件を検出します (ステータスレジスタの ADDRESS_ERROR ビットがセットされます)。

29.4.7 ストリームアクセス、ストリーム書き込み、およびストリーム読み出し (MultiMediaCard のみ)

ストリームモードでは、データはバイト単位で転送され、各ブロックの終わりに CRC は付加されません。

ストリーム書き込み (MultiMediaCard のみ)

WRITE_DAT_UNTIL_STOP (CMD20) は、SDIO カードホストからカードへのデータ転送を、指定されたアドレスから開始し、SDIO カードホストが停止コマンドを発行するまで続けます。部分的なブロックが許される場合 (CSD パラメータ WRITE_BL_PARTIAL がセットされている)、データストリームはカードアドレス空間内の任意のアドレスで開始と停止ができます。そうでない場合は、ブロック境界でのみ開始と停止ができます。転送データ量を事前に決めることができないので、CRC は使用できません。データ転送中にメモリ範囲の終わりに達し、SD カードホストから停止コマンドが送信されなかった場合、その後に転送されたデータは破棄されます。

ストリーム書き込み操作の最大クロック周波数は、カード固有データレジスタの次の式フィールドで与えられます。

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{writeblen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 書き込み最大周波数
- TRANSPEED = データ転送最大速度
- writeblen = 書き込みデータブロック最大長
- NSAC = データ読み出しアクセス時間 2 (CLK サイクル数)
- TAAC = データ読み出しアクセス時間 1
- R2WFACTOR = 書き込み速度係数

ホストがより高い周波数の使用を試みた場合、カードはデータを処理できずにプログラミングを停止し、ステータスレジスタの OVERRUN エラービットをセットします。それ以降のデータ転送を無視して、停止コマンドを待ちます (データ受信状態で)。ホストが書き込み保護領域への書き込みを試みた場合も、書き込み操作はアボートされます。ただし、この場合、カードは WP_VIOLATION ビットをセットします。

ストリーム読み出し (MultiMediaCard のみ)

READ_DAT_UNTIL_STOP (CMD11) は、ストリーム指向のデータ転送を制御します。

このコマンドは、SDIO カードホストが STOP_TRANSMISSION (CMD12) を送るまで、指定されたアドレスからデータを送信するようにカードに指示します。停止コマンドには、シリアルコマンド送信による実行の遅延があり、データ転送は停止コマンドの終了ビット後に停止します。データの送信中にメモリ範囲の割に達し、SDIO カードホストによって停止コマンドが送信されなかったときには、その後に送信されたデータは未定義とみなされます。

ストリーム読み出し操作の最大クロック周波数は、カード固有データレジスタの次の式および使用フィールドによって与えられます。

$$\text{Maximumspeed} = \text{MIN}(\text{TRANSPEED}, \frac{(8 \times 2^{\text{readblen}})(-\text{NSAC})}{\text{TAAC} \times \text{R2WFACTOR}})$$

- Maximumspeed = 読み出し最大周波数
- TRANSPEED = データ転送最大速度
- readblen = 読み出しデータブロック最大長
- writeblen = 書き込みデータブロック最大長
- NSAC = データ読み出しアクセス時間 2 (CLK サイクル数)
- TAAC = データ読み出しアクセス時間 1
- R2WFACTOR = 書き込み速度係数

ホストがより高い周波数の使用を試みた場合、カードはデータ転送を維持できません。これが発生した場合、カードはステータスレジスタの UNDERRUN エラービットをセットし、送信をアボートし、データ状態で停止コマンドを待ちます。

29.4.8 消去：グループ消去とセクタ消去

MultiMediaCard の消去可能単位は、消去グループです。消去グループは、カードの基本的な書き込み単位である書き込みブロックで測定されます。消去グループのサイズは、カード固有のパラメータであり、CSD で定義されます。

ホストは、連続した範囲の消去グループを消去できます。消去プロセスの開始は、3 ステップのシーケンスです。

まず、ホストは、ERASE_GROUP_START (CMD35) コマンドを使用して、範囲の開始アドレスを定義し、次に、ERASE_GROUP_END (CMD36) コマンドを使用して、範囲の最後のアドレスを定義し、最後に、ERASE (CMD38) コマンドを発行して、消去プロセスを開始します。消去コマンドのアドレスフィールドは、バイト単位の消去グループアドレスです。カードは、消去グループサイズ未満のすべての LSB を無視して、効果的にアドレスを消去グループの境界に合わせます。

消去コマンドが誤った順序で受信された場合、カードはステータスレジスタの ERASE_SEQ_ERROR ビットをセットして、シーケンス全体をリセットします。

シーケンス外 (SEND_STATUS を除き、どちらの消去コマンドでもない) のコマンドを受信した場合、カードはステータスレジスタの ERASE_RESET ビットをセットして、消去シーケンスをリセットし、最後のコマンドを実行します。

消去範囲に書き込み保護ブロックが含まれる場合、それらのブロックは消去されず、保護されていないブロックだけが消去されます。ステータスレジスタの WP_ERASE_SKIP ステータスビットがセットされます。

カードは、SDIO_D をローに保持することによって、消去が進行中であることを示します。実際の消去時間はかなり長く、ホストは CMD7 を発行して、カードを選択解除できます。

29.4.9 ワイドバス選択または選択解除

ワイドバス（4 ビットバス幅）操作は、SET_BUS_WIDTH (ACMD6) を使用して選択または選択解除されます。パワーアップ後または GO_IDLE_STATE (CMD0) 後のデフォルトのバス幅は、1 ビットです。SET_BUS_WIDTH (ACMD6) は、転送状態でのみ有効です。すなわち、バス幅を変更できるのは、カードが SELECT/DESELECT_CARD (CMD7) によって選択された後だけです。

29.4.10 保護管理

SDIO カードホストモジュールでは、カードのための 3 つの書き込み保護方法がサポートされています。

1. 内部カード書き込み保護（カード側の責任）
2. 機械的書き込み保護スイッチ（SDIO カードホストモジュール側のみの責任）
3. パスワード保護カードロック操作

内部カード書き込み保護

カードデータは、書き込みや消去から保護することができます。CSD の永久的または一時的書き込み保護ビットをセットすることによって、製造業者やコンテンツプロバイダからカード全体を永久的に保護することができます。セクタのグループの書き込み保護をサポートするカードの場合、CSD の WP_GRP_ENABLE ビットをセットすることで、データの一部を保護することができます。書き込み保護はアプリケーションによって変更できます。書き込み保護は、CSD で指定された WP_GRP_SIZE セクタ単位です。SET_WRITE_PROT および CLR_WRITE_PROT コマンドは、アドレス指定されたグループの保護を制御します。SEND_WRITE_PROT コマンドは、1 ブロックの読み出しコマンドと似ています。カードは、32 の書き込み保護ビット（指定されたアドレスから始まる 32 の書き込み保護グループを表す）に続いて 16 の CRC ビットを含むデータブロックを送信します。書き込み保護コマンドのアドレスフィールドは、バイト単位のグループアドレスです。

カードは、グループサイズ未満のすべての LSB を無視します。

機械的書き込み保護スイッチ

カードの横にある機械的なスライドタブによって、カードの書き込み保護をセット/クリアできます。スライドタブが開位置にあるときには、カードは書き込み保護され、閉位置にあるときには、カードの内容を変更できます。ソケット側の対応するスイッチは、カードが書き込み保護されていることを SDIO カードホストモジュールに示します。SDIO カードホストモジュールは、カードを保護する責任があります。書き込み保護スイッチの位置は、カードの内部回路にはわかりません。

パスワード保護

パスワード保護機能によって、SDIO カードホストモジュールはカードをパスワードでロック/アンロックできます。パスワードは 128 ビットの PWD レジスタに格納され、そのサイズは 8 ビットの PWD_LEN レジスタで設定されます。これらのレジスタは不揮発性なので、パワーサイクルでは消去されません。ロックされたカードは、特定のコマンドにตอบสนองし、実行します。すなわち、SDIO カードホストモジュールは、リセット、初期化、選択、およびステータスの問い合わせを行うことができますが、カード上のデータにアクセスすることはできません。パスワードが設定されると（PWD_LEN の非ゼロの値によって示される）、カードはパワーアップ後に自動的にロックされます。CSD および CID レジスタの書き込みコマンドと同様、ロック/アンロックコマンドは転送状態でのみ使用できます。この状態では、コマンドはアドレス引数を含まず、カードを使用する前にカードを選択しなければなりません。カードロック/アンロックコマンドは、通常の 1 ブロック書き込みコマンドの構造とバストランザクションタイプを持ちます。転送されるデータブロックは、コマンドに必要なすべての情報（パスワード設定モード、PWD そのもの、およびカードロック/アンロック）を含みます。コマンドデータブロックサイズは、カードロック/アンロックコマンドを送信する前に、SDIO カードホストモジュールによって定義され、表 183 に示されている構造を持ちます。



ビット設定は、次のとおりです。

- ERASE : セットすると、強制的に消去操作が行われます。他のすべてのビットはゼロでなければならない、コマンドバイトだけが送信されます。
- LOCK_UNLOCK : セットすると、カードをロックします。LOCK_UNLOCK は、SET_PWD と同時にセットできますが、CLR_PWD とは同時にセットできません。
- CLR_PWD : セットすると、パスワードデータをクリアします。
- SET_PWD : セットすると、パスワードデータをメモリに保存します。
- PWD_LEN : パスワードの長さ (バイト数) を定義します。
- PWD : パスワード (コマンドに応じて、新しいパスワードまたは現在使用中のパスワード)

以下のセクションでは、パスワードのセット/リセット、カードのロック/アンロック、および強制消去のコマンドシーケンスを説明します。

パスワードの設定

1. まだ選択されていない場合は、カードを選択します (SELECT/DESELECT_CARD、CMD7)。
2. 送信するブロック長 (SET_BLOCKLEN、CMD16) を定義します。8 ビットのカードロック/アンロックモード、8 ビットの PWD_LEN、および新しいパスワードのバイト数によって定義されます。パスワードの書き換えが行われるときには、古いパスワードと新しいパスワードの両方がコマンドとともに送信されることを考慮したブロックサイズでなければなりません。
3. 16ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともに LOCK/UNLOCK (CMD42) を送信します。データブロックは、モード (SET_PWD= 1)、長さ (PWD_LEN)、およびパスワード (PWD) そのものを示します。パスワードの書き換えが完了すると、長さ値 (PWD_LEN) には、古いパスワードと新しいパスワードの両方が含まれ、PWD フィールドには、古いパスワード (現在使用中) の後に新しいパスワードが続きます。
4. パスワードが一致すると、新しいパスワードとそのサイズが、それぞれ PWD と PWD_LEN に保存されます。送信された古いパスワードが予期されたパスワードと一致しない (サイズまたは内容) ときには、カードステータスレジスタの LOCK_UNLOCK_FAILED ビットがセットされ、パスワードは変更されません。

パスワード長フィールド (PWD_LEN) は、パスワードが現在設定されているかどうかを示します。このフィールドがゼロ以外のときには、パスワードが設定されていて、カードはパワーアップ後に自身をロックします。LOCK_UNLOCK ビットをセットするか (パスワードの設定時に)、カードをロックするための追加のコマンドを送信することによって、現在のパワーセッションでただちにカードをロックすることが可能です。

パスワードのリセット

1. まだ選択されていない場合は、カードを選択します (SELECT/DESELECT_CARD、CMD7)。
2. 送信するブロック長 (SET_BLOCKLEN、CMD16) を定義します。8ビットのカードロック/アンロックモード、8ビットの PWD_LEN、および現在使用中のパスワードのバイト数によって定義されます。
3. 16ビット のCRC を含めた、データライン上の適切なデータブロックサイズとともに LOCK/UNLOCK (CMD42) を送信します。データブロックは、モード (CLR_PWD=1)、長さ (PWD_LEN)、およびパスワード (PWD) そのものを示します。LOCK_UNLOCK ビットは無視されます。
4. パスワードが一致すると、PWD フィールドはクリアされ、PWD_LEN は 0 にセットされます。送信されたパスワードが予期されたパスワードと一致しないときには (サイズまたは内容)、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、パスワードは変更されません。

カードのロック

1. まだ選択されていない場合は、カードを選択します (SELECT/DESELECT_CARD、CMD7)。
2. 送信するブロック長 (SET_BLOCKLEN、CMD16) を定義します。8 ビットのカードロック / アンロックモード (表 183 のバイト 0)、8 ビットの PWD_LEN、および現在のパスワードのバイト数で定義されます。
3. 16 ビットの CRC を含めた、データライン上の適切なデータブロックサイズとともに LOCK/UNLOCK (CMD42) を送信します。データブロックは、モード (LOCK_UNLOCK = 1)、長さ (PWD_LEN)、およびパスワード (PWD) そのものを示します。
4. パスワードが一致すると、カードはロックされ、カードステータスレジスタの CARD_IS_LOCKED ビットがセットされます。送信されたパスワードが予期されたパスワードと一致しないときには (サイズまたは内容)、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、ロックは失敗します。

パスワードの設定とカードのロックを同じシーケンスで行うことが可能です。この場合、SDIO カードホストモジュールは、パスワードの設定に必要なすべてのステップを実行します ([パスワードの設定 \(846 ページ\)](#) を参照)。ただし、新しいパスワードコマンドを送信するときには、ステップ 3 で LOCK_UNLOCK ビットをセットする必要があります。

パスワードが以前に設定されているときには (PWD_LEN が 0 でない)、カードはパワーオンリセット後に自動的にロックされます。ロックされたカードをロックしようとしたり、パスワードが設定されていないカードをロックしようすると、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

カードのアンロック

1. まだ選択されていない場合は、カードを選択します (SELECT/DESELECT_CARD、CMD7)。
2. 送信するブロック長 (SET_BLOCKLEN、CMD16) を定義します。8 ビットのカードロック / アンロックモード (表 183 のバイト 0)、8 ビットの PWD_LEN、および現在のパスワードのバイト数によって定義されます。
3. 16 ビットの CRC を含めた、データライン上の適切なデータブロックサイズとともに LOCK/UNLOCK (CMD42) を送信します。データブロックは、モード (LOCK_UNLOCK = 0)、長さ (PWD_LEN)、およびパスワード (PWD) そのものを示します。
4. パスワードが一致すると、カードはアンロックされ、カードステータスレジスタの CARD_IS_LOCKED ステータスビットがクリアされます。送信されたパスワードのサイズまたは内容が、予期されたパスワードと一致しないときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、カードはロックされたままです。

アンロック機能は、現在のパワーセッションでのみ有効です。PWD フィールドがクリアされていないときには、カードは次のパワーアップで自動的にロックされます。

アンロックされたカードをアンロックしようすると、アンロックは失敗し、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

強制消去

ユーザがパスワード (PWD の内容) を忘れた場合は、カード上のすべてのデータをクリアした後で、カードにアクセスすることができます。この強制消去操作では、すべてのカードデータとすべてのパスワードデータが消去されます。

1. まだ選択されていない場合は、カードを選択します (SELECT/DESELECT_CARD、CMD7)。
2. ブロック長 (SET_BLOCKLEN、CMD16) を 1 バイトに設定します。8 ビットのカードロック/アンロックバイト (表 183 のバイト 0) だけが送信されます。
3. 16 ビットの CRC を含めた、データライン上の適切なデータバイトとともに LOCK/UNLOCK (CMD42) を送信します。データブロックは、モード (ERASE = 1) を示します。他のすべてのビットはゼロでなければなりません。
4. ERASE ビットがデータフィールドでセットされた唯一のビットであるときには、PWD および PWD_LEN フィールドを含めて、すべてのカード内容が消去され、カードはロックされなくなります。他にもセットされているビットがあるときには、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされ、カードはすべてのデータを保持し、ロックされたままとなります。

アンロックされたカードを強制消去しようすると、カードステータスレジスタの LOCK_UNLOCK_FAILED エラービットがセットされます。

29.4.11 カードステータスレジスタ

レスポンスフォーマット R1 には、カードステータスと呼ばれる 32 ビットのフィールドがあります。このフィールドは、カードステータス情報 (ローカルステータスレジスタに格納できます) をホストに送信するためのものです。特に記載がない限り、ステータスエントリは、常に、以前に発行されたコマンドに関連します。

表 170 に、さまざまなステータスエントリを示します。表中のタイプおよびクリア条件フィールドは、次のように略記されます。

タイプ :

- E : エラービット
- S : ステータスビット
- R : 実際のコマンドレスポンスのために検出され、セットされます。
- X : コマンド実行中に検出され、セットされます。SDIO カードホストは、ステータスコマンドを発行することによってカードにポーリングして、これらのビットを読み出さなければなりません。

クリア条件 :

- A : カードの現在の状態に応じて。
- B : 常に前のコマンドに関連して。有効なコマンドを受信すると、クリアされます (1 コマンドの遅延で)。
- C : 読み出しによってクリアされます。

表 170. カードステータス

ビット	識別子	タイプ	値	説明	クリア条件
31	ADDRESS_OUT_OF_RANGE	E R X	"0" = エラーなし "1" = エラー	コマンドアドレス引数が、このカードの許容範囲を超えています。 複数ブロックまたはストリーム読み出し/書き込み操作が、(有効なアドレスから開始していますが) カード容量を超える読み込みまたは書き込みを試みました。	C
30	ADDRESS_MISALIGN	-	"0" = エラーなし "1" = エラー	コマンドアドレス引数 (現在設定されているブロック長に従って) が指している最初のデータブロックは、カードの物理ブロックに揃っていません。 複数ブロック読み出し/書き込み操作は、(有効なアドレスから開始していますが) カードの物理ブロックに揃っていないデータブロックの読み込みまたは書き込みを試みています。	C
29	BLOCK_LEN_ERROR	-	"0" = エラーなし "1" = エラー	SET_BLOCKLEN コマンドの引数がカードの許容する最大値を超えているか、以前に定義されたブロック長が現在のコマンドに対して無効です (ホストが書き込みコマンドを発行し、現在のブロック長がカードの許容する最大値より小さく、部分的なブロックの書き込みが許されない場合など)。	C
28	ERASE_SEQ_ERROR	-	"0" = エラーなし "1" = エラー	消去コマンドのシーケンス中にエラーが発生しました。	C
27	ERASE_PARAM	E X	"0" = エラーなし "1" = エラー	消去に対して無効な消去グループの選択が発生しました。	C
26	WP_VIOLATION	E X	"0" = エラーなし "1" = エラー	書き込み保護されたブロックをプログラムしようとした。	C
25	CARD_IS_LOCKED	S R	"0" = カードはアンロック "1" = カードはロック	セットされると、カードがホストによってロックされたことを示します。	A
24	LOCK_UNLOCK_FAILED	E X	"0" = エラーなし "1" = エラー	ロック/アンロックコマンドで、シーケンスエラーまたはパスワードエラーが発生したときにセットされます。	C
23	COM_CRC_ERROR	E R	"0" = エラーなし "1" = エラー	以前のコマンドの CRC チェックが失敗しました。	B
22	ILLEGAL_COMMAND	E R	"0" = エラーなし "1" = エラー	カードの状態に対して無効なコマンドです。	B
21	CARD_ECC_FAILED	E X	"0" = 成功 "1" = 失敗	カードの内部 ECC が適用されましたが、データの訂正に失敗しました。	C
20	CC_ERROR	E R	"0" = エラーなし "1" = エラー	(標準では定義されていません) ホストコマンドに関係ないカードエラーが発生しました。	C
19	ERROR	E X	"0" = エラーなし "1" = エラー	(標準では定義されていません) 最後のホストコマンドの実行に関係した一般カードエラーが検出されました (読み出しまたは書き込みの失敗など)。	C
18	予約済み				

表 170. カードステータス (続き)

ビット	識別子	タイプ	値	説明	クリア条件
17	予約済み				
16	CID/CSD_OVERWRITE	E X	"0"= エラーなし "1"= エラー	次のいずれかのエラーです。 - CID レジスタはすでに書き込まれていて、上書きできません。 - CSD の読み出し専用セクションが、カードの内容と一致しません。 - コピーを元に戻す試み、または永久的な WP ビットを元に戻す (保護しない) 試みが行われました。	C
15	WP_ERASE_SKIP	E X	"0"= 保護されていません "1"= 保護されています	既存の書き込みが理由で、部分的なアドレス空間だけが消去されたときにセットされます。	C
14	CARD_ECC_DISABLED	S X	"0"= 有効 "1"= 無効	内部 ECC を使用せずに、コマンドが実行されました。	A
13	ERASE_RESET	-	"0"= クリア "1"= セット	誤った順序の消去コマンド (CMD35、CMD36、CMD38、または CMD13 以外の) が受信されたので、実行前に消去シーケンスがクリアされました。	C
12:9	CURRENT_STATE	S R	0 = Idle 1 = Ready 2 = Ident 3 = Stby 4 = Tran 5 = Data 6 = Rcv 7 = Prg 8 = Dis 9 = Btst 10-15 = 予約済み	コマンドを受信したときのカードの状態。コマンドの実行によってカードの状態が変化する場合、次のコマンドのレスポンスで、ホストに状態が知らされます。4 ビットは、2 進数で 0 から 15 までの数として解釈されます。	B
8	READY_FOR_DATA	S R	"0"= レディでない "1"= レディ	バス上のパッファ空信号に対応します。	-
7	SWITCH_ERROR	E X	"0"= エラーなし "1"= スイッチエラー	セットされた場合、カードは SWITCH コマンドで要求された予期されたモードに切り替わりませんでした。	B
6	予約済み				
5	APP_CMD	S R	"0"= 無効 "1"= 有効	カードは ACMD を予期しているか、コマンドが ACMD の補助として解釈されたことを示します。	C
4	SD I/O カード用に予約済み				
3	AKE_SEQ_ERROR	E R	"0"= エラーなし "1"= エラー	認証プロセスのシーケンスにエラーがあります。	C
2	アプリケーション固有のコマンド用に予約済み。				
1	製造業者のテストモード用に予約済み。				
0					

29.4.12 SD ステータスレジスタ

SD ステータスは、SD メモリカード独自の機能に関連するステータスビットを含み、将来のアプリケーション固有の用途で使用できます。SD ステータスのサイズは、512ビットのデータブロックです。このレジスタの内容は、ACMD13 が送信された場合に（CMD55 に続いて CMD13）、SDIO カードホストに送信されます。ACMD13 は、転送状態のカード（カードが選択されている）にのみ送信できます。

表 171 に、SD ステータスレジスタのさまざまなエントリを示します。表中のタイプおよびクリア条件フィールドは、次のように略記されます。

タイプ：

- E：エラービット
- S：ステータスビット
- R：実際のコマンドレスポンスのために検出され、セットされます。
- X：コマンド実行中に検出され、セットされます。SDIO カードホストは、ステータスコマンドを発行することによってカードにポーリングして、これらのビットを読み出さなければなりません。

クリア条件：

- A：カードの現在の状態に応じて。
- B：常に前のコマンドに関連して。有効なコマンドを受信すると、クリアされます（1 コマンドの遅延で）。
- C：読み出しによってクリアされます。

表 171. SD ステータス

ビット	識別子	タイプ	値	説明	クリア条件
511: 510	DAT_BUS_WIDTH	S R	"00" = 1 (デフォルト) "01" = 予約済み "10" = 4 ビット幅 "11" = 予約済み	SET_BUS_WIDTH コマンドによって定義された、現在定義されているバス幅を示します。	A
509	SECURED_MODE	S R	"0" = セキュアモードではありません。 "1" = セキュアモードです。	カードはセキュア操作モードです（「SD セキュリティ仕様」を参照）。	A
508: 496	予約済み				
495: 480	SD_CARD_TYPE	S R	"00xxh" = 物理仕様 Ver1.01-2.00 で定義された SD メモリカード ("x" = don't care) 現在定義されているカードは、次のとおりです。 "0000" = 通常の SD RD/WR カード "0001" = SD ROM カード	将来は、8 つの LSB を使用して、さまざまな種類の SD メモリカードを定義する予定です（各ビットが異なる SD タイプを定義します）。8 つの MSB は、現在の SD 物理レイヤ仕様に準拠しない SD カードを定義するために使用されます。	A
479: 448	SIZE_OF_PROTECTED_AREA	S R	保護領域のサイズ（下記を参照）	（下記を参照）	A
447: 440	SPEED_CLASS	S R	カードの速度クラス（下記を参照）	（下記を参照）	A
439: 432	PERFORMANCE_MOVE	S R	1 [MB/s] 単位で示される移動性能（下記を参照）	（下記を参照）	A
431: 428	AU_SIZE	S R	AU のサイズ（下記を参照） （下記を参照）	（下記を参照）	A



表 171. SD ステータス (続き)

ビット	識別子	タイプ	値	説明	クリア条件
427:424	予約済み				
423:408	ERASE_SIZE	S R	一度に消去される AU の数	(下記を参照)	A
407:402	ERASE_TIMEOUT	S R	UNIT_OF_ERASE_AU で指定された領域を消去する際のタイムアウト値	(下記を参照)	A
401:400	ERASE_OFFSET	S R	消去時間に追加される固定オフセット値	(下記を参照)	A
399:312	予約済み				
311:0	製造業者のために予約済み。				

SIZE_OF_PROTECTED_AREA

このフィールドの設定は、標準カードと大容量カードで異なります。標準容量のカードでは、保護領域の容量は次のように計算されます。

保護領域 = SIZE_OF_PROTECTED_AREA * MULT * BLOCK_LEN

SIZE_OF_PROTECTED_AREA は、MULT * BLOCK_LEN 単位で指定されます。

大容量のカードでは、保護領域の容量は、このフィールドで指定されます。

保護領域 = SIZE_OF_PROTECTED_AREA

SIZE_OF_PROTECTED_AREA は、バイト単位で指定されます。

SPEED_CLASS

この 8 ビットフィールドは速度クラスを示し、値は、 $P_W/2$ で求めることができます (P_W は書き込み性能)。

表 172. 速度クラスコードフィールド

SPEED_CLASS	値の定義
00h	クラス 0
01h	クラス 2
02h	クラス 4
03h	クラス 6
04h – FFh	予約済み

PERFORMANCE_MOVE

この 8 ビットフィールドは、Pm（移動パフォーマンス）を示し、値は、1 [MB/sec] 単位で設定できます。カードが使用済みの RU（記録ユニット）を移動しない場合、Pm は無限とみなす必要があります。このフィールドを FFh にセットすると、無限を意味します。

表 173. 移動パフォーマンスフィールド

PERFORMANCE_MOVE	値の定義
00h	定義されません。
01h	1 [MB/sec]
02h	02h 2 [MB/sec]
-----	-----
FEh	254 [MB/sec]
FFh	無限

AU_SIZE

この 4 ビットフィールドは、AU サイズを示し、値は、16 KB から 2 の累乗単位で選択できます。

表 174. AU_SIZE フィールド

AU_SIZE	値の定義
00h	定義されません。
01h	16 KB
02h	32 KB
03h	64 KB
04h	128 KB
05h	256 KB
06h	512 KB
07h	1 MB
08h	2 MB
09h	4 MB
Ah – Fh	予約済み

最大 AU サイズは、カード容量に依存し、表 175 に示されています。カードは、RU サイズと最大 AU サイズの間の任意の AU サイズに設定できます。

表 175. 最大 AU サイズ

容量	16 MB-64 MB	128 MB-256 MB	512 MB	1 GB-32 GB
最大 AU サイズ	512 KB	1 MB	2 MB	4 MB

ERASE_SIZE

この 16ビットフィールドは、NERASE を示します。AU の NERASE 数が消去されるときの、タイムアウト値は ERASE_TIMEOUT によって指定されます ([ERASE_TIMEOUT](#) を参照)。ホストは、1 回の操作で消去される正しい AU の数を決定する必要があります。それにより、ホストは消去操作の進捗状況を表示することができます。このフィールドが 0 にセットされた場合、消去タイムアウト計算はサポートされません。

表 176. 消去サイズフィールド

ERASE_SIZE	値の定義
0000h	消去タイムアウトの計算はサポートされません。
0001h	1 AU
0002h	2 AU
0003h	3 AU
-----	-----
FFFFh	65535 AU

ERASE_TIMEOUT

この 6ビットフィールドは、TERASE を示し、値は、ERASE_SIZE によって指定された複数の AU が消去されるときのオフセットからの消去タイムアウトを示します。ERASE_TIMEOUT の範囲は、63 秒までの間で定義でき、カード製造業者は実装に応じて、ERASE_SIZE と ERASE_TIMEOUT の任意の組み合わせを選ぶことができます。ERASE_TIMEOUT を決めると、ERASE_SIZE が決まります。

表 177. 消去タイムアウトフィールド

ERASE_TIMEOUT	値の定義
00	消去タイムアウトの計算はサポートされません。
01	1 [sec]
02	2 [sec]
03	3 [sec]
-----	-----
63	63 [sec]

ERASE_OFFSET

この 2ビットフィールドは、TOFFSET を示し、4 つの値から 1 つを選択できます。このフィールドは、ERASE_SIZE および ERASE_TIMEOUT フィールドが 0 にセットされている場合は意味を持ちません。

表 178. 消去オフセットフィールド

ERASE_OFFSET	値の定義
0h	0 [sec]
1h	1 [sec]
2h	2 [sec]
3h	3 [sec]

29.4.13 SD I/O モード

SD I/O 割り込み

SD I/O カードから MultiMediaCard/SD モジュールへの割り込みを可能にするために、SD インタフェースのピンに割り込み機能が用意されています。ピン 8 は、4ビット SD モードで動作するとき SDIO_D1 として使用され、MultiMediaCard/SD モジュールへのカード割り込みを示します。割り込みの使用は、各カードまたはカード内の各機能にとってオプションです。SD I/O 割り込みはレベル対応です。すなわち、割り込みラインは、MultiMediaCard/SD モジュールによって認識され、割り込みが処理されるか、割り込み期間の終了によってデアサートされるまで、アクティブ（ロー）に保たれる必要があります。MultiMediaCard/SD モジュールが割り込み処理を終えた後、割り込みステータスビットは、SD I/O カードの内部レジスタの該当するビットへの I/O 書き込みによってクリアされます。すべての SD I/O カードの割り込み出力はアクティブローであり、アプリケーションで、すべてのデータライン (SDIO_D[3:0]) 上に外部プルアップ抵抗を実装する必要があります。MultiMediaCard/SD モジュールは、割り込み期間に限って、割り込み検出器にピン 8 (SDIO_D/IRQ) のレベルをサンプリングします。その他の状況では、MultiMediaCard/SD モジュールはこの値を無視します。

割り込み期間は、メモリと I/O の両方の操作に適用されます。単一ブロックの操作の割り込み期間の定義は、複数ブロックのデータ転送の定義と異なります。

SD I/O のサスペンドとレジューム

マルチファンクション SD I/O または I/O とメモリの両方の機能を持つカードでは、MMC/SD バスへのアクセスを共有する複数のデバイス (I/O とメモリ) があります。複数デバイス間で MMC/SD モジュールへのアクセスを共有するために、SD I/O およびコンボカードはサスペンド/レジュームの概念をオプションで実装します。カードがサスペンド/レジュームをサポートするときには、MMC/SD モジュールは 1 つの機能またはメモリへのデータ転送を一時的に中止 (サスペンド) して、別の機能またはメモリへの、より優先順位の高い転送のためにバスを解放することができます。この優先順位の高い転送が完了した後、本来の転送が、中止された箇所から再開 (リスタート) されます。サスペンド/レジュームのサポートは、カードごとのオプションです。MMC/SD バス上でサスペンド/レジューム操作を行うには、MMC/SD モジュールは以下の手順を実行します。

1. SDIO_D [3:0] ラインを現在使用している機能を特定します。
2. 優先順位が低いトランザクションや時間のかかるトランザクションをサスペンドするように要求します。
3. トランザクションのサスペンドの完了を待ちます。
4. 優先順位の高いトランザクションを開始します。
5. 優先順位の高いトランザクションの完了を待ちます。
6. サスペンドされたトランザクションをリストアします。

SD I/O ReadWait

オプションの ReadWait (RW) 操作は、SD の 1ビットおよび 4ビットモードについてのみ定義されます。ReadWait 操作によって、MMC/SD モジュールは、複数のレジスタを読み出していることをカードに知らせて (IO_RW_EXTENDED、CMD53)、データ転送を一時的にストールし、その間に MMC/SD モジュールは SD I/O デバイス内の機能にコマンドを送信することができます。カードが ReadWait プロトコルをサポートしているかどうかを判断するには、MMC/SD モジュールは内部カードレジスタの機能ビットをテストする必要があります。ReadWait のタイミングは、割り込み期間に基づきます。

29.4.14 コマンドとレスポンス

アプリケーション固有コマンドと汎用コマンド

SD カードホストモジュールは、さまざまな種類のアプリケーションに標準インタフェースを提供するように設計されています。この環境では、特定の顧客/アプリケーション機能に対するニーズがあります。これらの機能を実装するために、この標準では 2 種類のコマンドが定義されています。すなわち、アプリケーション固有コマンド (ACMD) と汎用コマンド (GEN_CMD) です。

カードが APP_CMD (CMD55) コマンドを受信すると、カードは次のコマンドがアプリケーション固有コマンドであることを予期します。ACMD は、通常の MultiMediaCard コマンドと同じ構造であり、同じ CMD 番号にすることができます。カードは、APP_CMD (CMD55) の後に現れたコマンドを ACMD として認識します。APP_CMD (CMD55) のすぐ後に続くコマンドが定義済みのアプリケーション固有コマンドでないときには、標準コマンドが使用されます。たとえば、カードで SD_STATUS (ACMD13) が定義されているときには、APP_CMD (CMD55) の直後に CMD13 を受信すると、これは SD_STATUS (ACMD13) であると解釈されます。ただし、カードが APP_CMD (CMD55) の直後に CMD7 を受信し、カードで ACMD7 が定義されていないときには、これは標準 (SELECT/DESELECT_CARD) CMD7 であると解釈されます。

1 つの製造業者固有の ACMD を使用するには、SD カードホストは、以下の手順を実行する必要があります。

1. APP_CMD (CMD55) を送信します。
カードは、MultiMediaCard/SD モジュールにตอบสนองして、APP_CMD ビットがセットされていることと、ACMD が予期されていることを示します。
2. 必要な ACMD を送信します。
カードは、MultiMediaCard/SD モジュールにตอบสนองして、APP_CMD ビットがセットされていることと、受け取ったコマンドが ACMD として解釈されたことを示します。ACMD でないコマンドが送信されたときには、カードは通常の MultiMediaCard コマンドとして扱い、カードステータスレジスタの APP_CMD ビットはクリアされたままとなります。

無効なコマンド (ACMD でも CMD でもない) が送信されたときには、標準の MultiMediaCard 違反コマンドエラーとして処理されます。

GEN_CMD のバストランザクションは、単一ブロックの読み出しまたは書き込みコマンド (WRITE_BLOCK、CMD24、または READ_SINGLE_BLOCK、CMD17) と同じです。この場合、引数は、アドレスではなく、データ転送の方向を示し、データブロックはベンダ固有のフォーマットと意味を持ちます。

カードは、GEN_CMD (CMD56) を送信する前に選択されなければなりません (転送状態でなければなりません)。データブロックサイズは、SET_BLOCKLEN (CMD16) によって定義されます。GEN_CMD (CMD56) へのレスポンスは、R1b フォーマットです。

コマンドタイプ

アプリケーション固有コマンドと汎用コマンドは、どちらも、次の 4 つのタイプに分類されます。

- **ブロードキャストコマンド (BC)** : すべてのカードに送信されます。レスポンスは返されません。
- **レスポンス付きブロードキャストコマンド (BCR)** : すべてのカードに送信されます。レスポンスは、すべてのカードから同時に受信されます。
- **アドレス指定 (ポイントツーポイント) コマンド (AC)** : 選択されたカードに送信されます。SDIO_D ライン上のデータ転送を含みません。
- **アドレス指定 (ポイントツーポイント) データ転送コマンド (ADTC)** : 選択されたカードに送信されます。SDIO_D ライン上のデータ転送を含みます。

コマンドフォーマット

コマンドフォーマットについては、表 163 (833 ページ) を参照してください。

MultiMediaCard/SD モジュール用コマンド

表 179. ブロック指向書き込みコマンド

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD23	ac	[31:16] 0 に セット [15:0] ブロック数	R1	SET_BLOCK_COUNT	続く複数ブロック読み出しまたは書き込みコマンドで転送されるブロック数を定義します。
CMD24	adtc	[31:0] データ アドレス	R1	WRITE_BLOCK	SET_BLOCKLEN コマンドによって選択されたサイズのブロックを書き込みます。
CMD25	adtc	[31:0] データ アドレス	R1	WRITE_MULTIPLE_BLOCK	続く STOP_TRANSMISSION まで、または、要求されたブロック数が受信されるまで、データのブロックを書き込み続けます。
CMD26	adtc	[31:0] スタッ ビット	R1	PROGRAM_CID	カード識別レジスタのプログラミング。このコマンドは、1 つのカードにつき一度だけ発行しなければなりません。カードは、最初のプログラミングの後、この操作を防ぐためのハードウェアを含んでいます。通常、このコマンドは製造業者用に予約済みです。
CMD27	adtc	[31:0] スタッ ビット	R1	PROGRAM_CSD	CSD のプログラマブルビットのプログラミング。

表 180. ブロック指向書き込み保護コマンド

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD28	ac	[31:0] データ アドレス	R1b	SET_WRITE_PROT	カードに書き込み保護機能がある場合、このコマンドは、アドレス指定されたグループの書き込み保護ビットをセットします。書き込み保護のプロパティは、カード固有データ (WP_GRP_SZIE) にコード化されます。
CMD29	ac	[31:0] データ アドレス	R1b	CLR_WRITE_PROT	カードが書き込み保護機能を備えている場合、このコマンドは、アドレス指定されたグループの書き込み保護ビットをクリアします。
CMD30	adtc	[31:0] 書き込み 保護データ アクセス	R1	SEND_WRITE_PROT	カードが書き込み保護機能を備えている場合、このコマンドは、カードに書き込み保護ビットのステータスを送るよう要求します。
CMD31	予約済み				

表 181. 消去コマンド

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD32 ... CMD34		予約済み。これらのコマンドインデックスは、古いバージョンの MultiMediaCard との下位互換性を保つために、使用できません。			
CMD35	ac	[31:0] データアドレス	R1	ERASE_GROUP_START	消去対象として選択される範囲の最初の消去グループのアドレスを設定します。
CMD36	ac	[31:0] データアドレス	R1	ERASE_GROUP_END	消去対象として選択される連続した範囲内の最後の消去グループのアドレスを設定します。
CMD37		予約済み。このコマンドインデックスは、古いバージョンの MultiMediaCard との下位互換性を保つために、使用できません。			
CMD38	ac	[31:0] スタッフビット	R1	ERASE	以前に選択されたすべての書き込みブロックを消去します。

表 182. I/O モードコマンド

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD39	ac	[31:16] RCA [15:15] レジスタ書き込みフラグ [14:8] レジスタアドレス [7:0] レジスタデータ	R4	FAST_IO	8 ビット (レジスタ) データフィールドの読み出しおよび書き込みに使用されます。このコマンドは、カードとレジスタをアドレス指定し、書き込みフラグがセットされている場合は書き込み用データを提供します。R4 レスポンスは、アドレス指定されたレジスタから読み出されたデータを含みます。このコマンドは、MultiMediaCard 標準で定義されていないアプリケーション依存レジスタにアクセスします。
CMD40	bcr	[31:0] スタッフビット	R5	GO_IRQ_STATE	システムを割り込みモードにします。
CMD41		予約済み			

表 183. カードのロック

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD42	adtc	[31:0] スタッフビット	R1b	LOCK_UNLOCK	パスワードのセット/リセット、またはカードのロック/アンロックを行います。データブロックのサイズは、SET_BLOCK_LEN コマンドによって設定されます。
CMD43 ...CMD54		予約済み			

表 184. アプリケーション固有コマンド

CMD インデックス	タイプ	引数	レスポンス フォーマット	略語	説明
CMD55	ac	[31:16] RCA [15:0] スタッフトビット	R1	APP_CMD	次のコマンドビットが標準コマンドでなく、アプリケーション固有コマンドであることをカードに示します。
CMD56	adtc	[31:1] スタッフトビット [0]: RD/WR			汎用/アプリケーション固有コマンドで、データブロックをカードに転送するため、またはカードからデータを取得するために使用されます。データブロックのサイズは、SET_BLOCK_LEN コマンドによって設定されます。
CMD57 ...CMD59	予約済み。				
CMD60 ...CMD63	製造業者用に予約済み。				

29.5 レスポンスフォーマット

すべてのレスポンスは MCCMD コマンドライン SDIO_CMD 経由で送信されます。レスポンスの送信は、常に、レスポンスコードワードに対応するビット文字列の左のビットから始まります。コード長は、レスポンスのタイプに依存します。

レスポンスは、常に、開始ビット（常に 0）から始まり、送信方向を示すビットが続きます（カード = 0）。下の表で x で示されている値は、可変エントリを示します。R3 レスポンスを除くすべてのレスポンスは、CRC によって保護されます。すべてのコマンドコードワードは、終了ビット（常に 1）で終わります。

レスポンスには 5 つのタイプがあります。フォーマットは、次のとおりです。

29.5.1 R1（ノーマルレスポンスコマンド）

コード長 = 48ビット。45:40 ビットはレスポンスされるコマンドのインデックスを示します。この値は 2 進コード番号（0 ~ 63）として解釈されます。カードのステータスは 32ビットにコード化されます。

表 185. R1 レスポンス

ビット位置	幅（ビット数）	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	X	コマンドインデックス
[39:8]	32	X	カード・ステータス
[7:1]	7	X	CRC7
0	1	1	終了ビット

29.5.2 R1b

R1 と同じですが、オプションとしてデータライン上で送信されるビジー信号を含みます。カードは、コマンドを受信する前の状態に基づいて、これらのコマンドを受信した後、ビジーになることがあります。

29.5.3 R2 (CID、CSD レジスタ)

コード長 = 136 ビット。CID レジスタの内容は、CMD2 および CMD10 コマンドのレスポンスとして送信されます。CSD レジスタの内容は、CMD9 のレスポンスとして送信されます。CID および CSD のビット [127... 1] だけが転送されます。これらのレジスタの予約ビット [0] は、レスポンスの終了ビットで置き換えられます。カードは、MCDAT をローに維持することによって、消去が進行中であることを示します。実際の消去時間はかなり長く、ホストは CMD7 を発行して、カードを選択解除できます。

表 186. R2 レスポンス

ビット位置	幅 (ビット数)	値	説明
135	1	0	スタートビット
134	1	0	送信ビット
[133:128]	6	"111111"	コマンドインデックス
[127:1]	127	X	カードステータス
0	1	1	終了ビット

29.5.4 R3 (OCR レジスタ)

コード長 = 48 ビット。OCR レジスタの内容は、CMD1 へのレスポンスとして送信されます。レベルのコーディングは、次のようになります。制限された電圧ウィンドウ=ロー、カードビジー=ロー。

表 187. R3 レスポンス

ビット位置	幅 (ビット数)	値	説明
47	1	0	スタートビット
46	1	0	送信ビット
[45:40]	6	"111111"	予約済み
[39:8]	32	X	OCR レジスタ
[7:1]	7	"1111111"	予約済み
0	1	1	終了ビット

29.5.5 R4 (高速 I/O)

コード長 = 48ビット。引数フィールドは、アドレス指定されたカードの RCA、読み出しまたは書き込みレジスタアドレス、およびその内容を含みます。

表 188. R4 レスポンス

ビット位置		幅 (ビット数)	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	"100111"	CMD39
[39:8] 引数フィールド	[31:16]	16	X	RCA
	[15:8]	8	X	レジスタアドレス
	[7:0]	8	X	読み出しレジスタの内容
[7:1]		7	X	CRC7
0		1	1	終了ビット

29.5.6 R4b

SD I/O のみ: CMD5 を受信した SDIO カードは、一意な SDIO レスポンス R4 で応答します。フォーマットは、次のとおりです。

表 189. R4b レスポンス

ビット位置		幅 (ビット数)	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	x	予約済み
[39:8] 引数フィールド	39	16	X	カードはレディ状態
	[38:36]	3	X	I/O 機能数
	35	1	X	メモリあり
	[34:32]	3	X	スタッフビット
	[31:8]	24	X	I/O ORC
[7:1]		7	X	予約済み
0		1	1	終了ビット

SD I/O カードが CMD5 を受信すると、カードの I/O 部分は、以降のすべてのコマンドに対して通常どおりに応答できるようになります。I/O カード内の機能のこの I/O イネーブルは、カードがリセット、パワーサイクル、または I/O リセットするための書き込みの CMD52 を受信するまで、セットされたままです。SD メモリ専用カードは、CMD5 に応答する場合があることに注意してください。メモリ専用カードの正しいレスポンスは、*Present memory* = 1 と *Number of I/O functions* = 0 です。SD メモリカード仕様 version 1.0 に準拠するように作られたメモリ専用カードは、CMD5 を不正なコマンドとして検出して、応答しません。I/O を認識できるホストは、CMD5 を送信します。カードがレスポンス R4 で応答した場合、ホストはレスポンス R4 に含まれるデータから、カードの構成を判断します。



29.5.7 R5 (割り込みリクエスト)

MultiMediaCard 専用です。コード長 = 48ビット。ホストによってレスポンスが生成された場合、引数の RCA フィールドは 0x0 になります。

表 190. R5 レスポンス

ビット位置		幅 (ビット数)	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	"101000"	CMD40
[39:8] 引数フィールド	[31:16]	16	X	勝ったカードまたはホストの RCA [31:16]
	[15:0]	16	X	定義されていません。IRQ データとして使われる場合があります。
[7:1]		7	X	CRC7
0		1	1	終了ビット

29.5.8 R6

SD I/O 専用です。メモリデバイスによる CMD3 への通常のレスポンス。これを [表 191](#) に示します。

表 191. R6 レスポンス

ビット位置		幅 (ビット数)	値	説明
47		1	0	スタートビット
46		1	0	送信ビット
[45:40]		6	"101000"	CMD40
[39:8] 引数フィールド	[31:16]	16	X	勝ったカードまたはホストの RCA [31:16]
	[15:0]	16	X	定義されていません。IRQ データとして使われる場合があります。
[7:1]		7	X	CRC7
0		1	1	終了ビット

カード [23:8] ステータスビットは、CMD3 が I/O 専用カードに送信されると、変更されます。この場合、レスポンスの 16ビットは、SD I/O 専用の値です。

- ビット [15] COM_CRC_ERROR
- ビット [14] ILLEGAL_COMMAND
- ビット [13] ERROR
- ビット [12:0] 予約済み

29.6 SDIO I/O カード固有の操作

以下の機能は、SD I/O 固有の操作です。

- SDIO_D2 信号による SDIO 読み出しウェイト操作
- クロックの停止による SDIO 読み出しウェイト操作
- SDIO サスペンド/レジューム操作（書き込みおよび読み出しサスペンド）
- SDIO 割り込み

SDIO は、SDIO_DCTRL[11] ビットがセットされている場合のみ、これらの機能をサポートします。ただし、特定のハードウェア実装を必要としない読み出しサスペンドを除きます。

29.6.1 SDIO_D2 信号による SDIO I/O 読み出しウェイト操作

最初のブロックが受信される前に読み出しウェイトインターバルを開始することができます。データパスが有効（SDIO_DCTRL[0] ビットがセット）、SDIO 固有操作が有効（SDIO_DCTRL[11] ビットがセット）、読み出しウェイト開始（SDIO_DCTRL[10]=0 および SDI_DCTRL[8]=1）、データ方向がカードから SDIO（SDIO_DCTRL[1]=1）のとき、DPSM はアイドルから ReadWait へ直接移行します。ReadWait では、DPSM は 2 SDIO_CK クロックサイクル後に、SDIO_D2 を 0 に駆動します。この状態では、RWSTOP ビット（SDIO_DCTRL[9]）をセットすると、DPSM は、さらに 2 SDIO_CK クロックサイクルの間、ウェイト状態にとどまり、1 クロックサイクルで SDIO_D2 を 1 に駆動します（SDIO 仕様に従って）。その後、DPSM は、カードからデータを受信するまで、再び待機を開始します。DPSM は、読み出しウェイト開始がセットされている場合でも、ブロックを受信中は読み出しウェイトインターバルを開始しません。読み出しウェイトインターバルは、CRC の受信後に開始されます。新しい読み出しウェイト操作を開始するには、RWSTOP ビットをクリアする必要があります。読み出しウェイトインターバル中は、SDIO は SDIO_D1 上で SDIO 割り込みを検出することができます。

29.6.2 SDIO_CK 停止による SDIO 読み出しウェイト操作

SDIO カードが一昔前の読み出しウェイト方式をサポートしない場合、この SDIO は SDIO_CK を停止することによって読み出しウェイトを行うことができます（SDIO_DCTRL は [セクション 29.6.1](#) で示した方式と同様にセットされますが、SDIO_DCTRL[10]=1 となります）：DPSM は、現在の受信ブロックの終了ビット後、SDIO_CK を 2 サイクル分のクロックを停止し、読み出しウェイト開始ビットがセットされた後、クロックを再開させます。

SDIO_CK が停止されるので、任意のコマンドをカードに発行できます。読み出しウェイトインターバル中、SDIO は SDIO_D1 上の SDIO 割り込みを検出できます。

29.6.3 SDIO サスペンド/レジューム操作

カードにデータを送信しているときに、SDIO は書き込み操作をサスペンドできます。SDIO_CMD[11] ビットがセットされて、現在のコマンドがサスペンドコマンドであることを CPSM に示します。CPSM は、レスポンスを解析し、カードから ACK を受信すると（サスペンドが受け入れられた）、現在のブロックの CRC トークンの受信後にアイドル状態となることを DPSM に知らせます。

ハードウェアは、サスペンド操作（レジューム）を完了するために、残りの送信ブロック数を保存しません。

書き込み操作は、カードからサスペンドコマンドの ACK を受信したときに、DPSM を無効にするだけで（SDIO_DCTRL[0]=0）、ソフトウェアによってサスペンドできます。その場合、DPSM はアイドル状態に入ります。

読み出しをサスペンドするには : DPSM は Wait_r 状態で、サスペンドされる機能が完全なパケットを送信するのを待ってから、データ処理を停止します。アプリケーションは、FIFO が空になって、DPSM が自動的にアイドル状態になるまで、RxFIFO の読み出しを続けます。

29.6.4 SDIO 割り込み

SDIO_DCTRL[11] ビットがセットされると、SDIO_D1 ライン上で SDIO 割り込みが検出されます。

29.7 CE-ATA 固有の操作

以下の機能は、CE-ATA 固有の操作です。

- CE-ATA デバイスを無効にするコマンド完了信号の送信。
- CE-ATA デバイスからのコマンド完了信号の受信。
- ステータスビットや割り込みを使用して、CE-ATA コマンドの完了を CPU に通知。

SDIO は、SDIO_CMD[14] がセットされた場合、CE-ATA CMD61 コマンドについてのみ、これらの操作をサポートします。

29.7.1 コマンド完了信号ディセーブル

コマンド完了信号ディセーブルは、"CMD 完了イネーブル" ビット (SDIO_CMD[12]) がセットされず、"割り込みディセーブル" ビット (SDIO_CMD[13]) がセットされている場合に、**short** レスポンスの受信から 8 ビットサイクル後に送信されます。

CPSM は Pend 状態に入り、コマンドシフトレジスタにディセーブルシーケンス "00001" をロードし、コマンドカウンタに 43 をロードします。8 サイクル後、トリガは CPSM を送信状態へ移行します。コマンドカウンタが 48 に達すると、CPSM は、待ち受けるレスポンスがないので、アイドル状態になります。

29.7.2 コマンド完了信号イネーブル

"CMD 完了イネーブル" ビット (SDIO_CMD[12]) がセットされ、"割り込みディセーブル" ビット (SDIO_CMD[13]) がセットされた場合、CPSM は Waitcpl 状態でコマンド完了信号を待ちます。

CMD ライン上で "0" が受信されると、CPSM はアイドル状態に入ります。7 ビットサイクルの間、新しいコマンドを送信することはできません。7 サイクルのうちの最後の 5 サイクルの間、CMD ラインはプッシュプルモードで "1" に駆動されます。

29.7.3 CE-ATA 割り込み

コマンド完了は、ステータスビット SDIO_STA[23] によって CPU に通知されます。この静的ビットは、クリアビット SDIO_ICR[23] によってクリアできます。

SDIO_STA[23] ステータスビットは、マスクビット SDIO_MASKx[23] に応じて、各割り込みライン上に割り込みを生成することができます。

29.7.4 CMD61 のアボート

コマンド完了ディセーブル信号が送信されていず、CMD61 をアボートする必要がある場合には、コマンドステートマシンを無効にしなければなりません。コマンドステートマシンはアイドル状態になり、CMD12 コマンドを送信することができます。操作中は、コマンド完了ディセーブル信号は送信されません。

29.8 HW フロー制御

HW フロー制御機能は、FIFO アンダーラン (TX モード) およびオーバーラン (RX モード) エラーを回避するために使用されます。

その動作は、SDIO_CLK を停止して、SDIO ステートマシンをフリーズします。FIFO を使用してデータを送受信できない間、データ転送はストールします。SDIOCLK によってクロック供給されるステートマシンだけがフリーズし、APB2 インタフェースは動作を続けます。したがって、フロー制御が有効でも、FIFO が満杯になったり、空になることがあります。

HW フロー制御を有効にするには、SDIO_CLKCR[14] レジスタビットを 1 にセットしなければなりません。リセット後、フロー制御は無効になります。

29.9 SDIO レジスタ

デバイスは、APB2 経由でアクセス可能な 32ビット幅の制御レジスタによってシステムと通信します。

パリティレジスタには、ワード (32ビット) 単位でアクセスする必要があります。

29.9.1 SDIO 電源制御レジスタ (SDIO_POWER)

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	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み																PWRCT		RL													
																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 1:0 **PWRCTRL** : 電源制御ビット

これらのビットは、カードクロックの現在の機能状態を定義するために使用されます。

00 : パワーオフ : カードへのクロック供給は停止されます。

01 : 予約済み

10 : 予約済み パワーアップ

11 : パワーオン : カードにクロックが供給されます。

注 : このレジスタへの書き込みアクセス間隔は、HCLK クロック 7 周期分以上が必要です。

注 : データ書き込み後、このレジスタには、SDIOCLK (48 MHz) クロック 3 周期分プラス PCLK2 クロック 2 周期分の間、書き込むことができません。

29.9.2 SDI クロック制御レジスタ (SDIO_CLKCR)

アドレスオフセット : 0x04

リセット値 : 0x0000 0000

SDIO_CLKCR レジスタは、SDIO_CK 出力クロックを制御します。

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					
予約済み																	HWFC_EN	NEGEDGE	WID BUS		BYPASS	PWRSAP	CLKEN	CLKDIV												
																	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	r/w	r/w	r/w	r/w

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 **HWFC_EN** : HW フロー制御イネーブル

0b : HW フロー制御は無効です。

1b : HW フロー制御は有効です。

HW フロー制御が有効なときの TXFIFOE および RXFIFOE 割り込み信号の意味については、[セクション 29.9.11](#) の SDIO ステータスレジスタの定義を参照してください。

ビット 13 **NEGEDGE** : SDIO_CK 反位相選択ビット

0b : SDIO_CK は、マスタクロック SDIOCLK の立ち上がりエッジで生成されます。

1b : SDIO_CK は、マスタクロック SDIOCLK の立ち下がりエッジで生成されます。

ビット 12:11 **WIDBUS** : ワイドバスモードイネーブルビット

00 : デフォルトバスモード : SDIO_D0 が使用されます

01 : 4 ワイドバスモード : SDIO_D[3:0] が使用されます

10 : 8 ワイドバスモード : SDIO_D[7:0] が使用されます

ビット 10 **BYPASS** : クロック分周器バイパスイネーブルビット

0 : バイパスディセーブル : SDIOCLK は、SDIO_CK を駆動する前に、CLKDIV の値に従って分周されます。

1 : バイパスイネーブル : SDIOCLK は、SDIO_CK 出力信号を直接駆動します。

ビット 9 **PWRSAP** : 節電設定ビット

節電のために、バスがアイドル状態のときには、PWRSAP をセットすることによって SDIO_CK クロック出力を無効にできます。

0 : SDIO_CK クロックは常に有効です。

1 : SDIO_CK はバスがアクティブなときに限り有効です。

ビット 8 **CLKEN** : クロックイネーブルビット

0 : SDIO_CK は無効です。

1 : SDIO_CK は有効です。

ビット 7:0 **CLKDIV** : クロック分周ファクタ

このフィールドは、入力クロック (SDIOCLK) と出力クロック (SDIO_CK) の間の分周ファクタを定義します : $SDIO\_CK \text{ 周波数} = SDIOCLK / [CLKDIV + 2]$

注 : *SD/SDIO カードまたは MultiMediaCard が識別モードのときには、SDIO_CK 周波数は 400 kHz 未満でなければなりません。*

クロック周波数は、相対カードアドレスがすべてのカードに割り当てられているときには、最大カードバス周波数に変更できます。

データ書き込み後、このレジスタには、SDIOCLK (48 MHz) クロック 3 周期分プラス PCLK2 クロック 2 周期分の間、書き込むことができません。SDIO_CK は、SD I/O カードの読み出しウェイトイン

ターバルの間、停止することができます。この場合、SDIO_CLKCR レジスタは SDIO_CK を制御しません。

29.9.3 SDIO 引数レジスタ (SDIO_ARG)

アドレスオフセット : 0x08

リセット値 : 0x0000 0000

SDIO_ARG レジスタは、コマンドメッセージの一部としてカードに送信される 32ビットのコマンド引数を含みます。

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
CMDARG																															
RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

ビット 31:0 **CMDARG** : コマンド引数
コマンドメッセージの一部としてカードに送信されるコマンド引数。コマンドが引数を含む場合、コマンドをコマンドレジスタに書き込む前に、このレジスタにロードする必要があります。

29.9.4 SDIO コマンドレジスタ (SDIO_CMD)

アドレスオフセット : 0x0C

リセット値 : 0x0000 0000

SDIO_CMD レジスタは、コマンドインデックスおよびコマンドタイプビットを含みます。コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。コマンドタイプビットは、コマンドバスステートマシン (CPSM) を制御します。

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		
予約済み																	CE-ATACMD	nIEN	ENCMDcompl	SDIOSuspend	CPSMEN	WAITPEND	WAITINT	WAITRESP	CMDINDEX								
																	rW	rW	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 **ATACMD** : CE-ATA コマンド
ATACMD がセットされた場合、CPSM は CMD61 を送信します。
- ビット 13 **nIEN** : 割り込みディセーブル
このビットが 0 の場合、CE-ATA の割り込みは有効です。
- ビット 12 **ENCMDcompl** : CMD 完了イネーブル
このビットがセットされた場合、コマンド完了信号は有効です。
- ビット 11 **SDIOSuspend** : SD I/O サスペンドコマンド
このビットがセットされた場合、送信されるコマンドはサスペンドコマンドです (SDIO カードでのみ使用されます)。
- ビット 10 **CPSMEN** : コマンドバスステートマシン (CPSM) イネーブルビット
このビットがセットされた場合、CPSM は有効です。
- ビット 9 **WAITPEND** : CPSM のデータ転送終了待ち (CmdPend 内部信号)。
このビットがセットされた場合、CPSM は、データ転送の終了を待ってから、コマンドの送信を開始します。
- ビット 8 **WAITINT** : CPSM の割り込みリクエスト待ち
このビットがセットされた場合、CPSM は、コマンドタイムアウトを無効にして、割り込みリクエストを待ちます。
- ビット 7:6 **WAITRESP** : レスポンスビット待ち
これらは、CPSM がレスポンスを待つかどうか、および、待つ場合にはレスポンスの種類を構成するために使用されます。
00 : レスポンスなし。CMDSENT フラグを待ちます。
01 : ショートレスポンス。CMDREND または CCRCFAIL フラグを待ちます。
10 : レスポンスなし。CMDSENT フラグを待ちます。
11 : ロングレスポンス。CMDREND または CCRCFAIL フラグを待ちます。
- ビット 5:0 **CMDINDEX** : レスポンスコマンドインデックス
コマンドインデックスは、コマンドメッセージの一部としてカードに送信されます。

注 : データ書き込み後、このレジスタには、SDIOCLK (48 MHz) クロック 3 周期分プラス PCLK2 クロック 2 周期分の間、書き込むことができません。

MultiMediaCards はショートレスポンス (48 ビット長) またはロングレスポンス (136 ビット長) の 2 種類のレスポンスを送信できます。SD カードと SD I/O カードはショートレスポンスのみ送信で

き、引数はレスポンスのタイプに応じて変化します。ソフトウェアは、送信されたコマンドに応じてレスポンスのタイプを区別します。CE-ATA デバイスは、ショートレスポンスのみを送信します。

29.9.5 SDIO コマンドレスポンスレジスタ (SDIO_RESPCMD)

アドレスオフセット : 0x10

リセット値 : 0x0000 0000

SDIO_RESPCMD レジスタは、最後に受信されたコマンドレスポンスのコマンドインデックスフィールドを含みます。コマンドレスポンス送信がコマンドインデックスフィールドを含まない場合（ロングまたは OCR レスポンス）、RESPCMD フィールドは未知であり、111111b（レスポンスの予約フィールドの値）を含まなければならないかもしれません。

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
予約済み																								RESPCMD							
																								r	r	r	r	r	r		

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 5:0 **RESPCMD** : レスポンスコマンドインデックス
読み取り専用ビットフィールド。最後に受信されたコマンドレスポンスのコマンドインデックスを含みます。

29.9.6 SDIO レスポンス1..4 レジスタ (SDIO_RESPx)

アドレスオフセット : (0x10 + (4 × x)) ; x = 1..4

リセット値 : 0x0000 0000

SDIO_RESP0/1/2/3/4 レジスタは、受信したレスポンスの一部であるカードのステータスを含みます。

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
CARDSTATUSx																															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

ビット 31:0 **CARDSTATUSx** : 表 192 を参照してください。 .

カードステータスサイズは、レスポンスのタイプによって 32 または 127 ビットです。

表 192. レスポンスタイプと SDIO_RESPx レジスタ

レジスタ	ショートレスポンス	ロングレスポンス
SDIO_RESP1	Card Status[31:0]	Card Status [127:96]
SDIO_RESP2	使用されない	Card Status [95:64]
SDIO_RESP3	使用されない	Card Status [63:32]
SDIO_RESP4	使用されない	Card Status [31:1]0b

カードステータスの最下位ビットから先に受信されます。SDIO_RESP3 レジスタ LSB は常に 0b です。



29.9.7 SDIO データタイムレジスタ (SDIO_DTIMER)

アドレスオフセット : 0x24

リセット値 : 0x0000 0000

SDIO_DTIMER レジスタは、データタイムアウト期間（カードバスクロック周期単位）を含みます。

カウンタは SDIO_DTIMER レジスタから値をロードして、データバスステートマシンが Wait_R または ビジー状態に入ったときに、デクリメントを開始します。DPSM がこのどちらかの状態のときにタイマが 0 に達した場合、タイムアウトステータスフラグがセットされます。

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
DATETIME																															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

ビット 31:0 **DATETIME** : データタイムアウト期間
カードバスクロック周期で表されたデータタイムアウト期間。

注 : データ転送は、データ制御レジスタに書き込まれる前に、データタイムレジスタとデータ長レジスタに書き込まれなければなりません。

29.9.8 SDIO データ長レジスタ (SDIO_DLEN)

アドレスオフセット : 0x28

リセット値 : 0x0000 0000

SDIO_DLEN レジスタは、転送されるデータバイト数を含みます。値は、データ転送が開始されたときに、データカウンタにロードされます。

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
予約済み							DATALENGTH																								
							rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	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 24:0 **DATALENGTH** : データ長の値
転送されるデータバイト数。

注 : ブロックデータ転送では、データ長レジスタの値はブロックサイズの倍数でなければなりません (SDIO_CTRL を参照してください)。データ転送は、データ制御レジスタに書き込まれる前に、データタイムレジスタとデータ長レジスタに書き込まれなければなりません。
SDIO マルチバイト転送の場合、データ長レジスタの値は、1 から 512 の間である必要があります。

29.9.9 SDIO データ制御レジスタ (SDIO_DCTRL)

アドレスオフセット : 0x2C

リセット値 : 0x0000 0000

SDIO_DCTRL レジスタは、データパスステートマシン (DPSM) を制御します。

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
予約済み																				SDIOEN	RWMOD	RWSTOP	RWSTART	DBLOCKSIZE				DMAEN	DTMODE	DTDIR	DTEN
																				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 **SDIOEN** : SD I/O イネーブル機能

このビットがセットされた場合、DPSM は SD I/O カード固有の操作を実行します。

ビット 10 **RWMOD** : 読み出しウェイトモード

0 : SDIO_CK を停止する読み出しウェイト制御

1 : SDIO_CK を使用した読み出しウェイト制御

ビット 9 **RWSTOP** : 読み出しウェイト停止

0 : RWSTART ビットがセットされている場合、読み出しウェイトは進行中です。

1 : RWSTART ビットがセットされている場合、読み出しウェイト停止を有効にします。

ビット 8 **RWSTART** : 読み出しウェイト開始

このビットがセットされている場合、読み出しウェイト操作が開始されます。

ビット 7:4 **DBLOCKSIZE** : データブロックサイズ

ブロックデータ転送モードが選択されているときのデータブロック長を定義します。

0000 : (10 進数 0) ロック長 = 2^0 = 1バイト

0001 : (10 進数 1) ロック長 = 2^1 = 2バイト

0010 : (10 進数 2) ロック長 = 2^2 = 4バイト

0011 : (10 進数 3) ロック長 = 2^3 = 8バイト

0100 : (10 進数 4) ロック長 = 2^4 = 16バイト

0101 : (10 進数 5) ロック長 = 2^5 = 32バイト

0110 : (10 進数 6) ロック長 = 2^6 = 64バイト

0111 : (10 進数 7) ロック長 = 2^7 = 128バイト

1000 : (10 進数 8) ロック長 = 2^8 = 256バイト

1001 : (10 進数 9) ロック長 = 2^9 = 512バイト

1010 : (10 進数 10) ロック長 = 2^{10} = 1024バイト

1011 : (10 進数 11) ロック長 = 2^{11} = 2048バイト

1100 : (10 進数 12) ロック長 = 2^{12} = 4096バイト

1101 : (10 進数 13) ロック長 = 2^{13} = 8192バイト

1110 : (10 進数 14) ロック長 = 2^{14} = 16384バイト

1111 : (10 進数 15) 予約済み

ビット 3 **DMAEN** : DMA イネーブルビット

0 : DMA は無効です。

1 : DMA は有効です。



ビット 2 **DTMODE** : データ転送モード選択 1 : ストリームまたは SDIO マルチバイトデータ転送

0 : ブロックデータ転送

1 : ストリームまたは SDIO マルチバイトデータ転送

ビット 1 **DTDIR** : データ転送方向選択

0 : コントローラからカードへ

1 : カードからコントローラへ

ビット 0 **DTEN** : データ転送イネーブルビット

データ転送は、1b が DTEN ビットに書き込まれると開始します。方向ビット DTDIR に応じて、DPSM は、Wait_S、Wait_R、または転送の始めに直ちに RW Start がセットされた場合には ReadWait へ移行します。データ転送の終わりにイネーブルビットをクリアする必要はありませんが、新しいデータ転送を有効にするには、SDIO_DCTRL を更新する必要があります。

注 : データ書き込み後、このレジスタには、SDIOCLK (48 MHz) クロック 3 周期分プラス PCLK2 クロック 2 周期分の間、書き込むことができません。

SDIOEN ビットの値によって DTMODE ビットの意味が変わります。SDIOEN=0 かつ DTMODE=1 の場合、MultiMediaCard ストリームモードが有効化され、SDIOEN=1 かつ DTMODE=1 の場合、SDIO マルチバイト転送がペリフェラルによって有効化されます。

29.9.10 SDIO データカウンタレジスタ (SDIO_DCOUNT)

アドレスオフセット : 0x30

リセット値 : 0x0000 0000

SDIO_DCOUNT レジスタは、DPSM がアイドル状態から Wait_R または Wait_S 状態へ移行すると、データ長レジスタから値をロードします (SDIO_DLEN を参照してください)。データが転送される時、カウンタは 0 に達するまで値をデクリメントします。その後、DPSM はアイドル状態へ移行し、データステータス終了フラグ DATAEND がセットされます。

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
予約済み							DATACOUNT																								
							r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

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 24:0 **DATACOUNT** : データカウント値

このビットを読み出すと、残りの転送バイト数が返されます。書き込みは何の効果もありません。

注 : このレジスタは、データ転送が完了したときだけ読み出してください。

29.9.11 SDIO ステータスレジスタ (SDIO_STA)

アドレスオフセット : 0x34

リセット値 : 0x0000 0000

SDIO_STA レジスタは、読み出し専用レジスタです。2 種類のフラグを含みます。

- **スタティックフラグ** (ビット [23:22,10:0]) : これらのビットは、SDIO 割り込みクリアレジスタ (SDIO_ICR を参照) に書きこむことによってクリアされるまで、アサートされたままです。
- **ダイナミックフラグ** (ビット [21:11]) : これらのビットは、基礎回路の状態に応じて、状態が変わります (たとえば、FIFO フルおよびエンプティフラグは、データが FIFO に書き込まれることでアサートおよびデアサートされます)。

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
予約済み								CEATAEND	SDIOIT	RXDAVL	TXDAVL	RXFIFOE	TXFIFOE	RXFIFOF	TXFIFOF	RXFIFOHF	TXFIFOHE	RXACT	TXACT	CMDACT	DBCKEND	STBITERR	DATAEND	CMDSENT	CMDREND	RXOVERR	TXUNDERR	DTIMEOUT	CTIMEOUT	DCRCFAIL	CCRCFAIL
								r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

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 **CEATAEND** : CMD61 に対して CE-ATA コマンド完了信号が受信されました。

ビット 22 **SDIOIT** : SDIO 割り込みが受信されました。

ビット 21 **RXDAVL** : 受信 FIFO にデータがあります。

ビット 20 **TXDAVL** : 送信 FIFO にデータがあります。

ビット 19 **RXFIFOE** : 受信 FIFO が空です。

ビット 18 **TXFIFOE** : 送信 FIFO が空です。

HW フロー制御が有効なとき、FIFO に 2 ワードが含まれると、TXFIFOE 信号が有効になります。

ビット 17 **RXFIFOF** : 受信 FIFO が満杯です。

HW フロー制御が有効なとき、FIFO があと 2 ワードで満杯になるときに、RXFIFOF 信号が有効になります。

ビット 16 **TXFIFOF** : 送信 FIFO が満杯です。

ビット 15 **RXFIFOHF** : 受信 FIFO がハーフフル状態です。FIFO には少なくとも 8 ワード入っています。

ビット 14 **TXFIFOHE** : 送信 FIFO がハーフエンプティ状態です。FIFO には少なくとも 8 ワード書き込み可能です。

ビット 13 **RXACT** : データ受信中です。

ビット 12 **TXACT** : データ送信中です。

ビット 11 **CMDACT** : コマンド転送中です。

ビット 10 **DBCKEND** : データブロック送受信 (CRC 確認成功)

ビット 9 **STBITERR** : ワイドバスモードで、すべてのデータ信号で開始ビットが検出されませんでした。

ビット 8 **DATAEND** : データ終わり (データカウンタ SDIDCOUNT がゼロです)

ビット 7 **CMDSENT** : コマンドが送信されました (レスポンス不要)。

ビット 6 **CMDREND** : コマンドレスポンスが受信されました (CRC 確認成功)。

ビット 5 **RXOVERR** : FIFO オーバーランエラーを受信しました。

ビット 4 **TXUNDERR** : 送信 FIFO アンダーランエラー

ビット 3 **DTIMEOUT** : データタイムアウト



ビット 2 **CTIMEOUT** : コマンドレスポンスタイムアウト

コマンドタイムアウト期間は、固定値の 64 SDIO_CLK クロック周期です。

ビット 1 **DCRCFAIL** : データブロック送受信 (CRC 確認失敗)

ビット 0 **CCRCFAIL** : コマンドレスポンス受信 (CRC 確認失敗)

29.9.12 SDIO 割り込みクリアレジスタ (SDIO_ICR)

アドレスオフセット : 0x38

リセット値 : 0x0000 0000

SDIO_ICR レジスタは、書き込み専用レジスタです。ビットに 1b を書き込むと、SDIO_STA ステータスレジスタの対応するビットがクリアされます。

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
予約済み								CEATAENDC	SDIOITC	予約済み											DBCKENDC	STBITERRC	DATAENDC	CMDSENTC	CMDREND	RXOVERRC	TXUNDERRC	DTIMEOUTC	CTIMEOUTC	DCRCFAILC	CCRCFAILC
								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 **CEATAENDC** : CEATAEND フラグクリアビット

CEATAEND 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 : CEATAEND はクリアされません。

1 : CEATAEND がクリアされます。

ビット 22 **SDIOITC** : SDIOIT フラグクリアビット

SDIOIT 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 : SDIOIT はクリアされません。

1 : SDIOIT がクリアされます。

ビット 21:11 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 **DBCKENDC** : DBCKEND フラグクリアビット

DBCKEND 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 : DBCKEND はクリアされません。

1 : DBCKEND がクリアされます。

ビット 9 **STBITERRC** : STBITERR フラグクリアビット

STBITERR 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 : STBITERR はクリアされません。

1 : STBITERR がクリアされます。

ビット 8 **DATAENDC** : DATAEND フラグクリアビット

DATAEND 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 : DATAEND はクリアされません。

1 : DATAEND がクリアされます。

ビット 7 **CMDSENTC** : CMDSENT フラグクリアビット

CMDSENT 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 : CMDSENT はクリアされません。

1 : CMDSENT がクリアされます。

ビット 6 **CMDREND** : CMDREND フラグクリアビット

CMDREND 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 : CMDREND はクリアされません。

1 : CMDREND がクリアされます。

ビット 5 **RXOVERR** : RXOVERR フラグクリアビット

RXOVERR 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 : RXOVERR はクリアされません。

1 : RXOVERR がクリアされます。

ビット 4 **TXUNDERR** : TXUNDERR フラグクリアビット

TXUNDERR 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 : TXUNDERR はクリアされません。

1 : TXUNDERR がクリアされます。

ビット 3 **DTIMEOUT** : DTIMEOUT フラグクリアビット

DTIMEOUT 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 : DTIMEOUT はクリアされません。

1 : DTIMEOUT がクリアされます。

ビット 2 **CTIMEOUT** : CTIMEOUT フラグクリアビット

CTIMEOUT 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 : CTIMEOUT はクリアされません。

1 : CTIMEOUT がクリアされます。

ビット 1 **DCRCFAIL** : DCRCFAIL フラグクリアビット

DCRCFAIL 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 : DCRCFAIL はクリアされません。

1 : DCRCFAIL がクリアされます。

ビット 0 **CCRCFAIL** : CCRCFAIL フラグクリアビット

CCRCFAIL 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 : CCRCFAIL はクリアされません。

1 : CCRCFAIL がクリアされます。

29.9.13 SDIO マスクレジスタ (SDIO_MASK)

アドレスオフセット : 0x3C

リセット値 : 0x0000 0000

割り込みマスクレジスタは、対応するビットを 1b にセットすることによって、割り込みリクエストを生成するステータスフラグを決めます。

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
予約済み								CEATAENDIE	SDIOITIE	RXDAVLIE	TXDAVLIE	RXFIFOEIE	TXFIFOEIE	RXFIFOEIE	TXFIFOEIE	RXFIFOEIE	TXFIFOEIE	RXACTIE	TXACTIE	CMDACTIE	DBCKENDIE	STBITERRIE	DATAENDIE	CMDSENTIE	CMDRENDIE	RXOVERRIE	TXUNDERRIE	DTIMEOUTIE	CTIMEOUTIE	DCRCFAILIE	CCRCFAILIE
								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	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 **CEATAENDIE** : CE-ATA コマンド完了信号受信割り込みイネーブル

CE-ATA コマンド完了信号を受信したときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : CE-ATA コマンド完了信号受信割り込みは無効です。

1 : CE-ATA コマンド完了信号受信割り込みは有効です。

ビット 22 **SDIOITIE** : SDIO モード割り込み受信割り込みイネーブル

SDIO モード割り込みを受信したときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : SDIO モード割り込み受信割り込みは無効です。

1 : SDIO モード割り込み受信割り込みは有効です。

ビット 21 **RXDAVLIE** : Rx FIFO データ有無割り込みイネーブル

Rx FIFO のデータの有無によって生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Rx FIFO のデータの有無による割り込みは無効です。

1 : Rx FIFO のデータの有無による割り込みは有効です。

ビット 20 **TXDAVLIE** : Tx FIFO データ有無割り込みイネーブル

Tx FIFO のデータの有無によって生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Tx FIFO のデータの有無による割り込みは無効です。

1 : Tx FIFO のデータの有無による割り込みは有効です。

ビット 19 **RXFIFOEIE** : Rx FIFO エンプティ割り込みイネーブル

Rx FIFO が空になったときの割り込みを有効/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 : Rx FIFO エンプティ割り込みは無効です。

1 : Rx FIFO エンプティ割り込みは有効です。

ビット 18 **TXFIFOEIE** : Tx FIFO エンプティ割り込みイネーブル

Tx FIFO が空になったときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Tx FIFO エンプティ割り込みは無効です。

1 : Tx FIFO エンプティ割り込みは有効です。

ビット 17 **RXFIFOFIE** : Rx FIFO フル割り込みイネーブル

Rx FIFO が満杯になったときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Rx FIFO フル割り込みは無効です。

1 : Rx FIFO フル割り込みは有効です。

ビット 16 **TXFIFOFIE** : Tx FIFO フル割り込みイネーブル

Tx FIFO が満杯になったときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Tx FIFO フル割り込みは無効です。

1 : Tx FIFO フル割り込みは有効です。

ビット 15 **RXFIFOHFIE** : Rx FIFO ハーフフル割り込みイネーブル

Rx FIFO がハーフフル状態になったときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Rx FIFO ハーフフル割り込みは無効です。

1 : Rx FIFO ハーフフル割り込みは有効です。

ビット 14 **TXFIFOHEIE** : Tx FIFO ハーフエンプティ割り込みイネーブル

Tx FIFO がハーフエンプティ状態になったときに生成される割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Tx FIFO ハーフエンプティ割り込みは無効です。

1 : Tx FIFO ハーフエンプティ割り込みは有効です。

ビット 13 **RXACTIE** : データ受信割り込みイネーブル

データ受信中の割り込みを有効/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 : データ受信割り込みは無効です。

1 : データ受信割り込みは有効です。

ビット 12 **TXACTIE** : データ送信割り込みイネーブル

データ送信中の割り込みを有効/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 : データ送信割り込みは無効です。

1 : データ送信割り込みは有効です。

ビット 11 **CMDACTIE** : コマンド動作中割り込みイネーブル

コマンド転送中の割り込みを有効/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 : コマンド動作中割り込みは無効です。

1 : コマンド動作中割り込みは有効です。

ビット 10 **DBCKENDIE** : データブロックエンド割り込みイネーブル

データブロックエンドによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : データブロックエンド割り込みは無効です。

1 : データブロックエンド割り込みは有効です。

ビット 9 **STBITERRIE** : 開始ビットエラー割り込みイネーブル

開始ビットエラーによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : 開始ビットエラー割り込みは無効です。

1 : 開始ビットエラー割り込みは有効です。

ビット 8 **DATAENDIE** : データエンド割り込みイネーブル

データエンドによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : データエンド割り込みは無効です。

1 : データエンド割り込みは有効です。

ビット 7 CMDSENTIE : コマンド送信割り込みイネーブル

コマンド送信による割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : コマンド送信割り込みは無効です。

1 : コマンド送信割り込みは有効です。

ビット 6 CMDRENDIE : コマンドレスポンス受信割り込みイネーブル

コマンドレスポンスの受信による割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : コマンドレスポンス受信割り込みは無効です。

1 : コマンドレスポンス受信割り込みは有効です。

ビット 5 RXOVERRIE : Rx FIFO オーバーランエラー割り込みイネーブル

Rx FIFO オーバーランエラーによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Rx FIFO オーバーランエラー割り込みは無効です。

1 : Rx FIFO オーバーランエラー割り込みは有効です。

ビット 4 TXUNDERRIE : Tx FIFO アンダーランエラー割り込みイネーブル

Tx FIFO アンダーランエラーによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : Tx FIFO アンダーランエラー割り込みは無効です。

1 : Tx FIFO アンダーランエラー割り込みは有効です。

ビット 3 DTIMEOUTIE : データタイムアウト割り込みイネーブル

データタイムアウトによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : データタイムアウト割り込みは無効です。

1 : データタイムアウト割り込みは有効です。

ビット 2 CTIMEOUTIE : コマンドタイムアウト割り込みイネーブル

コマンドタイムアウトによる割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : コマンドタイムアウト割り込みは無効です。

1 : コマンドタイムアウト割り込みは有効です。

ビット 1 DCRCFAILIE : データ CRC 失敗割り込みイネーブル

データ CRC 失敗による割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : データ CRC 失敗割り込みは無効です。

1 : データ CRC 失敗割り込みは有効です。

ビット 0 CCRCFAILIE : コマンド CRC 失敗割り込みイネーブル

コマンド CRC 失敗による割り込みを有効/無効にするために、ソフトウェアによってセット/クリアされます。

0 : コマンド CRC 失敗割り込みは無効です。

1 : コマンド CRC 失敗割り込みは有効です。

29.9.14 SDIO FIFO カウンタレジスタ (SDIO_FIFOCNT)

アドレスオフセット : 0x48

リセット値 : 0x0000 0000

SDIO_FIFOCNT レジスタは、FIFO に書き込まれるか、FIFO から読み出される残りのワード数を含みます。FIFO カウンタは、データ制御レジスタ (SDIO_DCTRL レジスタ) のデータ転送イネーブルビット DTEN がセットされていて、DPSM がアイドル状態のときに、データ長レジスタ (SDIO_DLEN を参照) から値をロードします。データ長がワード境界 (4 の倍数) に合っていない場合、残りの 1 から 3 バイトは 1 ワードとみなされます。

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				
予約済み								FIFOCOUNT																											
								r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

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 23:0 **FIFOCOUNT** : FIFO に書き込まれるか、FIFO から読み出される残りのワード数を含みます。

29.9.15 SDIO データ FIFO レジスタ (SDIO_FIFO)

アドレスオフセット : 0x80

リセット値 : 0x0000 0000

受信および送信 FIFO は、32ビット幅のレジスタとして読み書きできます。FIFO は、32 の連続したアドレスに 32 個のエントリを含みます。このため、CPU は複数のオペランドをロードおよび格納して、FIFO の読み書きを行うことができます。

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
FIFOData																															
rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW	rW

ビット 31:0 **FIFOData** : 受信および送信 FIFO データ

FIFO データは、アドレス SDIO ベース+ 0x080 から SDIO ベース+ 0xFC までの 32ビットワードの 32 個のエントリを占有します。

29.9.16 SDIO レジスタマップ

次の表に、SDIO レジスタの要約を示します。

表 193. SDIO レジスタマップ

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	SDIO_POWER	予約済み																												PWRCTRL																									
0x04	SDIO_CLKCR	予約済み																		HWFC_EN		NEGEDGE		WIDBUS		BYPASS		PWRSAV		CLKEN		CLKDIV																							
0x08	SDIO_ARG	CMDARG																																																					
0x0C	SDIO_CMD	予約済み																		CE-ATACMD		nIEN		ENCMDcompl		SDIOSuspend		CPSMEN		WAITPEND		WAITINT		WAITRESP		CMDINDEX																			
0x10	SDIO_RESPCMD	予約済み																												RESPCMD																									
0x14	SDIO_RESP1	CARDSTATUS1																																																					
0x18	SDIO_RESP2	CARDSTATUS2																																																					
0x1C	SDIO_RESP3	CARDSTATUS3																																																					
0x20	SDIO_RESP4	CARDSTATUS4																																																					
0x24	SDIO_DTIMER	DATETIME																																																					
0x28	SDIO_DLEN	予約済み						DATALENGTH																																															
0x2C	SDIO_DCTRL	予約済み																		SDIOEN		RWMOD		RWSTOP		RWSTART		DBLOCKSIZE				DMAEN		DTMODE		DTDIR		DTEN																	
0x30	SDIO_DCOUNT	予約済み						DATACOUNT																																															
0x34	SDIO_STA	予約済み						CEATAEND		SDIOIT		RXDAVL		TXDAVL		RXFIOE		TXFIOE		RXFIOF		TXFIOF		RXFIOHF		TXFIOHE		RXACT		TXACT		CMDACT		DBCKEND		STBITERR		DATAEND		CMDSENT		CMDREND		RXOVERR		TXUNDERR		DTIMEOUT		CTIMEOUT		DCRCFAIL		CCRCFAIL	
0x38	SDIO_ICR	予約済み						CEATAENDC		SDIOITC		予約済み																DBCKENDC		STBITERRC		DATAENDC		CMDSENTC		CMDREND		RXOVERR		TXUNDERR		DTIMEOUTC		CTIMEOUTC		DCRCFAILC		CCRCFAILC							
0x3C	SDIO_MASK	予約済み						CEATAENDIE		SDIOITIE		RXDAVLIE		TXDAVLIE		RXFIOEIE		TXFIOEIE		RXFIOFIE		TXFIOFIE		RXFIOHFIE		TXFIOHEIE		RXACTIE		TXACTIE		CMDACTIE		DBCKENDIE		STBITERRIE		DATAENDIE		CMDSENTIE		CMDRENDIE		RXOVERRIE		TXUNDERRIE		DTIMEOUTIE		CTIMEOUTIE		DCRCFAILIE		CCRCFAILIE	
0x48	SDIO_FIFOCNT	予約済み						FIFOCOUNT																																															
0x80	SDIO_FIFO	FIFOData																																																					

30 デバッグサポート (DBG)

30.1 概要

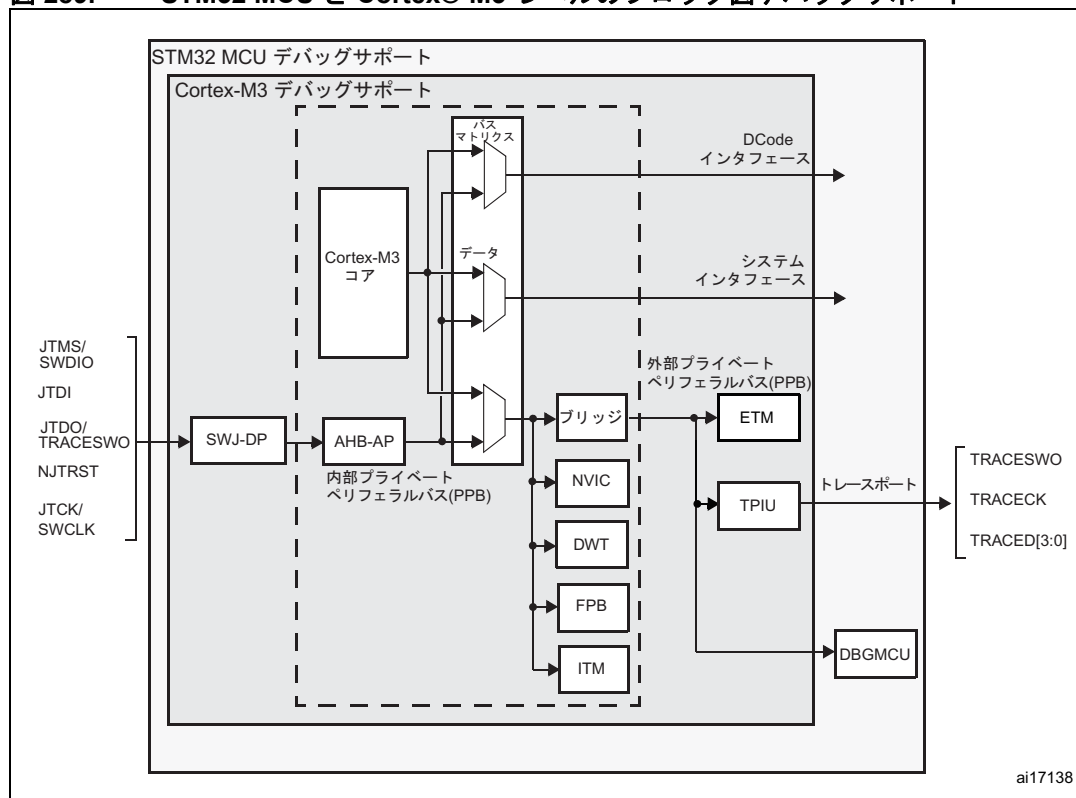
STM32L1xxxxは、Cortex®-M3 コアを中心にして構築されており、高度なデバッグ機能のためのハードウェア拡張機能を含んでいます。デバッグ拡張機能によって、コアは、特定の命令フェッチ時（ブレークポイント）またはデータアクセス時（ウォッチポイント）に停止することができます。停止したとき、コアの内部状態とシステムの外部状態を調査することができます。調査が終わったら、コアとシステムを復元して、プログラム実行を再開することができます。

デバッグ機能は、デバッガホストによって、STM32L1xxxx MCU への接続時とデバッグ時に使用されます。

デバッグ用に、次の 2 つのインターフェースを使用できます。

- シリアルワイヤ
- JTAG デバッグポート

図 289. STM32 MCU と Cortex®-M3-レベルのブロック図デバッグサポート



注： Cortex®-M3 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®-M3 コアは、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。以下の要素で構成されています。

- SWJ-DP : シリアルワイヤ/JTAG デバッグポート
- AHP-AP : AHB アクセスポート
- ITM : 計測トレースマクロセル (Instrumentation Trace Macrocell)
- FPB : フラッシュパッチブレークポイント (Flash Patch Breakpoint)
- DWT : データウォッチポイントトリガ (Data Watchpoint Trigger)
- TPIU : トレースポートインタフェースユニット (Trace Port Interface Unit : 対応するピンが配置される大型パッケージで使用)
- ETM : エンベデッドトレースマクロセル (Embedded Trace Macrocell : 対応するピンが配置される大型パッケージで使用)

STM32L1xxxx 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名割り当て
- 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®-M3 コアでサポートされるデバッグ機能の詳細については、Cortex®-M3-r2p0 Technical Reference Manual と CoreSight Design Kit-r2p0 TRM を参照してください(セクション 30.2: ARM® リファレンス資料を参照)。**

30.2 ARM® リファレンス資料

- Cortex®-M3 r2p0 Technical Reference Manual (TRM)
(1 ページの関連資料を参照してください。)
- ARM® Debug Interface V5
- ARM® CoreSight Design Kit revision r2p0 Technical Reference Manual

30.3 SWJ デバッグポート (シリアルワイヤと JTAG)

STM32L1xxxx コアには、シリアルワイヤ/JTAG デバッグポート (SWJ-DP) が組み込まれています。これは、JTAG-DP (5 ピン) インタフェースと SW-DP (2 ピン) インタフェースを組み合わせた ARM® 標準の CoreSight デバッグポートです。

- JTAG デバッグポート (JTAG-DP) は、AHP-AP ポートに 5 ピンの標準 JTAG インタフェースを提供します。
- シリアルワイヤデバッグポート (SW-DP) は、AHP-AP ポートに 2 ピン (クロック + データ) のインタフェースを提供します。

SWJ-DP では、SW-DP の 2 個の JTAG ピンは、JTAG-DP の 5 個の JTAG ピンの一部とマルチプレクスされています。

図 290. SWJ デバッグポート

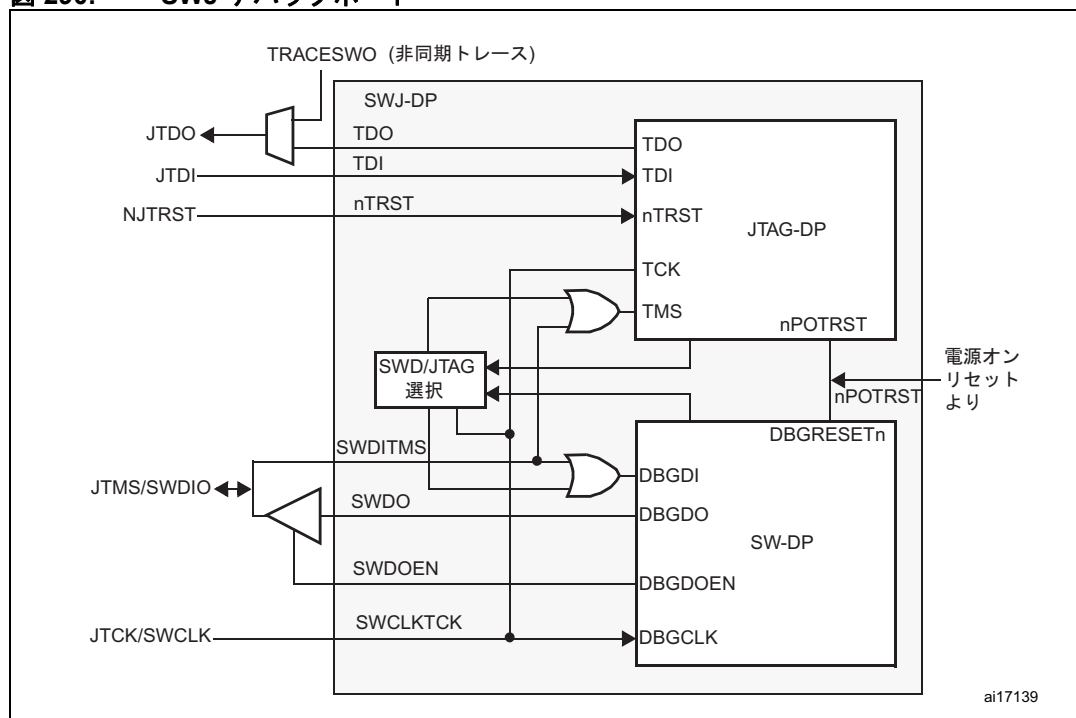


図 290 は、非同期 TRACE 出力 (TRACESWO) が TDO とマルチプレクスされていることを示します。これは、非同期トレースは、JTAG-DP ではなく、SW-DP でのみ使用できることを意味します。

30.3.1 JTAG-DP または SW-DP の選択メカニズム

デフォルトでは、JTAG デバッグポートが選択されています。

SW-DP に切り替えるには、デバッグホストは、TMS/TCK (それぞれ SWDIO/SWCLK に配置) に専用の JTAG シーケンスを提供して、JTAG-DP を無効にし SW-DP を有効にする必要があります。このように、SWCLK ピンと SWDIO ピンのみを使用して、SW-DP を有効にすることができます。

このシーケンスは、

1. TMS (SWDIO) =1 で 50 を超える TCK サイクルを送信します。
2. TMS (SWDIO) に 16ビットシーケンス (0111100111100111) を MSB ファーストで送信します。
3. TMS (SWDIO) =1 で 50 を超える TCK サイクルを送信します。

30.4 ピン名とデバッグポートピン

STM32L1xxxx MCU は、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。このため、ピンを使用する一部の機能 (ETM) は、パッケージによって異なることがあります。

30.4.1 SWJ デバッグポートピン

STM32L1xxxx からの SWJ-DP 出力として、汎用入出力のオルタネート機能として 5 本のピンが配置されています。これらのピンはすべてのパッケージで使用できます。

表 194. SWJ デバッグポートピン

SWJ-DP ピン名	JTAG デバッグポート		SW デバッグポート		ピン 割り当て
	タイプ	説明	タイプ	デバッグ割り当て	
JTMS/SWDIO	I	JTAG テストモード選択	IO	シリアルワイヤデータ入出力	PA13
JTCK/SWCLK	I	JTAG テストクロック	I	シリアルワイヤクロック	PA14
JTDI	I	JTAG テストデータ入力	-	-	PA15
JTDO/TRACESWO	O	JTAG テストデータ出力	-	非同期トレースが有効な場合は TRACESWO	PB3
NJTRST	I	JTAG テスト nReset	-	-	PB4

30.4.2 フレキシブルな SWJ-DP ピン配置

RESET (SYSRESETn または PORESETn) 後、SWJ-DP 用に使用される 5 本のピンはすべて、専用ピンとして割り当てられ、デバッグホストがすぐに使用できるようになります (なお、デバッグホストによって明示的にプログラミングされた場合を除いて、トレース出力は割り当てられません)。

しかしながら、STM32L1xxxx MCU では、SWJ-DP ポートの一部または全部を無効化することができ、関連するピンを開放し、汎用 IO (GPIO) として使用することができます。

表 195. フレキシブルな SWJ-DP ピン配置

使用可能なデバッグポート	SWJ IO ピンの割り当て				
	PA13 / JTMS / SWDIO	PA14 / JTCK / SWCLK	PA15 / JTDI	PB3 / JTDO	PB4 / NJTRST
全 SWJ (JTAG-DP + SW-DP)、リセット状態	X	X	X	X	X
全 SWJ (JTAG-DP + SW-DP)、NJTRST なし	X	X	X	X	
JTAG-DP 無効、SW-DP 有効	X	X			
JTAG-DP 無効、SW-DP 無効					解放

注: APB ブリッジの書き込みバッファが満杯のとき、GPIO_AFR レジスタへの書き込みには APB の 1 サイクル分が追加で必要になります。これは、コアの nTRST および TCK 入力信号でのクリーンレベルを保証するために、JTAGSW ピンの無効化が 2 サイクルで行われるからです。

- サイクル 1: コアへの JTAGSW 入力信号は 1 または 0 に (nTRST、TDI、および TMS では 1 に、TCK では 0 に) 接続されます。
- サイクル 2: GPIO コントローラが SWJTAG 入出力ピンの制御 (方向、プルアップ/ダウン、シュミットトリガの有効化などの制御) 信号を受け取ります。

30.4.3 JTAG ピンでの内部プルアップ/プルダウン

JTAG 入力ピンは、デバッグモード機能を制御するためにフリップフロップに直結されます。したがって、JTAG 入力ピンをフロート状態にしないことが必要です。これらのフリップフロップの一部のクロックに直結される SWCLK/TCK ピンについては、特に注意が必要です。

入出力レベルを正しく制御するため、デバイスには内部プルアップ/プルダウンが JTAG 入力ピンに内蔵されています。

- NJTRST : 内部プルアップ
- JTDI : 内部プルアップ
- JTMS/SWDIO : 内部プルアップ
- TCK/SWCLK : 内部プルダウン

JTAG 入出力がユーザソフトウェアによって解放されると、GPIO コントローラが再び制御権を獲得します。リセット状態では、GPIO 制御レジスタは入出力をこれと同等の状態に設定します。

- NJTRST : AF 入力プルアップ
- JTDI : AF 入力プルアップ
- JTMS/SWDIO : AF 入力プルアップ
- JTCK/SWCLK : AF 入力プルダウン
- JTDO : AF 入力フローティング

ソフトウェアはこれらの入出力を標準の GPIO 信号として使用することができます。

注 : *JTAG IEEE 規格では、TDI、TMS、および nTRST にプルアップを追加することを推奨していますが、TCK に関しては特別な推奨はありません。ただし、JTCK に関しては、デバイスは内蔵プルダウンを必要とします。*

プルアップとプルダウンを内蔵しているため、外部抵抗を追加する必要はありません。

30.4.4 シリアルワイヤの使用と、未使用のデバッグピンを GPIO として解放する方法

シリアルワイヤ DP を使用していくつかの GPIO を開放するには、ユーザソフトウェアによって、GPIO_MODER レジスタの GPIO(PA15, PB3 and PB4)設定モードを変更する必要があります。これによって PA15、PB3、および PB4 が解放され、GPIO として使用できるようになります。

デバッグ時には、ホストは次の動作を行います。

- システムリセット中に、すべての SWJ ピンが割り当てられます (JTAG-DP + SW-DP)。
- システムリセット中に、デバッグホストは、JTAG-DP から SW-DP に切り替える JTAG シーケンスを送信します。
- システムリセット中に、さらに、デバッグはベクタリセットにブレークポイントを設定します。
- システムリセットは解除され、コアは停止します。
- これ以降のすべてのデバッグ通信は、SW-DP を使用して行われます。残りの JTAG ピンは、ユーザソフトウェアによって GPIO として再割り当てできます。

注： ユーザソフトウェアの設計に関しては、次の点に注意してください。

デバッグピンを解放するには、リセット後にユーザソフトウェアがピンを解放するまでの一定の期間、デバッグピンは、まず、入力プルアップ (nTRST、TMS、TDI)、プルダウン (TCK)、または出カトライステート (TDO) に設定されることに注意してください。

デバッグピン (JTAG、SW、または TRACE) が配置されると、IOPORT コントローラの対応する入出力ピンの設定を変更しても効果はありません。

30.5 STM32L1xxxx JTAG TAP 接続

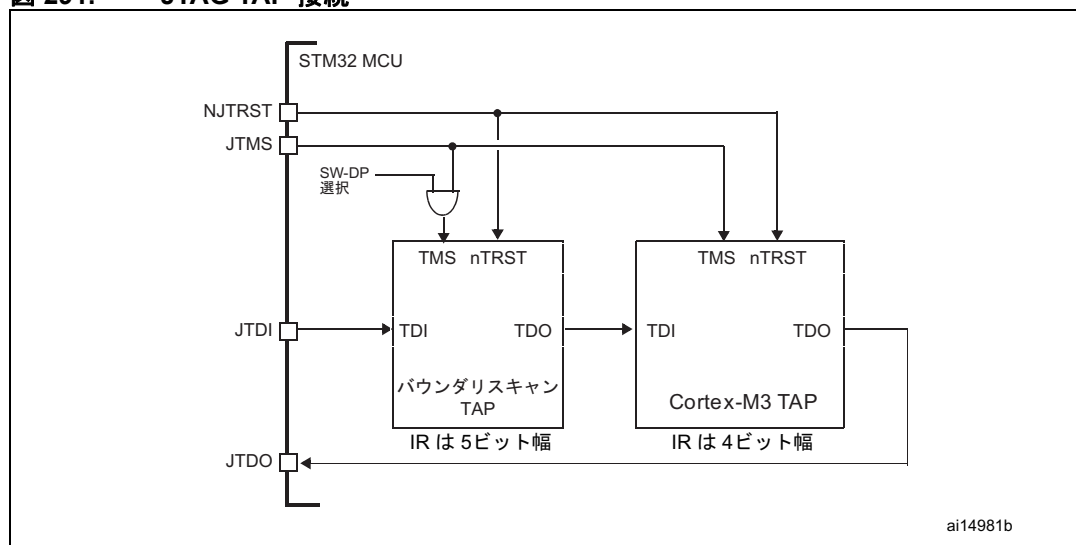
STM32L1xxxx MCU は、バウンダリスキャン TAP (IR は 5ビット幅) と Cortex®-M3 TAP (IR は 4ビット幅) という、シリアル接続された 2 個の JTAG TAP を内蔵しています。

デバッグ目的で Cortex®-M3 の TAP にアクセスするには、

1. まず、バウンダリスキャン TAP の BYPASS 命令をシフトする必要があります。
2. 次に、IR シフトごとに、スキャンチェーンには 9 (=5+4) ビットが格納され、未使用の TAP 命令を BYPASS 命令によってシフトインする必要があります。
3. データシフトごとに、BYPASS モードにある未使用の TAP は、データスキャンチェーンに 1 個のデータビットを追加します。

注： **重要：** 専用の ARM® JTAG シーケンスを使用してシリアルワイヤが選択されると、バウンダリスキャン TAP は自動的に無効になります (JTMS は強制的にハイレベルになります)。

図 291. JTAG TAP 接続



30.6 ID コードとロック機構

STM32L1xxxx MCU には、内部にいくつかの ID コードがあります。ツール設計者は、外部 PPB メモリマップのアドレス 0xE0042000 に配置されている MCU デバイス ID コードを使用して、デバッグをロックすることを強く推奨します。

30.6.1 MCU デバイス ID コード

STM32L1xxxx MCU には MCU ID コードが内蔵されています。この ID は、ST MCU の部品番号とダイのリビジョンを識別します。これは DBG_MCU 部品の一部であり、外部 PPB バスに配置されます (セクション 30.16 (904 ページ) を参照)。このコードにアクセスするには、JTAG デバッグポート (4~5 本のピン)、SW デバッグポート (2 本のピン)、またはユーザソフトウェアを使用します。アクセスは MCU がシステムリセット中でも可能です。

DBGMCU_IDCODE

アドレス : 0xE004 2000

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
予約済み				DEV_ID											
				r	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。

値	Cat.1 デバイス	Cat.2 デバイス	Cat.3 デバイス	Cat.4 / Cat.3 デバイス ⁽¹⁾	Cat.5 デバイス
0x1000 :	Rev A	Rev A		Rev A	Rev A
0x1008 :	Rev Y			Rev Z	
0x1018		Rev Z	Rev A	Rev Y	
0x1038 :	Rev W		Rev X		
0x1078 :	Rev V				

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 11:0 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。

0x416 : Cat.1 デバイス

0x429 : Cat.2 デバイス

0x427 : Cat.3 デバイス

0x436 : Cat.4 デバイス または Cat.3 デバイス⁽¹⁾

0x437 : Cat.5 デバイス

1. Cat.3 デバイス : WLCSP64 パッケージまたは 100 ピン以上のパッケージに封入された、文字“A”の付いた RPN エンディングの STM32L15xxC または STM32L16xxC デバイス。

30.6.2 バウンダリスキャン TAP

JTAG ID コード

STM32L1xxxx BSC (バウンダリスキャン) の TAP は、0x4BA00477 に等しい JTAG ID コードを内蔵しています。

30.6.3 Cortex®-M3 TAP

ARM® Cortex®-M3 の TAP は、JTAG ID コードを内蔵しています。この ID コードは ARM® のデフォルトであり、変更されていません。このコードは、JTAG デバッグポートからのみアクセスできます。このコードは、0x4BA00477 です (Cortex®-M3 r2p0 に対応します。[セクション 30.2 : ARM® リファレンス資料](#)を参照)。

デバッグ/プログラマツールでは、DEV_ID(11:0) のみを識別に使用してください。

30.6.4 Cortex®-M3 JEDEC-106 ID コード

ARM® Cortex®-M3 は JEDEC-106 ID コードを内蔵しています。これは内部 PPB バスのアドレス 0xE00FF000_0xE00FFFFF に配置された 4KB ROM テーブルに置かれています。

このコードは、JTAG デバッグポート (4~5 本のピン)、SW 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。

30.7 JTAG デバッグポート

標準的な JTAG ステートマシンは、4ビット命令レジスタ (IR) と 5 個のデータレジスタを搭載しています (詳細については、Cortex®-M3 r2p0 *Technical Reference Manual (TRM)* と [セクション 30.2 : ARM® リファレンス資料](#) を参照)。

表 196. JTAG デバッグポートのデータレジスタ

IR(3:0)	データレジスタ	詳細
1111	BYPASS [1ビット]	
1110	IDCODE [32ビット]	ID コード 0x4BA00477 (ARM® Cortex®-M3 r2p0 ID コード)
1010	DPACC [35ビット]	デバッグポートアクセスレジスタ デバッグポートを初期化し、デバッグポートレジスタへのアクセスを可能にします。 - 入力データ転送時 : ビット 34:3 = DATA[31:0] = 書き込みリクエスト用に転送する 32ビットデータ ビット 2:1 = A[3:2] = デバッグポートレジスタの 2ビットアドレス ビット 0 = RnW = 読み出しリクエスト (1) または書き込みリクエスト (0) - 出力データ転送時 : ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32ビットデータ ビット 2:0 = ACK[2:0] = 3ビット確認応答 : 010 = OK/FAULT 001 = WAIT その他 = 予約済み A(3:2) ビットの説明については、表 197 を参照してください。
1011	APACC [35ビット]	アクセスポートアクセスレジスタ アクセスポートを初期化し、アクセスポートレジスタへのアクセスを可能にします。 - 入力データ転送時 : ビット 34:3 = DATA[31:0] = 書き込みリクエスト用にシフトインする 32ビットデータ ビット 2:1 = A[3:2] = 2ビットアドレス (サブアドレス AP レジスタ) ビット 0 = RnW = 読み出しリクエスト (1) または書き込みリクエスト (0) - 出力データ転送時 : ビット 34:3 = DATA[31:0] = 読み出しリクエストに続いて読み出される 32ビットデータ ビット 2:0 = ACK[2:0] = 3ビット確認応答 : 010 = OK/FAULT 001 = WAIT その他 = 予約済み 次の項目の組み合わせとして、多くの AP レジスタ (AHB-AP を参照) 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
1000	ABORT [35ビット]	アボートレジスタ - ビット 31:1 = 予約済み - ビット 0 = DAPABORT : DAP アボートを生成するには 1 を書き込みます

表 197. 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。 – システムまたはデバッグのパワーアップリクエスト – AP アクセス用の転送動作設定 – プッシュアウト比較およびプッシュアウト検証動作の制御 – 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 上で有効な 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：一連の動作の後で（新しい JTAG-DP 動作をリクエストせずに）、デバッグが最終結果を得られるようにします。

30.8 SW デバッグポート

30.8.1 SW プロトコルの概要

この同期式シリアルプロトコルでは、2 本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 です）。

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。

30.8.2 SW プロトコルシーケンス

各シーケンスは 3 つのフェーズから構成されます。

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 198. 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 197 を参照)
5	パリティ	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®-M3r2p0 TRM を参照してください。

パケットリクエストの後には、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 199. ACK 応答 (3ビット)

ビット	名称	説明
0..2	ACK	001 : FAULT 010 : WAIT 100 : OK

READ トランザクションである場合や、WAIT または FAULT 確認応答が受信された場合にのみ、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 200. DATA 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	パリティ	32 データビットの 1 ビットパリティ

READ トランザクションの場合にのみ、DATA 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。

30.8.3 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 規格に準拠しています。この ID コードは、ARM® のデフォルトであり、0x4BA00477 (Cortex®-M3 r2p0 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 DP が JTAG から SWD に切り替えられた後、またはラインが 50 サイクルを超えてハイレベルにあった後では、SW-DP ステートマシンは RESET 状態になります。
- RESET 状態の後、ラインが 2 サイクル以上の間ローレベルである場合、SW-DP ステートマシンは IDLE 状態になります。
- RESET 状態の後、まず IDLE 状態に入り、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**。さもなければ、ターゲットは、他のトランザクションに対して FAULT 確認応答を発行します。

SW-DP ステートマシンの詳細については、Cortex®-M3 r2p0 TRM および CoreSight Design Kit r2p0 TRM を参照してください。

30.8.4 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。ターゲットは、ACK=OK の場合はただちに応答し、ACK=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/AP 書き込み用の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満杯のとき、ターゲットの確認応答は"WAIT"です。ただし、IDCODE 読み出し、CTRL/STAT 読み出し、および ABORT 書き込みは、書き込みバッファが満杯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 (IDLE 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。

30.8.5 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 201. SW-DP レジスタ

A(3:2)	R/W	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 ST 社のコードではありません。 0x4BA00477 (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、 – 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 – ステータスフラグの読み出し (オーバーラン、 パワーアップ確認応答)
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)
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		READ BUFFER	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得られま す)。 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。

30.8.6 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として、多くの AP レジスタ (AHB-AP を参照) 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

30.9 AHB-AP (AHB アクセスポート) - JTAG-DP と SW-DP の両方に有効

機能 :

- システムアクセスはプロセッサステータスと独立
- SW-DP または JTAG-DP のどちらかが AHB-AP にアクセスします。
- AHB-AP はバスマトリックスの AHB マスタです。したがって、AHB-AP は I-bus を除くすべてのデータバス (D-bus、S-Bus、内部/外部の PPB バス) にアクセスできます。
- ビットバンドトランザクションがサポートされます。
- AHB-AP トランザクションは FPB を迂回します。

32 ビット AHB-AP レジスタは 6 ビット幅 (最大 64 ワードまたは 256 バイト) であり、以下の構成となっています。

- f) ビット [7:4] = DP_SELECT レジスタのビット [7:4] APBANKSEL
- g) ビット [3:2] = SW-DP 用の 35 ビットパケットリクエストの 2 ビットアドレス A(3:2)

Cortex®-M3 の AHB-AP は、9 個の 32ビットレジスタを内蔵しています。

表 202. Cortex®-M3 AHB-AP レジスタ

アドレス オフセット	レジスタ名	注
0x00	AHB-AP 制御/ステータスワード	AHB インタフェースを通じて転送を設定および制御します (サイズ、hprot、現在の転送のステータス、アドレスインクリメントタイプ)。
0x04	AHB-AP 転送アドレス	-
0x0C	AHB-AP データ読み出し/書き込み	-
0x10	AHB-AP バンクデータ 0	転送アドレスレジスタを書き換えずに、4 個のアラインドデータワードを直接配置します。
0x14	AHB-AP バンクデータ 1	
0x18	AHB-AP バンクデータ 2	
0x1C	AHB-AP バンクデータ 3	
0xF8	AHB-AP デバッグ ROM アドレス	デバッグインタフェースのベースアドレス
0xFC	AHB-AP ID レジスタ	-

詳細については、Cortex®-M3 r2p0 TRM を参照してください。

30.10 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、*Advanced High-performance Bus* (AHB-AP) ポートを使用します。プロセッサは、内部の専用ペリフェラルバス (PPB) を介してこれらのレジスタに直接アクセスできます。

コアデバッグは 4 個のレジスタから構成されています。

表 203. 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 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と書き込みを行うための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。このレジスタには、TRACE を使用できるようにする TRCENA というビットがあります。

注： **重要：**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®-M3 r2p0TRM** を参照してください。

リセット時に停止させるには、

- 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。

30.11 システムリセット中のデバッグホスト接続機能

STM32L1xxx MCU のリセットシステムは、次のリセットソースから構成されます。

- POR (パワーオンリセット) : パワーアップのたびに RESET をアサートします。
- 内部ウォッチドッグリセット
- ソフトウェアリセット
- 外部リセット

Cortex®-M3 では、デバッグ部のリセット (一般に PORRESETn) とそれ以外のリセット (SYSRESETn) が区別されます。

こうすれば、リセットベクタをフェッチするときにコアを停止させるようコアデバッグレジスタをプログラミングすることで、デバッグはシステムリセット中に接続が可能になります。これによってホストはシステムリセットを解除でき、コアは命令を実行しなくても、ただちに停止します。さらに、システムリセット中にデバッグ機能をプログラミングできます。

注 : システムリセット中にデバッグホストを接続する (リセットベクタにブレークポイントを設定する) ことを強く推奨します。

30.12 FPB (フラッシュパッチブレークポイント)

FPB ユニットは、

- ハードウェアブレークポイントを実装します。
- コード空間からシステム空間に、コードとデータをパッチします。この機能によって、コードメモリ空間にあるソフトウェアバグの訂正が可能になることがあります。

ソフトウェアパッチとハードウェアブレークポイントは同時に使用できません。

FPB は以下から構成されます。

- コード空間からのリテラルロードと照合し、システム空間内の対応する領域に再配置する 2 個のリテラルコンパレータ。
- コード空間からの命令フェッチと照合する 6 個の命令コンパレータ。命令コンパレータを使用すると、システム空間内の対応する領域に再配置したり、コアへのブレークポイント命令を生成したりできます。

30.13 DWT (データウォッチポイントトリガ)

DWT ユニットは 4 個のコンパレータから構成されています。これらは、次の用途に設定できます。

- ハードウェアウォッチポイント
- ETM へのトリガ
- PC サンプラ
- データアドレスサンプラ

DWT は、プロファイル情報を提供することもできます。このため、次の数を得るためのカウンタにアクセスできます。

- クロックサイクル数
- フォールドされた命令数
- ロードストアユニット (LSU) の動作数
- スリープサイクル数
- CPI (命令あたりのクロック数)
- 割り込みオーバーヘッドの回数

30.14 ITM (計測トレースマクロセル)

30.14.1 概要

ITMは、アプリケーション駆動のトレースソースであり、オペレーティングシステム (OS) やアプリケーションのイベントをトレースする *printf* 方式のデバッグをサポートし、システムの診断情報を発行します。ITM がパケットとして発行するトレース情報は、次のように生成できます。

- **ソフトウェアトレース**：ソフトウェアは、ITM スティムラスレジスタに直接書き込みを行って、パケットを発行できます。
- **ハードウェアトレース**：DWT がパケットを生成し、ITM がそれを発行します。
- **タイムスタンプ**：タイムスタンプはパケットを基準にして発行されます。ITM は、タイムスタンプを生成するための 21 ビットカウンタを内蔵しています。カウンタは、Cortex®-M3 クロックまたはシリアルワイヤ ビューア (SWV) 出力のビットクロックレートによって駆動されます。

ITM によって発行されたパケットは、TPIU (トレースポートインタフェースユニット) に出力されます。TPIU のフォーマットは、いくつかのパケットを追加してから (TPIU を参照)、完全なパケットシーケンスをデバッガホストに出力します。

ITM をプログラミングまたは使用するには、その前に、デバッグ例外/モニタ制御レジスタの TRCEN ビットを有効にする必要があります。

STOP モードデバッグ (DBG_STOP ビットをセット) 中、SysTick タイマクロックは停止しません。このカウンタはデクリメントを継続し、割り込みが有効な場合はタイマ割り込みを生成できます。

30.14.2 タイムスタンプパケット、同期およびオーバーフローパケット

タイムスタンプパケットは、タイムスタンプ情報や一般的な制御および同期情報をコード化します。そのために使用される 21 ビットのタイムスタンプカウンタ（プリスケラ付きの場合もあります）は、タイムスタンプパケットが発行されるたびにリセットされます。このカウンタは、CPU クロックまたは SWV クロックによって駆動できます。

同期パケットは 6 バイトの値 0x80_00_00_00_00_00 から構成され、TPIU には 00 00 00 00 00 80 として LSB ファーストで発行されます。

同期パケットは、タイムスタンプ制御のためのパケットです。このパケットは、DWT トリガのたびに発行されます。

このため、DWT 制御レジスタの CYCCNTENA（ビット 0）をセットして、ITM をトリガするように DWT を設定する必要があります。さらに、ITM トレース制御レジスタのビット 2（SYNCENA）もセットする必要があります。

注： **SYNENA ビットがセットされていない場合、DWT は TPIU への同期トリガを生成し、TPIU 同期パケットのみが送信され、ITM 同期パケットは送信されません。**

オーバーフローパケットは、データの書き込み時に FIFO が満杯であったことを示す特殊なタイムスタンプパケットで構成されます。

表 204. 主な ITM レジスタ

アドレス	レジスタ	詳細
@E0000FB0	ITM ロックアクセス	他の ITM レジスタへの書き込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
@E0000E80	ITM トレース制御	ビット 31-24 = 常に 0
		ビット 23 = ビジー
		ビット 22-16 = トレースデータのソースを識別する 7 ビットの ATB ID
		ビット 15-10 = 常に 0
		ビット 9:8 = TSPrescale = タイムスタンププリスケラ
		ビット 7-5 = 予約済み
		ビット 4 = SWOENA = SWV 動作（タイムスタンプカウンタを SWV クロックによって駆動）を有効にします。
		ビット 3 = DWTENA : DWT スティムラスを有効にします。
		ビット 2 = SYNCENA : DWT が同期トリガを生成して TPIU が同期パケットを発行できるようにするには、このビットを 1 にする必要があります。
		ビット 1 = TSENA（タイムスタンプイネーブル）
		ビット 0 = ITMENA : ITM のグローバルイネーブルビット
@E0000E40	ITM トレース特権	ビット 3 : ポート 31:24 のトレースを有効にするマスク
		ビット 2 : ポート 23:16 のトレースを有効にするマスク
		ビット 1 : ポート 15:8 のトレースを有効にするマスク
		ビット 0 : ポート 7:0 のトレースを有効にするマスク

表 204. 主な ITM レジスタ (続き)

アドレス	レジスタ	詳細
@E0000E00	ITM トレースイネーブル	各ビットは、トレースを生成するのに対応するスティムラポートを有効にします。
@E0000000-E000007C	スティムラポートレジスタ 0-31	選択されたスティムラポート (32 個まで) に 32 ビットデータを書き込んでトレース出力します。

設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、DBGMCU_CR を設定することによって TRACE I/O を割り当てます ([セクション 30.17.2: TRACE ピンの割り当て](#)と [セクション 30.16.3: デバッグ MCU 設定レジスタ](#)を参照)。
- ITM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ITM レジスタへの書き込みアクセスをロック解除します。
- ITM トレース制御レジスタに 0x00010005 を書き込んで、同期を有効にした状態で ITM を有効にし、ATB ID を 0x00 以外の値にします。
- ITM トレースイネーブルレジスタに 0x1 を書き込んで、スティムラポート 0 を有効にします。
- ITM トレース特権レジスタに 0x1 を書き込んで、スティムラポート 7:0 をマスク解除します。
- 出力する値をスティムラポートレジスタ 0 に書き込みます。これにはソフトウェア (printf 関数) を使用します。

30.15 ETM (組み込みトレースマクロセル)

30.15.1 概要

ETM はプログラム実行の再構築を有効にします。データはデータウォッチポイントおよびトレース (DWT) コンポーネントまたは計測トレースマクロセル (ITM) を使用してトレースされます。一方、命令は組み込みトレースマクロセル (ETM) を使用してトレースされます。

ETMは、情報をパケットとして送信し、埋め込みリソースによってトリガされます。これらのリソースは個別にプログラミングする必要があり、トリガソースの選択にはトリガイベントレジスタ (0xE0041008) を使用します。イベントは、単一イベント (アドレスコンパレータからのアドレスマッチ) または 2 つのイベント間の論理式とすることができます。トリガソースは、DWT モジュールの 4 番目のコンパレータの 1 つです。次のイベントを監視できます。

- クロックサイクルマッチング
- データアドレスマッチング

トリガリソースの詳細については、[セクション 30.13: DWT \(データウォッチポイントトリガ\)](#)を参照してください。

ETM によって送信されたパケットは、TPIU (トレースポートインタフェースユニット) に出力されます。TPIU のフォーマットは、いくつかのパケットを追加してから ([セクション 30.17: TPIU \(トレースポートインタフェースユニット\)](#)を参照)、完全なパケットシーケンスをデバッグホストに出力します。

30.15.2 信号プロトコル、パケットタイプ

これについては、ARM® IHI 0014N ドキュメントの第 7 章「ETMv3 信号プロトコル」を参照してください。

30.15.3 主な ETM レジスタ

レジスタの詳細については、ARM® IHI 0014N 仕様の第 3 章を参照してください。

表 205. 主な ETM レジスタ

アドレス	レジスタ	詳細
0xE0041FB0	ETM ロックアクセス	他の ETM レジスタへの書き込みアクセスをロック解除するには、0xC5ACCE55 を書き込みます。
0xE0041000	ETM 制御	このレジスタは、ETM の一般的な動作（たとえば、トレースを有効にする方法）を制御します。
0xE0041010	ETM ステータス	このレジスタは、トレースおよびトリガロジックの現在のステータスに関する情報を提供します。
0xE0041008	ETM トリガイイベント	このレジスタは、トリガを制御するイベントを定義します。
0xE004101C	ETM トレースイネーブル制御	このレジスタは、どのコンパレータが選択されるかを定義します。
0xE0041020	ETM トレースイネーブルイベント	このレジスタは、トレース有効化イベントを定義します。
0xE0041024	ETM トレース開始/停止	このレジスタは、トレースをそれぞれ開始および停止するためにトリガソースによって使用されるトレースを定義します。

30.15.4 設定例

TPIU に単純な値を出力するには、

- TPIU を設定し、I/O_TRACEN を有効にして、のデバッグ設定レジスタに TRACE IO を割り当てます。
- ETM ロックアクセスレジスタに 0xC5ACCE55 を書き込んで、ETM レジスタへの書き込みアクセスをロック解除します。
- 制御レジスタに 0x00001D1E を書き込みます（トレースを設定します）。
- トリガイイベントレジスタに 0000406F を書き込みます（トリガイイベントを定義します）。
- トレースイネーブルイベントレジスタに 0000006F を書き込みます（開始/停止するイベントを定義します）。
- トレース開始/停止レジスタに 00000001 を書き込みます（トレースを有効にします）。
- ETM 制御レジスタに 0000191E を書き込みます（設定の終了）。

30.16 MCU デバッグコンポーネント (DBGMCU)

MCU デバッグコンポーネントは、デバッグによる次のサポート機能を支援します。

- 低電力モード
- タイマ、ウォッチドッグ (WWDG および IWDG)、および I2C のクロック制御
- TRACE ピンの割り当て制御

30.16.1 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、FCLK や HCLK をオフにすることはできません。これらは、デバッグ時のデバッグ接続に必要なので、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 モードでは、DBGMCU_CR レジスタの DBG_SLEEP 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、HCLK には FCLK と同じクロックが供給されます (システムクロックはソフトウェアによって事前に設定されています)。
- STOP モードでは、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 オシレータが、STOP モードで FCLK と HCLK にクロックを供給できます。

30.16.2 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、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。これは、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。これ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。

30.16.3 デバッグ MCU 設定レジスタ

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 : SLEEP モード、STOP モード、STANDBY モード
- TRACE ピンの割り当て

この DBGMCU_CR は、アドレス 0xE0042004 にある外部 PPB バスに配置されます。

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ありません)。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。

DBGMCU_CR レジスタ

アドレス : 0xE004 2004

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	
予約済み																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
予約済み								TRACE_ MODE[1:0]		TRACE_ IOEN		予約済み		DBG_ STANDBY	DBG_ STOP	DBG_ SLEEP
								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 7:5 **TRACE_MODE[1:0] および TRACE_IOEN** : TRACE ピンの割り当て制御

– TRACE_IOEN=0 の場合 :

TRACE_MODE=xx : TRACE ピンは割り当てられません (デフォルト状態)。

– TRACE_IOEN=1 の場合 :

- TRACE_MODE=00 : 非同期モード用の TRACE ピン割り当て
- TRACE_MODE=01 : 同期モード用の TRACE ピン割り当て (TRACEDATA サイズ = 1)
- TRACE_MODE=10 : 同期モード用の TRACE ピン割り当て (TRACEDATA サイズ = 2)
- TRACE_MODE=11 : 同期モード用の TRACE ピン割り当て (TRACEDATA サイズ = 4)

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 DBG_STANDBY : デバッグ STANDBY モード

0 : (FCLK=Off, HCLK=Off) 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します)。

1 : (FCLK=On, HCLK=On) この場合、デジタル部は電源オフ状態ではなく、FCLK と HCLK は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ベクタ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=Off, HCLK=Off) 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場合と同じになります (CPU は 16 MHz の内部 RC オシレータ (HSI) から供給されます)。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=On, HCLK=On) 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の内部 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 : (FCLK=On, HCLK=Off) 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。

SLEEP モードでは、クロックコントローラの設定はリセットされず、事前にプログラミング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 : (FCLK=On, HCLK=On) 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。

30.16.4 デバッグ MCU APB1 凍結レジスタ (DBGMCU_APB1_FZ)

DBGMCU_APB1_FZ レジスタは、DEBUG 中、MCU の設定に使用されます。これは、次の APB1 ペリフェラルに影響します。

- 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
- ウィンドウ型ウォッチドッグカウンタおよび独立型ウォッチドッグカウンタの凍結サポート

この DBGMCU_APB1_FZ は、アドレス 0xE0042008 にある外部 PPB バスに配置されます。

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ありません）。システムリセット中に、デバッガによって書き込みが可能です。

アドレス：0xE004 2008

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
予約済み									DBG_I2C2_SMBUS_TIMEOUT	DBG_I2C1_SMBUS_TIMEOUT	予約済み				
									rw	rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み			DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_WWDG_STOP	予約済み				DBG_TIM7_STOP	DBG_TIM6_STOP	予約済み	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
			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 **DBG_I2C2_SMBUS_TIMEOUT**：コア停止時に 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：SMBUS タイムアウトは凍結されます。

ビット 21 **DBG_I2C1_SMBUS_TIMEOUT**：コア停止時に 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：SMBUS タイムアウトは凍結されます。

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：コアが停止した場合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：コアが停止した場合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 : コアが停止した場合も RTC カウンタのクロックは供給されます。
1 : コア停止時に RTC カウンタのクロックは停止します。
注 : このビットは、Cat.2, Cat.3, Cat.4 および Cat.5 デバイスでのみ使用できます。
- ビット 9:6 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 **DBG_TIM7_STOP** : コア停止時に TIM7 カウンタは停止
0 : コアが停止した場合も TIM7 のカウンタクロックは供給されます。
1 : コア停止時に TIM7 のカウンタクロックは停止します。
- ビット 4 **DBG_TIM6_STOP** : コア停止時に TIM6 カウンタは停止
0 : コアが停止した場合も TIM6 のカウンタクロックは供給されます。
1 : コア停止時に TIM6 のカウンタクロックは停止します。
- ビット 3 **DBG_TIM5_STOP** : コア停止時に TIM5 カウンタは停止
0 : コアが停止した場合も TIM5 のカウンタクロックは供給されます。
1 : コア停止時に TIM5 のカウンタクロックは停止します。
注 : このビットは、Cat.3, Cat.4 および Cat.5 デバイスでのみ使用できます。
- ビット 2 **DBG_TIM4_STOP** : コア停止時に TIM4 カウンタは停止
0 : コアが停止した場合も TIM4 のカウンタクロックは供給されます。
1 : コア停止時に TIM4 のカウンタクロックは停止します。
- ビット 1 **DBG_TIM3_STOP** : コア停止時に TIM3 カウンタは停止
0 : コアが停止した場合も TIM3 のカウンタクロックは供給されます。
1 : コア停止時に TIM3 のカウンタクロックは停止します。
- ビット 0 **DBG_TIM2_STOP** : コア停止時に TIM2 カウンタは停止
0 : コアが停止した場合も TIM2 のカウンタクロックは供給されます。
1 : コア停止時に TIM2 のカウンタクロックは停止します。

30.16.5 デバッグ MCU APB2 凍結レジスタ (DBGMCU_APB2_FZ)

DBGMCU_APB2_FZ レジスタは、DEBUG 中、MCU の設定に使用されます。これは、次の APB2 ペリフェラルに影響します。

- 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 0xE004 200C にある外部 PPB バス上に配置されています。

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ありません)。システムリセット中に、デバッガによって書き込みが可能です。

アドレス : 0xE004 200C

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
予約済み															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
予約済み											DBG_TIM11_STOP	DBG_TIM10_STOP	DBG_TIM9_STOP	予約済み	
											rw	rw	rw		

ビット 31:5 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 4:2 **DBG_TIMx_STOP** : コア停止時に TIMx カウンタは停止 (x=9..11)

- 0 : コアが停止した場合も当該タイマカウンタのクロックは供給されます。
- 1 : コア停止時に当該タイマ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。



30.17 TPIU (トレースポートインタフェースユニット)

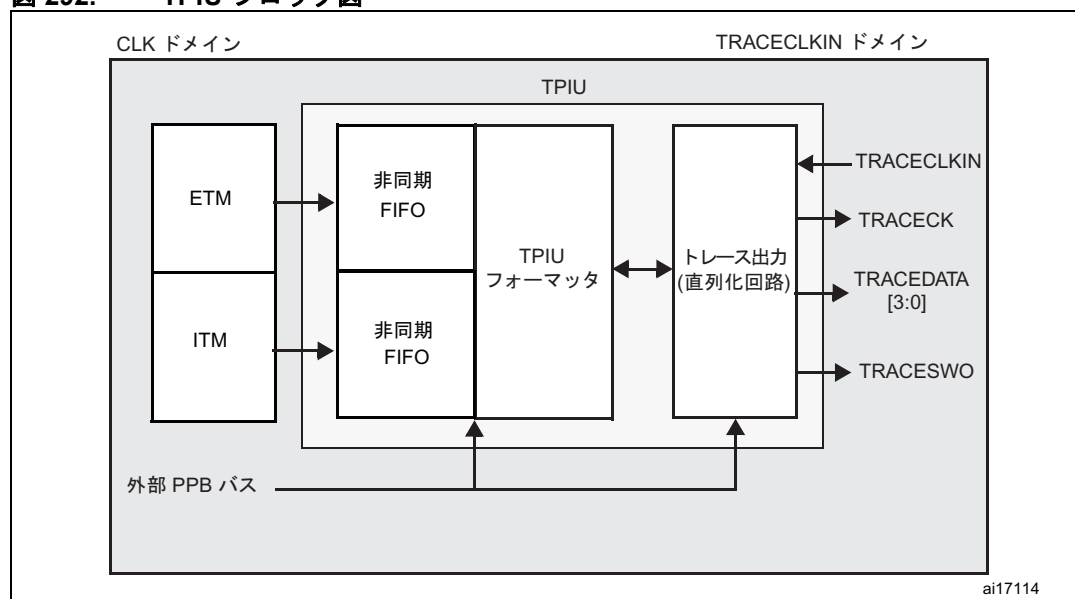
30.17.1 概要

TPIU は、ITM と ETM からオンチップトレースデータ間のブリッジとしての役割を果たします。

出力データストリームによってカプセル化されたトレースソース ID は、TPA (トレースポートアナライザ) によってキャプチャされます。

低コストのデバッグを可能にするため、コアには単純な TPIU (特別バージョンの CoreSight TPIU から構成) が内蔵されています。

図 292. TPIU ブロック図



30.17.2 TRACE ピンの割り当て

- 非同期モード

非同期モードは、1本のピンを追加で必要とし、すべてのパッケージで使用できます。非同期モードは、JTAG モードではなくシリアルワイヤモードを使用する場合にのみ使用できます。

表 206. 非同期 TRACE ピンの割り当て

TPIU ピン名	TRACE 同期モード		STM32L1xxxx ピン割り当て
	タイプ	説明	
TRACESWO	O	TRACE 非同期データ出力	PB3

- 同期モード

同期モードは、データトレースサイズに応じて 2~6 本のピンを追加で必要とし、大型パッケージでのみ使用できます。さらに、JTAG モードとシリアルワイヤモードでも使用でき、非同期トレースよりも優れた帯域幅出力機能を提供します。

表 207. 同期 TRACE ピンの割り当て

TPIU ピン名	TRACE 同期モード		STM32L1xxxx ピン割り当て
	タイプ	説明	
TRACECK	O	TRACE クロック	PE2
TRACED[3:0]	O	TRACE 同期データ出力 1、2、または 4 とすることが出来ます。	PE[6:3]

TPIU TRACE ピンの割り当て

デフォルトでは、これらのピンは割り当てられません。これらのピンを割り当てるには、**MCU デバッグコンポーネント設定レジスタ**の TRACE_IOEN および TRACE_MODE ビットをセットします。この設定はデバッグホストで行う必要があります。

さらに、割り当てるピン数は、トレースの設定（非同期/同期）によって異なります。

- 非同期モード：1本のピンが追加で必要です。
- 同期モード：データトレースポートレジスタのサイズ（1、2、または 4）に応じて、2~5 本のピンが追加で必要です。
 - TRACECK
 - TRACED(0)、ポートサイズが 1、2、または 4 に設定された場合
 - TRACED(1)、ポートサイズが 2 または 4 に設定された場合
 - TRACED(2)、ポートサイズが 4 に設定された場合
 - TRACED(3)、ポートサイズが 4 に設定された場合

TRACE ピンを割り当てるには、デバッグホストは、デバッグ MCU 設定レジスタ (DBGMCU_CR) の TRACE_IOEN ビットと TRACE_MODE[1:0] ビットをプログラミングする必要があります。デフォルトでは、TRACE ピンは割り当てられません。

このレジスタは、外部 PPB に配置され、PORESET によってリセットされます（システムリセットではありません）。このレジスタは、システムリセット中にデバッグによる書き込みができます。

表 208. フレキシブルな TRACE ピンの割り当て

DBGMCU_CR レジスタ		ピン割り当て:	TRACE IO ピンの割り当て					
TRACE _IOEN	TRACE_ MODE [1:0]		PB3 / JTDO/ TRACESWO	PE2/ TRACECK	PE3 / TRACED[0]	PE4 / TRACED[1]	PE5 / TRACED[2]	PE6 / TRACED[3]
0	XX	トレースなし (デフォルト 状態)	解放 ⁽¹⁾	-				
1	00	非同期トレース	TRACESWO	-	-	解放 (GPIO として使用可能)		
1	01	同期トレース、 1 ビット	解放 ⁽¹⁾	TRACECK	TRACED[0]	-	-	-
1	10	同期トレース、 2ビット		TRACECK	TRACED[0]	TRACED[1]	-	-
1	11	同期トレース、 4ビット		TRACECK	TRACED[0]	TRACED[1]	TRACED[2]	TRACED[3]

1. シリアルワイヤモードが使用されているときには解放されます。しかし、JTAG が使用されると、JTDO に割り当てられます。

注: デフォルトでは、TPIU の TRACECLKIN 入力クロックは GND に接続されています。このクロックは、TRACE_IOEN ビットがセットされてから 2 クロックサイクル後に HCLK に割り当てられます。

デバッグは、TPIU の SPP_R (選択ピンプロトコル) レジスタの PROTOCOL[1:0] ビットに書き込むことによって、トレースモードをプログラミングする必要があります。

- PROTOCOL=00: トレースポートモード (同期モード)。
- PROTOCOL=01 または 10: シリアルワイヤ (マンチェスタまたは NRZ) モード (非同期モード)。デフォルト状態は 01 です。

次に、TPIU の CPSPS_R (カレント同期化ポートサイズレジスタ) のビット [3:0] に書き込むことによって TRACE ポートサイズも設定します。

- 0x1: 1 ピンの場合 (デフォルト状態)
- 0x2: 2 ピンの場合
- 0x8: 4 ピンの場合

30.17.3 TPIU フォーマッタ

フォーマッタプロトコルは、データを 16 バイトフレームで出力します。

- 7 バイトのデータ
- 8 バイトの混合バイトであり、以下の構成になっています。
 - 1 ビット (LSB): データバイト ("0") または ID バイト ("1") を示します。
 - 7 ビット (MSB): データまたはソース ID トレースの変更とすることができます。
- 1 バイトの補助ビット: 各ビットは 8 つの混合バイトのいずれかに対応します。
 - 対応するバイトがデータであった場合、このビットはデータのビット 0 を与えます。
 - 対応するバイトが ID 変更であった場合、このビットはその ID 変更が有効になるタイミングを示します。

注: 詳細については、ARM® CoreSight Architecture Specification v1.0 (ARM® IHI 0029B) を参照してください。

30.17.4 TPIU フレーム同期パケット

TPIU は 2 種類の同期パケットを生成できます。

- フレーム同期パケット (フルワード同期パケット)
これはワード 0x7F_FF_FF_FF (LSB ファーストで発行) から構成されます。ID ソースコード 0x7F が使用されていない場合、このシーケンスは他のタイミングでは発生できません。
このパケットは、フレーム間で定期的に出力されます。
連続モードでは、同期フレームが検出されると、TPA はこれらすべてのフレームを破棄する必要があります。
- ハーフワード同期パケット
これはハーフワード 0x7F_FF (LSB ファーストで発行) から構成されます。
このパケットはフレーム間またはフレーム内で定期的に出力されます。
これらのパケットは連続モードでのみ生成され、TPA は TRACE ポートが IDLE モードである (TRACE はキャプチャされない) ことを検出できるようになります。TPA によって検出されたパケットは、破棄する必要があります。

30.17.5 同期フレームパケットの送信

コアの TPIU には、同期カウンタレジスタは実装されていません。したがって、同期トリガは DWT によってのみ生成できます。DWT 制御レジスタ (SYNCTAP[11:10] ビット) と DWT カレント PC サンプラサイクルカウントレジスタの説明を参照してください。

TPIU フレーム同期パケット (0x7F_FF_FF_FF) は次の場合に発行されます。

- 毎回の TPIU リセットの解除後。このリセットは、TRACECLKIN クロックの立ち上がりエッジに同期して解除されます。つまり、DBGMCU_CFG レジスタの TRACE_IOEN ビットがセットされると、このパケットが送信されます。この場合、ワード 0x7F_FF_FF_FF の後に、フォーマットされたパケットは続きません。
- 毎回の DWT トリガ時 (DWT は事前に設定されているものとします)。次の 2 つの場合があります。
 - ITM の SYNENA ビットがリセットされた場合、フォーマットされたストリームなしで、ワード 0x7F_FF_FF_FF だけが発行されます。
 - ITM の SYNENA ビットがセットされた場合、TPIU によってフォーマット (トレースソース ID を追加) された ITM 同期パケットが続きます (0x80_00_00_00_00_00)。

30.17.6 同期モード

トレースデータの出力サイズは 4、2、または 1 ピンから選択できます (TRACED[3:0])。

出力クロックはデバッグに出力されます (TRACECK)。

なお、TRACECLKIN は内部で駆動され、TRACE が使用される場合にのみ HCLK に接続されます。

注： 同期モードでは、必ずしも安定したクロック周波数を提供する必要はありません。

TRACE I/O (TRACECK を含む) は TRACCLKIN (HCLK と同等) の立ち上がりエッジで駆動されます。したがって、TRACECK の出力周波数は HCLK/2 になります。

30.17.7 非同期モード

これは 1 ピン (非同期出力ピン TRACESWO) のみを使用してトレースを出力する低コストな方法ですが、その帯域幅は制限されます。

SW-DP ピンを使用する場合、TRACESWO ピンは JTDO ピンとマルチプレクスされます。このように、この機能はすべての STM32L1xxxx パッケージで使用できます。

この非同期モードでは、TRACECLKIN に一定の周波数が要求されます。標準的な UART (NRZ) のキャプチャ機構では、5 % の精度が必要です。マンチェスタエンコード方式では、10 % までの誤差が許されます。

30.17.8 内部の TRACECLKIN 接続 STM32L1xxxx

STM32L1xxxx では、この TRACECLKIN 入力は内部で HCLK に接続されます。つまり、非同期トレースモードでは、アプリケーションは、CPU 周波数が安定しているタイムフレームのみを使用できます。

注： **重要：** 非同期トレースを使用する場合は、以下のことに注意してください。

STM32L1xxxx MCU のデフォルトクロックは、内部 RC オシレータです。リセット中の周波数はリセット解除後の周波数とは異なります。これは、RC 較正はシステムリセット中のデフォルト動作であり、システムリセット解除のたびに更新されるからです。

したがって、トレースポートアナライザ (TPA) は、システムリセット中に (TRACE_IOEN ビットによって) トレースを有効にするべきではありません。なぜなら、同期フレームパケットは、リセット解除後に送信されるトレースパケットとは異なるビット時間で発行されるからです。

30.17.9 TPIU レジスタ

TPIU APB レジスタは、デバッグ例外およびモニタ制御レジスタ (DEMCR) の TRCENA ビットがセットされた場合にのみ読み出し/書き込みが可能です。そうでない場合、これらのレジスタはゼロとして読み出されます (このビットの出力は TPIU の PCLK を有効にします)。

表 209. 重要な TPIU レジスタ

アドレス	レジスタ	説明
0xE0040004	現在のポートサイズ	トレースポートサイズの選択 : ビット 0 : ポートサイズ = 1 ビット 1 : ポートサイズ = 2 ビット 2 : ポートサイズ = 3、サポートされません ビット 3 : ポートサイズ = 4 1 ビットのみセットする必要があります。デフォルトでは、ポートサイズは 1 ビットです。(0x00000001)
0xE00400F0	選択ピンプロトコル	トレースポートプロトコルの選択 : ビット 1:0 = 00 : 同期トレースポートモード 01 : シリアルワイヤ出力 - マンチェスタ (デフォルト値) 10 : シリアルワイヤ出力 - NRZ 11 : 予約済み
0xE0040304	フォーマッタおよびフラッシュ制御	ビット 31-9 = 常に "0" ビット 8 = TrgIn = 常に "1" で、トリガが指定されたことを示します ビット 7-4 = 常に 0 ビット 3-2 = 常に 0 ビット 1 = EnFCont。同期トレースモード (Select_Pin_Protocol レジスタのビット 1:0 = 00) では、このビットは強制的に "1" にされ、フォーマッタは自動的に連続モードで有効にされます。非同期モード (Select_Pin_Protocol レジスタのビット 1:0 <> 00) では、このビットの書き込みによって、フォーマッタを有効/無効にできます。 ビット 0 = 常に 0 このレジスタのデフォルト値は 0x102 です。 注: 同期モードでは、TRACECTL ピンがチップ外に配置されないため、フォーマッタは常に連続モードで有効になります。このように、フォーマッタは、トレースパケットの転送元を識別するための制御パケットを挿入します。
0xE0040300	フォーマッタおよびフラッシュステータス	Cortex®-M3 では使用されず、常に 0x00000008 が読み出されます。

30.17.10 設定例

- デバッグ例外およびモニタ制御レジスタ (DEMCR) の TRCENA ビットをセットします。
- TPIU カレントポートサイズレジスタに希望する値を書き込みます (デフォルトは 0x1 で 1 ビットのポートサイズ)。
- TPIU フォーマッタおよびフラッシュ制御レジスタに 0x102 (デフォルト値) を書き込みます。
- TPIU 選択ピンプロトコルレジスタに書き込んで同期/非同期モードを選択します。例: 非同期 NRZ モード (UART に類似) の場合は 0x2 を書き込みます。
- DBGMCU 制御レジスタに 0x20 (IO_TRACEN ビット) を書き込み、TRACE I/O を非同期モードとします。このとき、TPIU 同期パケット (FF_FF_FF_7F) が発行されます。
- ITM を設定し、ITM スティムラスレジスタに書き込んで値を出力します。



30.18 DBG レジスタマップ

次の表に、デバッグレジスタの要約を示します。

表 210. 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	
0xE0042000	DBGMCU_IDCODE	REV_ID																予約済み	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	X	X	X	
0xE0042004	DBGMCU_CR	予約済み																							TRACE_MODE[1:0]		TRACE_IOEN		予約済み	DBG_	DBG_	DBG_		
	リセット値																								0	0	0	0		0	0			
0xE0042008	DBGMCU_APB1_FZ	予約済み										DBG_I2C2_SMBUS_TIMEOUT	DBG_I2C1_SMBUS_TIMEOUT	予約済み							DBG_IWDG_STOP	DBG_WWDG_STOP	DBG_RTC_STOP	予約済み					DBG_TIM7_STOP	DBG_TIM6_STOP	DBG_TIM5_STOP	DBG_TIM4_STOP	DBG_TIM3_STOP	DBG_TIM2_STOP
	リセット値											0	0								0	0	0						0	0	0	0	0	0
0xE004200C	DBGMCU_APB2_FZ	予約済み																											DBG_TIM11_STOP			DBG_TIM10_STOP	DBG_TIM9_STOP	予約済み
	リセット値																												0	0	0			

1. リセット値は製品に依存します。詳細については、[セクション 30.6.1 : MCU デバイス ID コード](#)を参照してください。

31 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、すべての STM32L1xxxx デバイスに適用されます。

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。電子署名には、出荷時にプログラミングされた識別データが含まれています。この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 STM32L1xxxx マイクロコントローラの特性に合わせて自動的に整合させることができます。

31.1 メモリサイズレジスタ

31.1.1 フラッシュサイズレジスタ

ベースアドレス : Cat.1 および Cat.2 デバイスの場合、0x1FF8004C

ベースアドレス : Cat.3、Cat.4、Cat.5 デバイスの場合、0x1FF800CC

読み出し専用 = 0xXXXX (X は出荷時にプログラミングされます)

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

DEV_ID = 0x416 または 0x427 または 0x429 または 0x437 の場合、このフィールド値は、デバイスのフラッシュメモリサイズをキロバイト (K バイト) で示します。

例 : 0x0080 = 128 KB。

DEV_ID = 0x436 の場合、フィールド値は“0”または“1”を取り得ますが、384 K バイトの場合“0”、256 K バイトの場合、“1”となります。

注 : DEV_ID = 0x429 の場合、F_SIZE の LSB 部分のみ、即ち F_SIZE[7:0] が有効です。MSB 部分である F_SIZE[15:8] は予約済みであり、無視されます。

31.2 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ミングする前に、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

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：Cat.1 および Cat.2 デバイスの場合、0x1FF80050 であり、Cat.3、Cat.4、Cat.5 デバイスの場合、0x1FF800D0 となります。

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ミングされます）

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ミングされます）

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 = 0xxxxx xxxx (X は出荷時にプログラミングされます)

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

索引

A

ADC_CCR 317
ADC_CR1 84, 300
ADC_CR2 303
ADC_CSR 316
ADC_DR 314
ADC_HTR 309
ADC_JDRx 314
ADC_JOFRx 309
ADC_JSQR 313
ADC_LTR 310
ADC_SMPR0 315
ADC_SMPR1 306
ADC_SMPR2 212, 213, 307
ADC_SMPR3 308
ADC_SQR1 310
ADC_SQR2 311
ADC_SQR3 311
ADC_SQR4 312
ADC_SQR5 312
ADC_SR 298, 359
AES_CR 598
AES_DINR 601
AES_DOUTR 601
AES_IVR 603
AES_KEYRx 602
AES_SR 600

C

COMP_CSR 351
CRC_DR 96
CRC_IDR 97

D

DAC_CR 333
DAC_DHR12L1 338
DAC_DHR12L2 339
DAC_DHR12LD 341
DAC_DHR12R1 337
DAC_DHR12R2 339
DAC_DHR12RD 340
DAC_DHR8R1 338
DAC_DHR8R2 340
DAC_DHR8RD 342
DAC_DOR1 342
DAC_DOR2 343

DAC_SR 343
DAC_SWTRIGR 336
DBGMCU_APB1_FZ 907
DBGMCU_APB2_FZ 909
DBGMCU_CR 905
DBGMCU_IDCODE 890
DMA_CCRx 267
DMA_CMARx 269
DMA_CNDTRx 268
DMA_CPARx 269
DMA_IFCR 266
DMA_ISR 265

E

EXTI_EMR 249
EXTI_FTSR 250
EXTI_IMR 248
EXTI_PR 251
EXTI_RTSR 249
EXTI_SWIER 250

F

FSMC_BCR1..4 671
FSMC_BTR1..4 674
FSMC_BWTR1..4 676

G

GPIOx_AFRH 193
GPIOx_AFRL 192
GPIOx_BRR 193
GPIOx_BSRR 190
GPIOx_IDR 189
GPIOx_LCKR 191
GPIOx_MODER 187
GPIOx_ODR 190
GPIOx_OSPEEDR 188
GPIOx_OTYPER 188
GPIOx_PUPDR 189

I

I2C_CCR 713
I2C_CR1 700
I2C_CR2 703
I2C_DR 706
I2C_OAR1 705



I2C_OAR2 705
I2C_SR1 707
I2C_SR2 711
I2C_TRISE 714
IWDG_KR 569
IWDG_PR 570
IWDG_RLR 570
IWDG_SR 571

L

LCD_CLR 390
LCD_CR 386
LCD_RAM 391

P

PWR_CR 123
PWR_CSR 126

R

RCC_AHB1RSTR 151
RCC_AHBENR 157, 163
RCC_APB1ENR 161, 167
RCC_APB1RSTR 154
RCC_APB2ENR 159, 165
RCC_APB2RSTR 153
RCC_CFGR 145
RCC_CIR 148
RCC_CR 142
RCC_CSR 169
RI_ASCR1 209
RI_ASCR2 211
RI_HYSCR1 212
RI_HYSCR4 214
RI_ICR 207
RTC_ALRMAR 553
RTC_ALRMBR 554
RTC_ALRMBSSR 563
RTC_BKxR 564
RTC_CALIBR 552
RTC_CALR 559
RTC_CR 544
RTC_DR 543
RTC_ISR 547
RTC_PRER 549
RTC_SHIFTR 556
RTC_SSR 555
RTC_TCR 560
RTC_TR 542
RTC_TSDR 558
RTC_TSSSR 558

RTC_TSTR 557
RTC_WPR 555
RTC_WUTR 551

S

SDIO_CLKCR 866
SDIO_DCOUNT 872
SDIO_DCTRL 871
SDIO_DLEN 870
SDIO_DTIMER 870
SDIO_FIFO 879
SDIO_FIFOCNT 879
SDIO_ICR 874
SDIO_MASK 876
SDIO_POWER 865
SDIO_RESPCMD 869
SDIO_RESPx 869
SDIO_STA 873
SPI_CR1 813
SPI_CR2 815
SPI_CRCPR 819
SPI_DR 818
SPI_I2SCFGR 821
SPI_I2SPR 822
SPI_RXCR 820
SPI_SR 817
SPI_TXCR 820
SYSCFG_EXTICR1 226
SYSCFG_EXTICR2 227
SYSCFG_EXTICR3 227
SYSCFG_EXTICR4 228
SYSCFG_MEMRMP 224

T

TIM2_OR 452
TIMx_ARR 447, 490, 504, 521
TIMx_CCER 445, 489, 503
TIMx_CCMR1 441, 486, 500
TIMx_CCMR2 444
TIMx_CCR1 448, 491, 492, 505, 506, 507
TIMx_CCR2 448, 491
TIMx_CCR3 449
TIMx_CCR4 449
TIMx_CNT 447, 490, 504, 520
TIMx_CR1 432, 477, 495, 496, 517
TIMx_CR2 434, 479, 519
TIMx_DCR 450
TIMx_DIER 437, 482, 497, 519
TIMx_DMAR 451
TIMx_EGR 440, 485, 499, 520
TIMx_PSC 447, 490, 504, 521

TIMx_SMCR 435, 480
TIMx_SR 438, 483, 497, 520

U

USART_BRR 760
USART_CR1 761
USART_CR2 763
USART_CR3 764
USART_DR 760
USART_GTPR 766
USART_SR 758
USB_ADDRn_RX 635
USB_ADDRn_TX 634
USB_BTABLE 628
USB_CNTR 622
USB_COUNTn_RX 636
USB_COUNTn_TX 635
USB_DADDR 628
USB_EPnR 629
USB_FNR 627
USBISTR 624

W

WWDG_CFR 578
WWDG_CR 577
WWDG_SR 578



32 改版履歴

表 211. 改版履歴

日付	リビジョン	変更内容
2010 年 7 月 2 日	1	初版リリース。
2010 年 10 月 1 日	2	セクション 7.3.2 以降、セクション 表 37. で注を変更。 図 15 : クロックツリー (132 ページ) を更新 表 32 : Standby モード (120 ページ) (ウェイクアップ遅延時間) を変更 セクション 12.12 : 温度センサと内部基準電圧 (295 ページ) を更新 セクション 16.5 : LCD レジスタ (386 ページ) で SOF と SOFC のビット説明を更新 RTC レジスタ書き込み保護 (529 ページ) を更新 I2C マスタレシーバ (689 ページ) を更新
2010 年 11 月 29 日	3	セクション 5.3.1 : 低消費電力モードでのクロック動作 (64 KHz の代わりに 65 kHz) を変更 セクション 5.3.9 : RTC とコンパレータを使用した、Stop モードおよび Standby モードからのデバイスのウェイクアップ (121 ページ) を変更 Stop モードからの RTC 自動ウェイクアップ (AWU) (122 ページ) とセクション 5.4.1 : PWR 電源制御レジスタ (PWR_CR) (123 ページ) でシーケンスの順番を変更 セクション 6.2.3 : MSI クロック (135 ページ) を変更 セクション 6.3.2 : 内部クロックソース校正レジスタ (RCC_ICSCR) (144 ページ) で MSIRANGE ビットの説明を変更 セクション 5.4.1 : PWR 電源制御レジスタ (PWR_CR) で PLS[2:0] ビットの説明を変更 セクション 6.2.6 : LSI クロック (137 ページ) を変更 セクション 8.4.7 : RI ヒステリシス制御レジスタ (RI_HYSCR4) (214 ページ) (「ST」の代わりに「SCM」) を変更 セクション 8.5.7 : SYSCFG レジスタマップ (229 ページ) (「SYSCFG_MEMRM」の代わりに「SYSCFG_MEMRMP」) を変更 セクション 12.3 : ADC の機能詳細 で 注 : を更新 セクション 12.3.3 : チャネルの選択 を更新。 図 53 : 温度センサと VREFINT チャネルのブロック図を含む セクション 12.12 : 温度センサと内部基準電圧 全体を更新 セクション 12.15.2 : ADC 制御レジスタ 1 (ADC_CR1) (300 ページ) で AWDCH ビットの説明を更新 セクション 12.15.3 : ADC 制御レジスタ 2 (ADC_CR2) (303 ページ)。 セクション 12.15.15 : ADC インジェクトシーケンスレジスタ (ADC_JSQR) (313 ページ) で、JSQ ビットの説明を更新し、注を追加 図 65 : COMP2 の接続 (Cat.1 と Cat.2 デバイス) (348 ページ) を変更 セクション 14.4 : コンパレータ 1 (COMP1) (346 ページ) と セクション 14.9.1 : COMP コンパレータ制御およびステータスレジスタ (COMP_CSR) (351 ページ) を変更 セクション 16.2 : LCD の主な機能 (366 ページ) を変更

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2010 年 11 月 29 日	3 (続き)	セクション 20 : リアルタイムクロック (RTC) (523 ページ) の内容を変更し、ビット名とレジスタ名を変更、セクション 20.3.6 (531 ページ) で、APB 対 RTCCLK 周波数についての注を追加 表 91 : 37 kHz での最小/最大 IWDG タイムアウト期間 (LSI) (568 ページ) を変更 セクション : LIN 受信 (743 ページ) を変更 で注を変更 パケットバッファの構造と使用 (612 ページ) セクション : (890 ページ) を変更 図 68 : ウィンドウモードのコンパレータ (350 ページ) を変更 セクション 30.6.1 : MCU デバイス ID コード (890 ページ) で説明されている REV_ID(15:0) を変更 セクション 31 : デバイス電子署名 (917 ページ) を追加。 セクション 31.1.1 : フラッシュサイズレジスタ (917 ページ) を追加。
2010 年 2 月 24 日	4	表 48 : ベクタテーブル (Cat.1 および Cat.2 デバイス) (238 ページ) (TIM9 および LCD) を変更 図 72 : LCD コントローラのブロック図 (368 ページ) を変更 セクション 16.5.2 : LCD フレーム制御レジスタ (LCD_FCR) (387 ページ) で、PON[2:0]、CC[2:0] および PS[3:0] ビットの説明を変更 セクション 5.3 : 低消費電力モードを変更 セクション 7.3.13 : OSC32_IN/OSC32_OUT ピンの GPIO PC14/PC15 ポートピンとしての使用 および セクション 7.3.14 : OSC_IN/OSC_OUT ピンの GPIO PH0/PH1 ポートピンとしての使用 を変更 セクション 13.1 : DAC の概要 (321 ページ) を変更 セクション 16.2 : LCD の主な機能 (366 ページ) に注 2 を追加 セクション 17.4.17 : TIMx DMA 制御レジスタ (TIMx_DCR) (450 ページ) で ビットの説明を変更 セクション 17.4.18 : 完全転送の TIMx DMA アドレス (TIMx_DMAR) (451 ページ) で DMAB[15:0] ビットの説明を変更 セクション 26.3.7 : DMA リクエスト (695 ページ) を変更 図 211 : スレーブシーバの転送シーケンス図 (685 ページ) の下に注を追加 セクション : スレーブ通信の終了 (685 ページ) を変更 セクション 26.6.6 : I²C ステータスレジスタ 1(I2C_SR1) (707 ページ) を変更 セクション 26.6.7 : I²C ステータスレジスタ 2(I2C_SR2) (711 ページ) に注を追加 セクション 26.6.8 : I²C クロック制御レジスタ (I2C_CCR) (713 ページ) で注を変更 セクション 28 : シリアルペリフェラルインタフェース (SPI) (769 ページ) を変更 図 211 : スレーブシーバの転送シーケンス図 (685 ページ) の下に注を追加
2012 年 1 月 17 日	5	CRC 計算ユニットを セクション 4 に移動 Cat.1 デバイスに対して、セクション 9 : タッチセンシング I/O、セクション 15 : オペアンプ (OPAMP)、セクション 23 : 高度暗号化標準ハードウェアアクセラレータ (AES)、セクション 25 : フレキシブルスタティックメモリコントローラ (FSMC)、セクション 29 : セキュアデジタル入力/出力インタフェース (SDIO) を追加

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2012 年 1 月 17 日	5 (続き)	セクション 6.2.9 : クロックセキュリティシステム (CSS) (138 ページ) を変更 セクション 6.3.1 : クロック制御レジスタ (RCC_CR) (142 ページ) を変更
2012 年 7 月 13 日	6	中容量以上のデバイスについて更新。 図 2 : システムアーキテクチャ (Cat.3 デバイス) (43 ページ) を追加。 表 9 : フラッシュメモリモジュール構成 (Cat.3 デバイス) (55 ページ) を追加 図 26 : Cat.3 製品のルーティングインタフェース (RI) ブロック図 (199 ページ) を追加 表 49 : ベクタテーブル (Cat.3 デバイス) (240 ページ) を追加 図 35 : Cat.3STM32L1xxxx デバイスの DMA ブロック図 (255 ページ) を追加 セクション 5.2.3 : プログラマブル電圧検出器 (PVD) で、V_{DDA} を削除 セクション 10.2 : 外部割り込み/イベントコントローラ (EXTI) で、「パルスまたはペンディング」を「イベントまたは割り込み」で置換 セクション 28 : シリアルペリフェラルインタフェース (SPI) で、「単方向通信」と「単方向モード」を「半二重通信」と「半二重モード」に置換 図 25 : Cat.1 および Cat.2 製品のルーティングインタフェース (RI) ブロック図 (198 ページ) および 図 27 : Cat.4 および Cat.5 製品のルーティングインタフェース (RI) ブロック図 (200 ページ) を訂正 セクション 9 : タッチセンシング I/O で、「最大 36 チャンネルの容量センシングをサポート」で、「36」を「34」に置換 「最大 6 チャンネルの容量センシングに対して 1 つのサンプリングコンデンサ」で、「6」を「4」に置換 「タッチに適合」を「タッチキーに適合」に置換 「STM32 タッチセンシング」を「STM32L1xx STM タッチ」に置換 「STM32 タッチセンシングライブラリ STM32L1xx」を「STM タッチセンシングライブラリ」に置換 図 29 : 表面電荷移動アナログ I/O グループ構造 (232 ページ) で、G1_IO2 ピンへの接続を訂正 セクション 17.4.3 : TIMx スレープモード制御レジスタ (TIMx_SMCR) で、ETF[3:0] ビットフィールドの説明の表示を訂正 図 213 : マスタレーバの転送シーケンス図 (690 ページ) を追加 セクション 30.6.1 : MCU デバイス ID コードで、値 = "0x1018" を 1 行追加 表 31 : Stop モード (119 ページ) で、「WUF ビット・・・をクリアします。」を追加 セクション 6.3.14 : 制御/ステータスレジスタ (RCC_CSR) で、RTCSEL ビットの説明を変更し、「リセット値」を「パワーオンリセット値」に変更 セクション 30.6.1 : MCU デバイス ID コード (890 ページ) で、REV_ID ビットフィールドにおいて、「Rev A」ラベルを、行 0x1018 に移動し、説明中、「中容量以上」を追加

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2012 年 7 月 13 日	6 (続き)	セクション 12.3.3 : チャネルの選択 (278 ページ) で、“2728 変換”を訂正し、セクション 12.3 : ADC の機能詳細 (274 ページ) で、“内部接続のために・・・”の注を変更 図 31 : タイマモードアクイジションのロジック (235 ページ) で、OC2 マルチプレクサと NOR ゲート間の矢印を更新 図 159 : 標準の 8 MHz HSE を使用したオーディオ周波数精度 (Cat.3, Cat.4, Cat.5 デバイスの場合) (806 ページ) で、“7.9370 %”を訂正 セクション 8.4 : RI レジスタ (207 ページ) で、全 CMR5 レジスタに対するビットの説明を更新 セクション 8.4.2 : RI アナログスイッチ制御レジスタ (RI_ASCR1) (209 ページ) で、“CH31 GR7-1”、“COMP1_SW1”および“CH31 GR11-5”を置換 セクション 8.4.2 : RI アナログスイッチ制御レジスタ (RI_ASCR1) (209 ページ) で、ビット 25:22、ビット 5、ビット 4 の説明を変更し、セクション 14.9.1 : COMP コンパレータ制御およびステータスレジスタ (COMP_CSR) (351 ページ) で、ビット 5 SW1 の説明を変更 セクション 15.3.2 : ADC 入力として OPAMP 出力を使用 (357 ページ) で、RI(RI_ASRCR1) セクションへのクロスリファレンスを変更 表 44 : RI レジスタマップとリセット値 (222 ページ) および 表 70 : COMP レジスタマップとリセット値 (354 ページ) を更新 セクション 8.4.3 : RI アナログスイッチ制御レジスタ 2 (RI_ASCR2) (211 ページ) で、ビット 29 「GR5-4」 およびビット 15 「GR4-4」を追加 セクション 5.3.4 : 低消費電力 run モード (LP run) (114 ページ) で、“f_MSI range1”を追加
2013 年 3 月 14 日	7	セクション 7 : 汎用 I/O (GPIO) で OSPEEDR の説明を更新。 セクション 20.3.1 : クロックとプリスケアラで、最大入力周波数を更新。 セクション 25 : フレキシブルスタティックメモリコントローラ (FSMC) で、 - 図 188 : FSMC のブロック図を更新。 - セクション 25.5.4 : NOR 型フラッシュ/PSRAM コントローラ非同期トランザクションを更新。 - セクション : モード 2/B - NOR 型フラッシュで、モード B とモード 1 の差異を変更。 - セクション : モード C - NOR 型フラッシュ - OE トグリングで、モード C とモード 1 の差異を変更。 - セクション : モード D - 拡張アドレスによる非同期アクセスで、モード D とモード 1 の差異を変更。 - 図 203 : 読み出しアクセス中の非同期ウェイト、図 204 : 書き込みアクセス中の非同期ウェイト、図 205 : ウェイト設定、図 206 : 同期マルチプレクス読み出しモード-NOR、PSRAM (CRAM) および 図 207 : 同期マルチプレクス書き込みモード-NOR、PSRAM (CRAM) で、NWAIT 信号を更新。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2013 年 3 月 14 日	7 (続き)	– セクション : SRAM/NOR 型フラッシュチップセレクト制御レジスタ 1.4 (FSMC_BCR1..4) を更新。 – 表 132 : FSMC_BCRx ビットフィールド と 表 134 : FSMC_BCRx ビットフィールド で WAITEN の定義を更新。 – 表 116 : FSMC_BCRx ビットフィールド から 表 135 : FSMC_BTRx ビットフィールド を更新。 セクション 30.6.1 : MCU デバイス ID コード で、ビット 11:0 の説明を更新。 セクション 31.1.1 : フラッシュサイズレジスタ および セクション 31.2 : ユニークデバイス ID レジスタ (96 ビット) を更新。
2013 年 4 月 19 日	8	STM32L100xx バリュールラインを追加 表紙で、PM0056 へのリファレンスを追加 セクション 5.1.3 : RTC レジスタと RTC バックアップレジスタ を更新 セクション 5.1.7 : 2.0 V 以下に VDD 低下した場合の電圧レギュレータとクロックの管理、セクション 5.3.1 : 低消費電力モードでのクロック動作 および セクション 6.2.8 : システムクロックソース周波数対電圧範囲 で、4.2 MHz を更新 セクション 6.1.2 : 電源リセット で、「各内部リセットソースに対して・・・」を更新 セクション 8 : システム設定コントローラ (SYSCFG) とルーティンゲインタフェース (RI) で、第 1 パラグラフを削除。 セクション 8.4.3 : RI アナログスイッチ制御レジスタ 2 (RI_ASCR2) で、GR5-4 ビットと GR4-4 ビットを削除し、ビット 28:16 の定義を訂正 セクション 8.5.1 : SYSCFG メモリ再配置レジスタ (SYSCFG_MEMRMP) を更新し、MEM_MODE ビットに FSMC を追加 セクション 8.5.2 : SYSCFG ペリフェラルモード設定レジスタ (SYSCFG_PMC) に、LCD_CAPA を追加 セクション 8.5.3 : SYSCFG 外部割り込み設定レジスタ 1 (SYSCFG_EXTICR1)、セクション 8.5.4 : SYSCFG 外部割り込み設定レジスタ 2 (SYSCFG_EXTICR2)、セクション 8.5.5 : SYSCFG 外部割り込み設定レジスタ 3 (SYSCFG_EXTICR3) および セクション 8.5.6 : SYSCFG 外部割り込み設定レジスタ 4 (SYSCFG_EXTICR4) を更新 バリュールラインデバイスに対し、セクション 9.1 : 概要 を更新 セクション 10 : 割り込みとイベント で、23 行を 24 行で置換 セクション 10.1 : ネスト化されたベクタ割り込みコントローラ (NVIC) を更新 セクション 10.2.5 : 外部割り込み／イベントラインのマッピング で、GPIO の数を更新 図 33 : 外部割り込み／イベントの GPIO マッピング を更新 セクション 10.3 : EXTI レジスタ にビット 23 を追加 セクション 18.3.14 : エンコーダインタフェースモード (TIM9 の場合) を追加 セクション 15.3.3 : 較正 で、OPAMP_OTR と OPAMP_CSR を OPAMP_OTR で置換 セクション 15.4.2 : 通常モードの OPAMP オフセットトリミングレジスタ (OPAMP_OTR) で、ビット 31 の OT_USER を更新

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2013 年 4 月 19 日	8 (続き)	セクション 12 : A/D コンバータ (ADC)、セクション 14 : コンパレータ (COMP)、セクション 15 : オペアンプ (OPAMP)、セクション 17 : 汎用タイマ (TIM2 から TIM5)、セクション 20 : リアルタイムクロック (RTC)、セクション 24 : ユニバーサルシリアルバス (USB) フルスピードデバイスインタフェース、セクション 25 : フレキシブルスタティックメモリコントローラ (FSMC)、セクション 28 : シリアルペリフェラルインタフェース (SPI) および セクション 29 : セキュアデジタル入力/出力インタフェース (SDIO) で、第 1 パラグラフを削除 セクション 16.2 : LCD の主な機能 で、「VLCD レール・・・」の項目を追加 図 73 : 1/3 バイアス、1/4 デューティ を更新 セクション 16.4.5 : 電圧ジェネレータに 外部デカップリング を追加 図 80 : LCD 電圧制御 を更新 セクション 16.4.7 : ダブルバッファメモリ を追加 セクション 18.2.2 : TIM10/TIM11 の主な機能 で、パルスモードを削除 セクション 20.3.13 : タンパ検出 を更新 セクション 20.6.4 : RTC 初期化およびステータスレジスタ (RTC_ISR) で、システムリセット値を更新 表 119、表 120、表 122、表 123、表 125 で、アドレス設定の最小値を 0 に変更 セクション 31.1.1 : フラッシュサイズレジスタ で、DEV_ID = 0x436 に更新
2014 年 3 月 3 日	9	全文書において、「低容量」、「中容量」、「中容量以上」、「高容量」のカテゴリを、Cat.1、Cat.2、Cat.3、Cat.4、Cat.5 で置換。 全文書で、フラッシュメモリを不揮発性メモリ (フラッシュメモリ) に置換 (この用語が全体としてフラッシュメモリにあてはまる場合)。 セクション 1.3 : 用語 を追加。 セクション 1.4 : 製品カテゴリの定義 を更新。 図 2 : システムアーキテクチャ (Cat.3 デバイス) で、GPIOF と GPIOG を削除。 図 4 : システムアーキテクチャ (Cat.5 デバイス) を追加。 セクション 2.2 : メモリ構成 で、第 1 パラグラフを更新。 図 5 : レジスタ境界アドレス で、0x4002 3C00 から 0x4002 3FFF ラインを更新。 セクション 2.5 : フラッシュメモリの概要 を更新。 旧セクション 2.6.1 内蔵フラッシュメモリ を削除。 表 7 : メモリマッピング対ブートモード/物理リマップ を更新。 セクション 3.1 : フラッシュメモリの概要 および セクション 3.2 : フラッシュメモリ構成 を更新。 表 9 : フラッシュメモリモジュール構成 (Cat.3 デバイス) および 表 10 : フラッシュメモリモジュール構成 (Cat.4 デバイス) を更新。 表 11 : フラッシュメモリモジュール構成 (Cat.5 デバイス) を追加。 表 12 : CPU クロック (HCLK) 周波数に対応したウェイトステート数 (WS) を更新。 表 13 : FLASH_ACR に対する許容設定 を更新。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2014 年 3 月 3 日	9 (続き)	セクション 3.4.1 : メモリのロック/アンロック、セクション 3.4.2 : メモリの消去 および セクション 3.4.4 : 書き込み中読出し (RWW) を更新。 表 17 : オプションバイト構成 および 表 18 : オプションバイトの説明を更新。 セクション 3.7.2 : プログラムメモリの書き込み保護機能 (WRP) を更新。 セクション 3.8 : 割り込み で、RDERR を追加。 セクション 3.7.4 : PCROP を更新。 セクション 3.9.7 : ステータスレジスタ (FLASH_SR) で、RDERR ビットを追加。 セクション 3.9.8 : オプションバイトレジスタ (FLASH_OBR) で、SPRMOD ビットを追加。 セクション 3.9.9 : 書き込み保護機能レジスタ (FLASH_WRP_{PRx}) を更新。 表 23 : レジスタマップとリセット値 を更新。 図 8 : 電源の概要 を更新。 セクション 5.1.7 : 2.0 V 以下に VDD 低下した場合の電圧レギュレータとクロックの管理 で、リスト項目 3 を更新。 セクション 5.1.5 : ダイナミック電圧スケーリング管理 および セクション 5.3.6 : 低消費電力 sleep モード (LP スリープ) を更新。 セクション 5.3.7 : Stop モード を更新。 セクション 5.4.1 : PWR 電源制御レジスタ (PWR_CR) で VOS ビットの説明を更新。 表 25 : 性能対 V_{CORE} 範囲 を更新。 セクション 5.4.2 : PWR 電源制御/ステータスレジスタ (PWR_CSR) で WUF ビットの説明を更新。 表 31 : Stop モードで、「モードへの移行」の説明を更新。 図 16 : HSE/LSE クロックソース を更新。 セクション 6.2.2 : HSI クロック および セクション 6.2.3 : MSI クロック を更新。 セクション 6.2.5 : LSE クロック を更新。 セクション 6.2.10 : LSE のクロックセキュリティシステム を更新。 セクション 6.3.5 : AHB ペリフェラルリセットレジスタ (RCC_AHB_{STR}) で、GPIOGRST、GPIOFRST、GPIOHRST の説明において、Cat.2 を削除。 セクション 6.3.8 : AHB ペリフェラルクロックイネーブルレジスタ (RCC_AHB_{ENR}) で、GPIOGEN、GPIOFEN の説明において、Cat.2 を削除。 セクション 6.3.11 : 低消費電力モードレジスタでの AHB ペリフェラルクロックイネーブル (RCC_AHB_{LPENR}) で、GPIOGLPEN、GPIOFLPEN の説明において、Cat.2 を削除。 図 17 : TIM9/TIM10/TIM11 チャンネル 1 入力キャプチャを使用する周波数測定 を更新。 セクション 6.2.14 : TIM9/TIM10/TIM11 による内部/外部クロックの測定 を更新。 OSPEEDR ビットの説明で、周波数値を削除。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2014 年 3 月 3 日	9 (続き)	誤植訂正 : 「GPIOx_IDR」レジスタで、「IDRy[15:0]」を「IDRy」で置換、「GPIOx_ODR」レジスタで、「ODRy[15:0]」を「ODRy」で置換、「GPIOx_OTYPER」レジスタで、「OTy[1:0]」を「OTy」で置換。 セクション 7.3.15 : RTC_AF1 オルタネート機能の選択を更新。 表 39 : GPIO レジスタマップとリセット値を更新。 セクション 7.4.11 : GPIO ポートビットリセットレジスタ (GPIOx_BRR) (x=A..H) を追加。 図 26 : Cat.3 製品のルーティングインタフェース (RI) ブロック図および図 27 : Cat.4 および Cat.5 製品のルーティングインタフェース (RI) ブロック図を更新。 セクション 8.3.1 : 特別 I/O 設定を更新。 表 40 : I/O グループと選択を更新。 図 25 : Cat.1 および Cat.2 製品のルーティングインタフェース (RI) ブロック図、図 26 : Cat.3 製品のルーティングインタフェース (RI) ブロック図および図 27 : Cat.4 および Cat.5 製品のルーティングインタフェース (RI) ブロック図を更新。 セクション 8.3.1 : 特別 I/O 設定を更新。 セクション 表 40. : I/O グループと選択を更新。 セクション 10.1 : ネスト化されたベクタ割り込みコントローラ (NVIC)、セクション 10.3.5 : EXTI ソフトウェア割り込みイベントレジスタ (EXTI_SWIER) および セクション 10.3.6 : EXTI ペンディングレジスタ (EXTI_PR) を更新。 表 48 : ベクタテーブル (Cat.1 および Cat.2 デバイス)、表 49 : ベクタテーブル (Cat.3 デバイス) および 表 50 : ベクタテーブル (Cat.4 および Cat.5 デバイス) を更新。 セクション 11.2 : DMA の主な機能で、最後から 2 番目の項目と、セクションの最後にある注を更新。 図 35 : Cat.3STM32L1xxxx デバイスの DMA ブロック図を更新。 セクション 12.3 : ADC の機能詳細で注 : を更新 セクション 12.3.3 : チャネルの選択を更新。 図 53 : 温度センサと VREFINT チャネルのブロック図を含む セクション 12.12 : 温度センサと内部基準電圧全体を更新 セクション 12.15.2 : ADC 制御レジスタ 1 (ADC_CR1) (300 ページ) で AWDCH ビットの説明を更新 セクション 12.15.3 : ADC 制御レジスタ 2 (ADC_CR2) (303 ページ) で ADC_CFG ビットの説明を更新。 セクション 12.15.15 : ADC インジェクトシーケンスレジスタ (ADC_JSQR) (313 ページ) で、JSQ ビットの説明を更新し、注を追加。 セクション 12.15.2 : ADC 制御レジスタ 1 (ADC_CR1) (300 ページ) で AWDCH ビットの説明を更新。 セクション 12.15.3 : ADC 制御レジスタ 2 (ADC_CR2) (303 ページ) で ADC_CFG ビットの説明を更新。 セクション 12.15.15 : ADC インジェクトシーケンスレジスタ (ADC_JSQR) (313 ページ) で、JSQ ビットの説明を更新し、注を追加。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2014 年 3 月 3 日	9 (続き)	旧の図 63 : コンパレータブロック図を削除。 図 63 : COMP1 の接続 (Cat.1 と Cat.2 デバイス)、図 64 : COMP1 の接続 (Cat.3、Cat.4、Cat.5 デバイス)、図 65 : COMP2 の接続 (Cat.1 と Cat.2 デバイス) および 図 66 : COMP2 の接続 (Cat.3、Cat.4、Cat.5 デバイス) を更新 セクション 14.9.1 : COMP コンパレータ制御およびステータスレジスタ (COMP_CSR) で FCH8 と FCH3 のビット説明を更新。 図 69 : OPAMP1 の信号ルーティングを更新。 図 70 : OPAMP2 の信号ルーティングを更新。セクション 16.4.3 : コモンドライバ、セクション 16.4.5 : 電圧ジェネレータを更新。 図 80 : LCD 電圧制御を更新。 セクション 16.5.2 : LCD フレーム制御レジスタ (LCD_FCR) を更新。 セクション 17.4.12 : TIMx 自動再ロードレジスタ (TIMx_ARR) を更新。 表 81 : TIMx レジスタマップとリセット値を更新。 MMS ビットを削除。 セクション 17.4.7 : TIMx およびキャプチャ/比較モードレジスタ 1 (TIMx_CCMR1) で、IC1F ビットに関連する注を更新。 表 79 : TIMx 内部トリガ接続を更新。 セクション 18.5.12 : TIM10 オプションレジスタ 1 (TIM10_OR) を更新。 セクション 19.4.1 : TIM6 および TIM7 制御レジスタ 1 (TIMx_CR1) を更新。 セクション 20.6.3 : RTC 制御レジスタ (RTC_CR) で、ビット 9 の ALRBIE を ALRBE に変更。 セクション 20.3.11 : RTC の高精度デジタル較正 (Cat.2、Cat.3、Cat.4 および Cat.5 製品のみ) を更新。 と 表 119 : FSMC_BTRx ビットフィールドで BUSTURN の定義を更新。 セクション 26 : I^2C (Inter-integrated Circuit) インタフェース で、略語 Sm (スタンダードモード) と Fm (高速モード) を案内。 すべての「パワーオンリセット」を「バックアップドメインリセット」に置換。 セクション 20.6.20 : RTC バックアップレジスタ (RTC_BKPxR) で、「またはフラッシュ読み出し保護が無効である場合」を削除。 「システムリセット : BYPSHAD = 0 の場合、0x0000 0000」に置換。 セクション 20.6.1 : RTC 時間レジスタ (RTC_TR) において、「RSF フラグがセットされる前のシステムリセット : 0x0000 0000 で、その後正しい値が得られる」場合、BYPSHAD = 1 のときは、影響を受けません。 「システムリセット : BYPSHAD = 0 の場合、0x0000 2101」に置換。c において、「RSF フラグがセットされる前のシステムリセット : 0x0000 2101 で、その後正しい値が得られる」場合、BYPSHAD = 1 のときは、影響を受けません。 RTC_ISR で、SHPF ビットタイプを“r”に変更。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2014 年 3 月 3 日	9 (続き)	セクション 20.2 : RTC の主な機能、セクション 20.3.3 : プログラム可能なアラーム、セクション 20.3.4 : 定期的な自動ウェイクアップ、セクション 20.3.12 : タイムスタンプ機能、セクション 20.3.13 : タンパ検出、セクション 20.3.14 : 較正されたクロックの出力、セクション 20.3.15 : アラーム出力、セクション 20.6.3 : RTC 制御レジスタ (RTC_CR)、セクション 20.6.17 : RTC タンパおよび代替機能設定レジスタ (RTC_TAFRCR) を更新。 セクション 図 164. : RTC のブロック図 (Cat.1 製品) および セクション 図 165. : RTC のブロック図 (Cat.2, Cat.3, Cat.4 および Cat.5 製品) を更新。 図 166 : 独立型ウォッチドッグのブロック図を訂正。 セクション 25 : フレキシブルスタティックメモリコントローラ (FSMC) 全体で、すべての DATALAT を DATLAT に置換。 セクション 25.1 : FSMC の主な機能を更新。セクション全体で、SRAM/CRAM を SRAM/PSRAM に置換。 FSMC_BWTR1.4 のビット 27 から 20 を予約済みに変更。 表 116、表 118、表 121、表 124、表 127、表 130 および 表 134 で、WREN ビットを更新。 表 119 で、ACCMOD を更新。 セクション 25.5.4 : NOR 型フラッシュ/PSRAM コントローラ非同期トランザクション、セクション : SRAM/NOR 型フラッシュチップセレクト制御レジスタ 1.4 (FSMC_BCR1.4)、セクション : SRAM/NOR 型フラッシュチップセレクトタイミングレジスタ 1.4 (FSMC_BTR1.4) および セクション : SRAM/NOR 型フラッシュ書き込みタイミングレジスタ 1.4 (FSMC_BWTR1.4) を更新 セクション 26.3.7 : DMA リクエストを更新。DMA レジスタ名を訂正し、“チャンネル”の表現を“ストリーム”に変更。 セクション 26.1 : I^2C の概要で、周波数についての情報を更新。 SCL マスタクロック生成の下位区分を追加。 セクション 26.6.2 : I^2C 制御レジスタ 2(I2C_CR2) で、FREQ ビットを更新。 セクション 26.6.9 : I^2C TRISE レジスタ (I2C_TRISE) で、TRISE ビットを更新。 セクション 26.2 : I^2C の主な機能 および セクション 26.6.2 : I^2C 制御レジスタ 2(I2C_CR2) を更新。 セクション 27.3.13 : DMA を使用した連続通信で、注を削除。 セクション 27.3.8 : LIN (local interconnection network) モードを更新。 欠落した 図 265 : 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 を追加。 SPI_CR2 レジスタで、ERRIE ビットの定義を更新。 図 291 : JTAG TAP 接続を更新。 セクション 30.4.3 : JTAG ピンでの内部プルアップ/プルダウンで、JTDO を更新。 表 119 : FSMC_BTRx ビットフィールドで、BUSTURN の定義を更新。 セクション 25 : フレキシブルスタティックメモリコントローラ (FSMC) の導入行を更新。

表 211. 改版履歴 (続き)

日付	リビジョン	変更内容
2014 年 3 月 3 日	9 (続き)	セクション 29 : セキュアデジタル入力/出力インタフェース (SDIO) の導入行を更新。 セクション 30.6.1 : MCU デバイス ID コードで、REV_ID と DEV_ID を更新。 セクション 30.16.3 : デバッグMCU 設定レジスタで、DBG_STOP ビットの説明を更新。 セクション 31.1.1 : フラッシュサイズレジスタで、F_SIZE を更新。
2014 年 5 月 9 日	10	セクション : 関連ドキュメントを更新。 表 1 : 製品カテゴリとメモリサイズの概要、表 3 : STM32L15xxx 製品カテゴリ および 表 4 : STM32L162xx 製品カテゴリ を更新。 注 : 1 を更新。 セクション : データ EEPROM のダブルワード書き込みで、最終項目を更新。 表 18 : オプションバイトの説明 直前の文を削除。 セクション 5.2.2 : ブラウンアウトリセットのタイトルを更新。 セクション 5.2.4 : 内部電圧基準 (VREFINT) における第 1 パラグラフの最終文を更新。 セクション : 較正 を更新。 図 111 : キャプチャ/比較チャネル (チャネル 1) の出力ステージを更新。 セクション 17.3.10 : ワンパルスモードで、IC2S を CC2S で置換し、tPULSE の定義を更新。 セクション 17.4.7 : TIMx およびキャプチャ/比較モードレジスタ 1 (TIMx_CCMR1) と セクション 18.5.6 : TIM10/11 キャプチャ/比較モードレジスタ 1 (TIMx_CCMR1) で、IC1F ビットに関連する注を削除。 セクション 20.3.14 : 較正されたクロックの出力で、注を追加。 セクション : DBGMCU_IDCODE で、Rev X を追加。

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。

ST の販売条件に規定される場合を除き、STは、商品性、特定目的への適合性(その他管轄の如何を問わず法律で認められる同等のもの)、若しくは特許権、著作権その他の知的財産権の侵害に関する黙示の保証を含め、ST 製品の使用または販売に関する明示または黙示の保証をすべて放棄します。

ST製品は、(A) 生命維持装置、能動型埋め込み機器、若しくは機能安全要件を備える装置などの安全が重大な結果に関わる用途、(B) 航空用途、(C) 車載用途または車載環境、及び／または (D) 航空宇宙用途又は航空宇宙環境での使用のために設計、認可されたものではありません。ST製品がこれらの用途のために設計されていない場合、たとえSTが事前に書面で使用用途を知らされていたとしても、STの製品設計仕様に従い、「自動車、自動車の安全性、または医療」の産業分野向けであるとSTが明確に指定する場合を除き、ST製品のご使用は購入者ご自身の責任で行って頂くものとします。

ESCC、QML、JAN規格の正式な認証を得た製品は、該当する政府機関によって航空宇宙用途への適合性があると判断されております。

本書に説明されている記述または技術的特徴とは異なる条件でST 製品が再販された場合、本書に記載されるST 製品またはサービスについてのST による保証は直ちに失われるものとし、いかなる形であれ、ST の責任を生ぜしめること、若しくはその責任が延長されることはないものとします。

ST およびST ロゴは各国におけるSTMicroelectronics の商標または登録商標です。

本書の情報はそれ以前に提供された全ての情報に優先します。

ST ロゴはSTMicroelectronics の登録商標です。その他の名称は、それぞれの所有者に帰属します。

© 2015 STMicroelectronics - All rights reserved

STMicroelectronics group of companies

Australia - Belgium - Brazil - Canada - China - Czech Republic - Finland - France - Germany - Hong Kong - India - Israel - Italy - Japan - Malaysia - Malta - Morocco - Philippines - Singapore - Spain - Sweden - Switzerland - United Kingdom - United States of America

www.st.com